

---

# **nifcloud-sdk-python Documentation**

***Release 1.8.0***

**FUJITSU CLOUD TECHNOLOGIES**

**Aug 10, 2023**



---

## Contents

---

<b>1</b>	<b>Available Services</b>	<b>3</b>
1.1	computing . . . . .	3
1.2	dns . . . . .	331
1.3	ess . . . . .	340
1.4	hatoba . . . . .	351
1.5	nas . . . . .	424
1.6	rdb . . . . .	452
1.7	script . . . . .	556
1.8	serviceactivity . . . . .	558
1.9	storage . . . . .	563
<b>2</b>	<b>Indices and tables</b>	<b>599</b>
	<b>Index</b>	<b>601</b>



The NIFCLOUD SDK for Python is data-driven SDK. It works by feeding AWS-SDK-compatible model JSONs to botocore module.

Contents:



## 1.1 computing

### 1.1.1 Client

**class** `computing.Client`

A low-level client representing NIFCLOUD Computing

```
client = session.create_client('computing')
```

These are the available methods:

*computing* / Client / `allocate_address`

### `allocate_address`

`computing.Client.allocate_address(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.allocate_address(
    Domain='string',
    InstanceId='string',
    NiftyPrivateIp=True|False,
    Placement={
        'AvailabilityZone': 'string'
    }
)
```

#### Parameters

- **Domain** (*string*) –

- **InstanceId** (*string*) –
- **NiftyPrivateIp** (*boolean*) –
- **Placement** (*dict*) –
  - **AvailabilityZone** (*string*) –

**Return type** dict

#### Returns

##### Response Syntax

```
{
  'Placement': {
    'AvailabilityZone': 'string'
  },
  'PrivateIpAddress': 'string',
  'PublicIp': 'string',
  'RequestId': 'string'
}
```

##### Response Structure

- (*dict*) –
  - **Placement** (*dict*) –
    - \* **AvailabilityZone** (*string*) –
  - **PrivateIpAddress** (*string*) –
  - **PublicIp** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / `associate_address`

## associate\_address

`computing.Client.associate_address` (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.associate_address(
    AllocationId='string',
    AllowReassociation=True|False,
    InstanceId='string',
    NetworkInterfaceId='string',
    NiftyReboot='force'|'true'|'false',
    PrivateIpAddress='string',
    PublicIp='string'
)
```

#### Parameters

- **AllocationId** (*string*) –
- **AllowReassociation** (*boolean*) –
- **InstanceId** (*string*) – [REQUIRED]



- **NetworkInterfaceId**(*string*) –
- **NiftyReboot**(*string*) –
- **PrivateIpAddress**(*string*) –
- **PublicIp**(*string*) –

**Return type** dict

#### Returns

##### Response Syntax

```
{
  'RequestId': 'string',
  'Return': True|False
}
```

##### Response Structure

- (*dict*) –
  - **RequestId**(*string*) –
  - **Return**(*boolean*) –

*computing* / Client / `associate_multi_ip_address_group`

### `associate_multi_ip_address_group`

`computing.Client.associate_multi_ip_address_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.associate_multi_ip_address_group(
    InstanceUniqueId='string',
    MultiIpAddressGroupId='string',
    NiftyReboot='force'|'true'|'false'
)
```

#### Parameters

- **InstanceUniqueId**(*string*) – [REQUIRED]
- **MultiIpAddressGroupId**(*string*) – [REQUIRED]
- **NiftyReboot**(*string*) –

**Return type** dict

#### Returns

##### Response Syntax

```
{
  'RequestId': 'string',
  'Return': True|False
}
```

##### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / `associate_route_table`

## `associate_route_table`

`computing.Client.associate_route_table(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.associate_route_table(  
    Agreement=True|False,  
    RouteTableId='string',  
    RouterId='string',  
    RouterName='string',  
    SubnetId='string'  
)
```

### Parameters

- **Agreement** (*boolean*) –
- **RouteTableId** (*string*) – [REQUIRED]
- **RouterId** (*string*) –
- **RouterName** (*string*) –
- **SubnetId** (*string*) –

**Return type** dict

### Returns

#### Response Syntax

```
{  
    'AssociationId': 'string',  
    'RequestId': 'string'  
}
```

#### Response Structure

- (*dict*) –
  - **AssociationId** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / `associate_users`

## `associate_users`

`computing.Client.associate_users(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

## Request Syntax

```
response = client.associate_users(
    FunctionName='LB',
    Users=[
        {
            'UserId': 'string'
        },
    ]
)
```

### Parameters

- **FunctionName** (*string*) – [REQUIRED]
- **Users** (*list*) – [REQUIRED]
  - (*dict*) –
    - \* **UserId** (*string*) – [REQUIRED]

Return type dict

### Returns

#### Response Syntax

```
{
    'AssociateUsersResult': {
        'Users': [
            {
                'UserId': 'string'
            },
        ]
    },
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

### Response Structure

- (*dict*) –
  - **AssociateUsersResult** (*dict*) –
    - \* **Users** (*list*) –
      - (*dict*) –
        - **UserId** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*computing* / Client / attach\_iso\_image

## attach\_iso\_image

computing.Client.**attach\_iso\_image**(\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.attach_iso_image(  
    InstanceUniqueId='string',  
    IsoImageId='string'  
)
```

#### Parameters

- **InstanceUniqueId** (*string*) – [REQUIRED]
- **IsoImageId** (*string*) – [REQUIRED]

**Return type** dict

#### Returns

##### Response Syntax

```
{  
    'RequestId': 'string',  
    'Return': True|False  
}
```

##### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / attach\_network\_interface

## attach\_network\_interface

`computing.Client.attach_network_interface(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.attach_network_interface(  
    InstanceId='string',  
    InstanceUniqueId='string',  
    NetworkInterfaceId='string',  
    NiftyReboot='force'|'true'|'false'  
)
```

#### Parameters

- **InstanceId** (*string*) –
- **InstanceUniqueId** (*string*) –
- **NetworkInterfaceId** (*string*) – [REQUIRED]
- **NiftyReboot** (*string*) –

**Return type** dict

#### Returns

##### Response Syntax

```
{
    'AttachmentId': 'string',
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (*dict*) –
  - **AttachmentId** (*string*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / attach\_volume

### attach\_volume

`computing.Client.attach_volume(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.attach_volume(
    Device='string',
    InstanceId='string',
    VolumeId='string'
)
```

#### Parameters

- **Device** (*string*) –
- **InstanceId** (*string*) – [REQUIRED]
- **VolumeId** (*string*) – [REQUIRED]

**Return type** dict

#### Returns

#### Response Syntax

```
{
    'AttachTime': 'string',
    'Device': 'string',
    'InstanceId': 'string',
    'InstanceUniqueId': 'string',
    'RequestId': 'string',
    'Status': 'string',
    'VolumeId': 'string',
    'VolumeUniqueId': 'string'
}
```

#### Response Structure

- (*dict*) –
  - **AttachTime** (*string*) –

- **Device** (*string*) -
- **InstanceId** (*string*) -
- **InstanceUniqueId** (*string*) -
- **RequestId** (*string*) -
- **Status** (*string*) -
- **VolumeId** (*string*) -
- **VolumeUniqueId** (*string*) -

*computing* / Client / `authorize_security_group_ingress`

## `authorize_security_group_ingress`

`computing.Client.authorize_security_group_ingress (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.authorize_security_group_ingress(
    GroupName='string',
    IpPermissions=[
        {
            'Description': 'string',
            'FromPort': 123,
            'InOut': 'IN'|'OUT',
            'IpProtocol': 'ANY'|'TCP'|'UDP'|'ICMP'|'SSH'|'HTTP'|'HTTPS'|'RDP'|'GRE
→ '| 'ESP'|'AH'|'VRRP'|'L2TP'|'ICMPv6-all',
            'ListOfRequestGroups': [
                {
                    'GroupName': 'string',
                    'UserId': 'string'
                },
            ],
            'ListOfRequestIpRanges': [
                {
                    'CidrIp': 'string'
                },
            ],
            'ToPort': 123
        },
    ],
    UserId='string'
)
```

### Parameters

- **GroupName** (*string*) - [REQUIRED]
- **IpPermissions** (*list*) -
  - (*dict*) -
    - \* **Description** (*string*) -
    - \* **FromPort** (*integer*) -
    - \* **InOut** (*string*) -

- \* **IpProtocol** (*string*) –
- \* **ListOfRequestGroups** (*list*) –
  - (*dict*) –
  - **GroupName** (*string*) –
  - **UserId** (*string*) –
- \* **ListOfRequestIpRanges** (*list*) –
  - (*dict*) –
  - **CidrIp** (*string*) –
- \* **ToPort** (*integer*) –
- **UserId** (*string*) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / can\_paginate

### can\_paginate

`computing.Client.can_paginate(operation_name)`

Check if an operation can be paginated.

**Parameters** **operation\_name** (*string*) – The operation name. This is the same name as the method name on the client. For example, if the method name is `create_foo`, and you'd normally invoke the operation as `client.create_foo(**kwargs)`, if the `create_foo` operation can be paginated, you can use the call `client.get_paginator("create_foo")`.

**Returns** True if the operation can be paginated, False otherwise.

*computing* / Client / cancel\_copy\_instances

### cancel\_copy\_instances

`computing.Client.cancel_copy_instances(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.cancel_copy_instances(  
    InstanceId='string'  
)
```

**Parameters** `InstanceId` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{  
    'RequestId': 'string'  
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –

*computing* / Client / cancel\_upload

## cancel\_upload

`computing.Client.cancel_upload(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.cancel_upload(  
    ConversionTaskId='string'  
)
```

**Parameters** `ConversionTaskId` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{  
    'RequestId': 'string',  
    'Return': True|False  
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / clear\_load\_balancer\_session

## clear\_load\_balancer\_session

`computing.Client.clear_load_balancer_session(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax



```
response = client.clear_load_balancer_session(
    InstancePort=123,
    LoadBalancerName='string',
    LoadBalancerPort=123
)
```

**Parameters**

- **InstancePort** (*integer*) – [REQUIRED]
- **LoadBalancerName** (*string*) – [REQUIRED]
- **LoadBalancerPort** (*integer*) – [REQUIRED]

**Return type** dict**Returns****Response Syntax**

```
{
    'ClearLoadBalancerSessionResult': 'string',
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

**Response Structure**

- (*dict*) –
  - **ClearLoadBalancerSessionResult** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*computing* / Client / close**close**

`computing.Client.close()`  
 Closes underlying endpoint connections.

*computing* / Client / configure\_health\_check**configure\_health\_check**

`computing.Client.configure_health_check(**kwargs)`  
 See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.configure_health_check(
    HealthCheck={
        'HealthyThreshold': 123,
        'Interval': 123,
        'Target': 'string',
        'Timeout': 123,
        'UnhealthyThreshold': 123
    },
    InstancePort=123,
    LoadBalancerName='string',
```

(continues on next page)

(continued from previous page)

```
LoadBalancerPort=123
)
```

**Parameters**

- **HealthCheck** (*dict*) – [REQUIRED]
  - **HealthyThreshold** (*integer*) –
  - **Interval** (*integer*) – [REQUIRED]
  - **Target** (*string*) – [REQUIRED]
  - **Timeout** (*integer*) –
  - **UnhealthyThreshold** (*integer*) – [REQUIRED]
- **InstancePort** (*integer*) – [REQUIRED]
- **LoadBalancerName** (*string*) – [REQUIRED]
- **LoadBalancerPort** (*integer*) – [REQUIRED]

**Return type** dict**Returns****Response Syntax**

```
{
  'ConfigureHealthCheckResult': {
    'HealthCheck': {
      'HealthyThreshold': 123,
      'Interval': 123,
      'Target': 'string',
      'UnhealthyThreshold': 123
    }
  },
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

**Response Structure**

- (*dict*) –
  - **ConfigureHealthCheckResult** (*dict*) –
    - \* **HealthCheck** (*dict*) –
      - **HealthyThreshold** (*integer*) –
      - **Interval** (*integer*) –
      - **Target** (*string*) –
      - **UnhealthyThreshold** (*integer*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*computing* / Client / copy\_from\_backup\_instance

## copy\_from\_backup\_instance

computing.Client.**copy\_from\_backup\_instance** (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.copy_from_backup_instance(
    AccountingType='1'|'2',
    BackupInstanceUniqueId='string',
    Description='string',
    DisableApiTermination=True|False,
    InstanceId='string',
    InstanceType='e-mini'|'h2-mini'|'mini'|'c-small'|'e-small'|'h2-small'|'small'|
    ↪ 'c-small2'|'e-small2'|'h2-small2'|'small2'|'c-small4'|'e-small4'|'h2-small4'|
    ↪ 'small4'|'e-small8'|'h2-small8'|'small8'|'e-small16'|'h2-small16'|'small16'|'c-
    ↪ medium'|'e-medium'|'h2-medium'|'medium'|'c-medium4'|'e-medium4'|'h2-medium4'|
    ↪ 'medium4'|'c-medium8'|'e-medium8'|'h2-medium8'|'medium8'|'e-medium16'|'h2-
    ↪ medium16'|'medium16'|'e-medium24'|'h2-medium24'|'medium24'|'c-large'|'e-large'|
    ↪ 'h2-large'|'large'|'c-large8'|'e-large8'|'h2-large8'|'large8'|'e-large16'|'h2-
    ↪ large16'|'large16'|'e-large24'|'h2-large24'|'large24'|'e-large32'|'h2-large32'|
    ↪ 'large32'|'e-extra-large8'|'h2-extra-large8'|'extra-large8'|'e-extra-large16'|
    ↪ 'h2-extra-large16'|'extra-large16'|'e-extra-large24'|'h2-extra-large24'|'extra-
    ↪ large24'|'e-extra-large32'|'h2-extra-large32'|'extra-large32'|'e-extra-large48'|
    ↪ 'h2-extra-large48'|'extra-large48'|'e-double-large16'|'h2-double-large16'|
    ↪ 'double-large16'|'e-double-large24'|'h2-double-large24'|'double-large24'|'e-
    ↪ double-large32'|'h2-double-large32'|'double-large32'|'e-double-large48'|'h2-
    ↪ double-large48'|'double-large48'|'e-double-large64'|'h2-double-large64'|'double-
    ↪ large64'|'e-double-large96'|'h2-double-large96'|'double-large96'|'h2-triple-
    ↪ large32'|'triple-large32'|'h2-triple-large48'|'triple-large48'|'h2-triple-
    ↪ large64'|'triple-large64'|'h2-triple-large96'|'triple-large96'|'h2-triple-
    ↪ large128'|'triple-large128'|'h2-quad-large64'|'quad-large64'|'h2-quad-large96'|
    ↪ 'quad-large96'|'h2-quad-large128'|'quad-large128'|'h2-septa-large128'|'septa-
    ↪ large128',
    NetworkInterface=[
        {
            'IpAddress': 'string',
            'NetworkId': 'string',
            'NetworkName': 'string'
        },
    ],
    SecurityGroup=[
        'string',
    ]
)
```

### Parameters

- **AccountingType** (*string*) –
- **BackupInstanceUniqueId** (*string*) – [REQUIRED]
- **Description** (*string*) –
- **DisableApiTermination** (*boolean*) –
- **InstanceId** (*string*) –
- **InstanceType** (*string*) –
- **NetworkInterface** (*list*) –
  - (*dict*) –
  - \* **IpAddress** (*string*) –

- \* **NetworkId** (*string*) –
- \* **NetworkName** (*string*) –
- **SecurityGroup** (*list*) –
  - (*string*) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'GroupSet': [
    {
      'GroupId': 'string'
    },
  ],
  'Instance': {
    'AccountingType': 'string',
    'Admin': 'string',
    'AmiLaunchIndex': 'string',
    'Architecture': 'string',
    'BlockDeviceMapping': [
      {
        'DeviceName': 'string',
        'Ebs': {
          'AttachTime': 'string',
          'DeleteOnTermination': 'string',
          'Status': 'string',
          'VolumeId': 'string',
          'VolumeUniqueId': 'string'
        }
      },
    ],
    'Description': 'string',
    'DnsName': 'string',
    'ImageId': 'string',
    'InstanceId': 'string',
    'InstanceLifecycle': 'string',
    'InstanceState': {
      'Code': 123,
      'Name': 'string'
    },
    'InstanceType': 'string',
    'InstanceUniqueId': 'string',
    'IpAddress': 'string',
    'IpAddressV6': 'string',
    'IpType': 'string',
    'KernelId': 'string',
    'KeyName': 'string',
    'LaunchTime': 'string',
    'Monitoring': {
      'State': 'string'
    },
    'NetworkInterfaceSet': [
      {
        'Association': {
          'AllocationId': 'string',
          'AssociationId': 'string',
```

(continues on next page)

(continued from previous page)

```

        'IpOwnerId': 'string',
        'PublicDnsName': 'string',
        'PublicIp': 'string',
        'PublicIPv6': 'string'
    },
    'Attachment': {
        'AttachTime': 'string',
        'AttachmentId': 'string',
        'DeleteOnTermination': 'string',
        'DeviceIndex': 'string',
        'InstanceId': 'string',
        'InstanceOwnerId': 'string',
        'Status': 'string'
    },
    'Description': 'string',
    'GroupSet': [
        {
            'GroupId': 'string'
        },
    ],
    'MacAddress': 'string',
    'NetworkInterfaceId': 'string',
    'NiftyNetworkId': 'string',
    'NiftyNetworkName': 'string',
    'OwnerId': 'string',
    'PrivateDnsName': 'string',
    'PrivateIpAddress': 'string',
    'PrivateIpAddressesSet': [
        {
            'Association': {
                'AllocationId': 'string',
                'AssociationId': 'string',
                'IpOwnerId': 'string',
                'PublicDnsName': 'string',
                'PublicIp': 'string',
                'PublicIPv6': 'string'
            },
            'Primary': 'string',
            'PrivateDnsName': 'string',
            'PrivateIpAddress': 'string'
        },
    ],
    'SourceDestCheck': 'string',
    'Status': 'string',
    'SubnetId': 'string',
    'VpcId': 'string'
    },
],
'NiftyPrivateIpType': 'string',
'Placement': {
    'AvailabilityZone': 'string'
},
'Platform': 'string',
'PrivateDnsName': 'string',
'PrivateIpAddress': 'string',
'PrivateIpAddressV6': 'string',
'ProductCodes': [

```

(continues on next page)

(continued from previous page)

```

        {
            'ProductCode': 'string'
        },
    ],
    'RamdiskId': 'string',
    'Reason': 'string',
    'RootDeviceName': 'string',
    'RootDeviceType': 'string',
    'SpotInstanceRequestId': 'string',
    'StateReason': {
        'Code': 'string',
        'Message': 'string'
    },
    'SubnetId': 'string',
    'VpcId': 'string'
},
'OwnerId': 'string',
'RequestId': 'string',
'ReservationId': 'string',
'Return': True|False
}

```

**Response Structure**

- *(dict)* –
  - **GroupSet** (*list*) –
    - \* *(dict)* –
      - **GroupId** (*string*) –
  - **Instance** (*dict*) –
    - \* **AccountingType** (*string*) –
    - \* **Admin** (*string*) –
    - \* **AmiLaunchIndex** (*string*) –
    - \* **Architecture** (*string*) –
    - \* **BlockDeviceMapping** (*list*) –
      - *(dict)* –
      - **DeviceName** (*string*) –
      - **Ebs** (*dict*) –
      - **AttachTime** (*string*) –
      - **DeleteOnTermination** (*string*) –
      - **Status** (*string*) –
      - **VolumeId** (*string*) –
      - **VolumeUniqueId** (*string*) –
    - \* **Description** (*string*) –
    - \* **DnsName** (*string*) –
    - \* **ImageId** (*string*) –

- \* **InstanceId** (*string*) –
- \* **InstanceLifecycle** (*string*) –
- \* **InstanceState** (*dict*) –
  - **Code** (*integer*) –
  - **Name** (*string*) –
- \* **InstanceType** (*string*) –
- \* **InstanceUniqueId** (*string*) –
- \* **IpAddress** (*string*) –
- \* **IpAddressV6** (*string*) –
- \* **IpType** (*string*) –
- \* **KernelId** (*string*) –
- \* **KeyName** (*string*) –
- \* **LaunchTime** (*string*) –
- \* **Monitoring** (*dict*) –
  - **State** (*string*) –
- \* **NetworkInterfaceSet** (*list*) –
  - (*dict*) –
  - **Association** (*dict*) –
  - **AllocationId** (*string*) –
  - **AssociationId** (*string*) –
  - **IpOwnerId** (*string*) –
  - **PublicDnsName** (*string*) –
  - **PublicIp** (*string*) –
  - **PublicIpV6** (*string*) –
  - **Attachment** (*dict*) –
  - **AttachTime** (*string*) –
  - **AttachmentId** (*string*) –
  - **DeleteOnTermination** (*string*) –
  - **DeviceIndex** (*string*) –
  - **InstanceId** (*string*) –
  - **InstanceOwnerId** (*string*) –
  - **Status** (*string*) –
  - **Description** (*string*) –
  - **GroupSet** (*list*) –
  - (*dict*) –
  - **GroupId** (*string*) –

- **MacAddress** (*string*) –
- **NetworkInterfaceId** (*string*) –
- **NiftyNetworkId** (*string*) –
- **NiftyNetworkName** (*string*) –
- **OwnerId** (*string*) –
- **PrivateDnsName** (*string*) –
- **PrivateIpAddress** (*string*) –
- **PrivateIpAddressesSet** (*list*) –
- (*dict*) –
- **Association** (*dict*) –
- **AllocationId** (*string*) –
- **AssociationId** (*string*) –
- **IpOwnerId** (*string*) –
- **PublicDnsName** (*string*) –
- **PublicIp** (*string*) –
- **PublicIPv6** (*string*) –
- **Primary** (*string*) –
- **PrivateDnsName** (*string*) –
- **PrivateIpAddress** (*string*) –
- **SourceDestCheck** (*string*) –
- **Status** (*string*) –
- **SubnetId** (*string*) –
- **VpcId** (*string*) –
- \* **NiftyPrivateIpType** (*string*) –
- \* **Placement** (*dict*) –
  - **AvailabilityZone** (*string*) –
- \* **Platform** (*string*) –
- \* **PrivateDnsName** (*string*) –
- \* **PrivateIpAddress** (*string*) –
- \* **PrivateIpAddressV6** (*string*) –
- \* **ProductCodes** (*list*) –
  - (*dict*) –
  - **ProductCode** (*string*) –
- \* **RamdiskId** (*string*) –
- \* **Reason** (*string*) –
- \* **RootDeviceName** (*string*) –



- \* **RootDeviceType** (*string*) –
- \* **SpotInstanceRequestId** (*string*) –
- \* **StateReason** (*dict*) –
  - **Code** (*string*) –
  - **Message** (*string*) –
- \* **SubnetId** (*string*) –
- \* **VpcId** (*string*) –
- **OwnerId** (*string*) –
- **RequestId** (*string*) –
- **ReservationId** (*string*) –
- **Return** (*boolean*) –

*computing* / Client / copy\_instances

## copy\_instances

`computing.Client.copy_instances (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.copy_instances(
    CopyCount=123,
    CopyInstance={
        'AccountingType': '1'|'2',
        'InstanceName': 'string',
        'InstanceType': 'e-mini'|'h2-mini'|'mini'|'c-small'|'e-small'|'h2-small'|
        ↪ 'small'|'c-small2'|'e-small2'|'h2-small2'|'small2'|'c-small4'|'e-small4'|'h2-
        ↪ small4'|'small4'|'e-small8'|'h2-small8'|'small8'|'e-small16'|'h2-small16'|
        ↪ 'small16'|'c-medium'|'e-medium'|'h2-medium'|'medium'|'c-medium4'|'e-medium4'|
        ↪ 'h2-medium4'|'medium4'|'c-medium8'|'e-medium8'|'h2-medium8'|'medium8'|'e-
        ↪ medium16'|'h2-medium16'|'medium16'|'e-medium24'|'h2-medium24'|'medium24'|'c-
        ↪ large'|'e-large'|'h2-large'|'large'|'c-large8'|'e-large8'|'h2-large8'|'large8'|
        ↪ 'e-large16'|'h2-large16'|'large16'|'e-large24'|'h2-large24'|'large24'|'e-large32
        ↪ '| 'h2-large32'|'large32'|'e-extra-large8'|'h2-extra-large8'|'extra-large8'|'e-
        ↪ extra-large16'|'h2-extra-large16'|'extra-large16'|'e-extra-large24'|'h2-extra-
        ↪ large24'|'extra-large24'|'e-extra-large32'|'h2-extra-large32'|'extra-large32'|
        ↪ 'e-extra-large48'|'h2-extra-large48'|'extra-large48'|'e-double-large16'|'h2-
        ↪ double-large16'|'double-large16'|'e-double-large24'|'h2-double-large24'|'double-
        ↪ large24'|'e-double-large32'|'h2-double-large32'|'double-large32'|'e-double-
        ↪ large48'|'h2-double-large48'|'double-large48'|'e-double-large64'|'h2-double-
        ↪ large64'|'double-large64'|'e-double-large96'|'h2-double-large96'|'double-large96
        ↪ '| 'h2-triple-large32'|'triple-large32'|'h2-triple-large48'|'triple-large48'|'h2-
        ↪ triple-large64'|'triple-large64'|'h2-triple-large96'|'triple-large96'|'h2-
        ↪ triple-large128'|'triple-large128'|'h2-quad-large64'|'quad-large64'|'h2-quad-
        ↪ large96'|'quad-large96'|'h2-quad-large128'|'quad-large128'|'h2-septa-large128'|
        ↪ 'septa-large128',
        'IpType': 'static'|'none',
        'ListOfRequestLoadBalancers': [
            {
                'InstancePort': 123,
```

(continues on next page)

(continued from previous page)

```

        'LoadBalancerName': 'string',
        'LoadBalancerPort': 123
    },
],
'ListOfRequestSecurityGroup': [
    'string',
],
'RequestPlacement': {
    'AvailabilityZone': 'string',
    'RegionName': 'string'
}
},
InstanceId='string',
NetworkInterface=[
    {
        'DeviceIndex': 123,
        'IpAddress': 'string',
        'ListOfRequestSecurityGroupId': [
            'string',
        ],
        'NetworkId': 'string',
        'NetworkName': 'string'
    },
]
)

```

**Parameters**

- **CopyCount** (*integer*) –
- **CopyInstance** (*dict*) – **[REQUIRED]**
  - **AccountingType** (*string*) –
  - **InstanceName** (*string*) – **[REQUIRED]**
  - **InstanceType** (*string*) –
  - **IpType** (*string*) –
  - **ListOfRequestLoadBalancers** (*list*) –
    - \* (*dict*) –
      - **InstancePort** (*integer*) –
      - **LoadBalancerName** (*string*) –
      - **LoadBalancerPort** (*integer*) –
  - **ListOfRequestSecurityGroup** (*list*) –
    - \* (*string*) –
  - **RequestPlacement** (*dict*) –
    - \* **AvailabilityZone** (*string*) –
    - \* **RegionName** (*string*) –
- **InstanceId** (*string*) – **[REQUIRED]**
- **NetworkInterface** (*list*) –
  - (*dict*) –
    - \* **DeviceIndex** (*integer*) –
    - \* **IpAddress** (*string*) –
    - \* **ListOfRequestSecurityGroupId** (*list*) –
      - (*string*) –
    - \* **NetworkId** (*string*) –
    - \* **NetworkName** (*string*) –

**Return type** dict**Returns**

### Response Syntax

```
{
  'CopyInstanceSet': [
    {
      'InstanceId': 'string',
      'InstanceState': 'string',
      'InstanceUniqueId': 'string'
    },
  ],
  'RequestId': 'string'
}
```

### Response Structure

- (dict) –
  - **CopyInstanceSet** (list) –
    - \* (dict) –
      - **InstanceId** (string) –
      - **InstanceState** (string) –
      - **InstanceUniqueId** (string) –
  - **RequestId** (string) –

*computing* / Client / create\_backup\_instances

### create\_backup\_instances

`computing.Client.create_backup_instances(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.create_backup_instances(
    InstanceBackupRuleId='string'
)
```

**Parameters** `InstanceBackupRuleId` (string) – **[REQUIRED]**

**Return type** dict

**Returns**

### Response Syntax

```
{
  'InstanceBackupRule': {
    'AvailabilityZone': 'string',
    'BackupInstanceMaxCount': 123,
    'Description': 'string',
    'InstanceBackupRuleId': 'string',
    'InstanceBackupRuleName': 'string',
    'InstancesSet': [
      {
        'BackupInstancesSet': [
          {
            'BackupInstanceCreateTime': 'string',
            'BackupInstanceUniqueId': 'string',
            'Status': 'string'
          },
        ],
      },
    ],
  },
}
```

(continues on next page)

(continued from previous page)

```

        'InstanceId': 'string',
        'InstanceUniqueId': 'string'
    },
    ],
    'RegionName': 'string',
    'Status': 'string',
    'TimeSlotId': 'string'
},
'RequestId': 'string',
'Return': True|False
}

```

**Response Structure**

- (dict) –
  - **InstanceBackupRule** (dict) –
    - \* **AvailabilityZone** (string) –
    - \* **BackupInstanceMaxCount** (integer) –
    - \* **Description** (string) –
    - \* **InstanceBackupRuleId** (string) –
    - \* **InstanceBackupRuleName** (string) –
    - \* **InstancesSet** (list) –
      - (dict) –
      - **BackupInstancesSet** (list) –
      - (dict) –
      - **BackupInstanceCreateTime** (string) –
      - **BackupInstanceUniqueId** (string) –
      - **Status** (string) –
      - **InstanceId** (string) –
      - **InstanceUniqueId** (string) –
    - \* **RegionName** (string) –
    - \* **Status** (string) –
    - \* **TimeSlotId** (string) –
  - **RequestId** (string) –
  - **Return** (boolean) –

*computing* / Client / create\_customer\_gateway

**create\_customer\_gateway**

`computing.Client.create_customer_gateway(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```

response = client.create_customer_gateway(
    BgpAsn=123,
    IpAddress='string',
    NiftyCustomerGatewayDescription='string',
    NiftyCustomerGatewayName='string',
    NiftyLanSideCidrBlock='string',
    NiftyLanSideIpAddress='string',
    Type='IPsec'|'IPsec VTI'|'L2TPv3 / IPsec'
)

```

**Parameters**

- **BgpAsn** (*integer*) –
- **IpAddress** (*string*) – [REQUIRED]
- **NiftyCustomerGatewayDescription** (*string*) –
- **NiftyCustomerGatewayName** (*string*) –
- **NiftyLanSideCidrBlock** (*string*) –
- **NiftyLanSideIpAddress** (*string*) –
- **Type** (*string*) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'CustomerGateway': {
    'CreatedTime': datetime(2015, 1, 1),
    'CustomerGatewayId': 'string',
    'IpAddress': 'string',
    'NiftyCustomerGatewayDescription': 'string',
    'NiftyCustomerGatewayName': 'string',
    'NiftyLanSideCidrBlock': 'string',
    'NiftyLanSideIpAddress': 'string',
    'State': 'string',
    'TagSet': [
      {
        'Key': 'string',
        'Value': 'string'
      },
    ],
  },
  'RequestId': 'string'
}
```

#### Response Structure

- (*dict*) –
  - **CustomerGateway** (*dict*) –
    - \* **CreatedTime** (*datetime*) –
    - \* **CustomerGatewayId** (*string*) –
    - \* **IpAddress** (*string*) –
    - \* **NiftyCustomerGatewayDescription** (*string*) –
    - \* **NiftyCustomerGatewayName** (*string*) –
    - \* **NiftyLanSideCidrBlock** (*string*) –
    - \* **NiftyLanSideIpAddress** (*string*) –
    - \* **State** (*string*) –
    - \* **TagSet** (*list*) –
      - (*dict*) –
      - **Key** (*string*) –
      - **Value** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / create\_dhcp\_options

### create\_dhcp\_options

`computing.Client.create_dhcp_options (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.create_dhcp_options(  
    DhcpConfiguration=[  
        {  
            'Key': 'default-router'|'domain-name'|'domain-name-servers'|'ntp-  
→servers'|'netbios-name-servers'|'netbios-node-type'|'lease-time',  
            'ListOfRequestValue': [  
                'string',  
            ]  
        },  
    ]  
)
```

**Parameters** `DhcpConfiguration` (*list*) – [REQUIRED]

- (*dict*) –
  - **Key** (*string*) – [REQUIRED]
  - **ListOfRequestValue** (*list*) – [REQUIRED]
    - \* (*string*) –

**Return type** `dict`

**Returns**

#### Response Syntax

```
{  
    'DhcpOptions': {  
        'DhcpConfigurationSet': [  
            {  
                'Key': 'string',  
                'ValueSet': [  
                    {  
                        'Value': 'string'  
                    },  
                ],  
            },  
        ],  
        'DhcpOptionsId': 'string'  
    },  
    'RequestId': 'string'  
}
```

#### Response Structure

- (*dict*) –
  - **DhcpOptions** (*dict*) –
    - \* **DhcpConfigurationSet** (*list*) –
      - (*dict*) –
      - **Key** (*string*) –
      - **ValueSet** (*list*) –
        - (*dict*) –
        - **Value** (*string*) –
    - \* **DhcpOptionsId** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / create\_image

## create\_image

computing.Client.**create\_image** (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```

response = client.create_image(
    Description='string',
    InstanceId='string',
    LeftInstance=True|False,
    Name='string',
    NoReboot=True|False,
    Placement={
        'AvailabilityZone': 'string',
        'RegionName': 'string'
    }
)

```

### Parameters

- **Description** (*string*) –
- **InstanceId** (*string*) – [REQUIRED]
- **LeftInstance** (*boolean*) –
- **Name** (*string*) – [REQUIRED]
- **NoReboot** (*boolean*) –
- **Placement** (*dict*) –
  - **AvailabilityZone** (*string*) –
  - **RegionName** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```

{
    'ImageId': 'string',
    'ImageState': 'string',
    'RequestId': 'string'
}

```

### Response Structure

- (*dict*) –
  - **ImageId** (*string*) –
  - **ImageState** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / create\_instance\_backup\_rule

## create\_instance\_backup\_rule

computing.Client.**create\_instance\_backup\_rule** (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```

response = client.create_instance_backup_rule(
    BackupInstanceMaxCount=123,
    Description='string',

```

(continues on next page)

(continued from previous page)

```

InstanceBackupRuleName='string',
InstanceUniqueId=[
    'string',
],
TimeSlotId='1'|'2'|'3'|'4'|'5'|'6'|'7'|'8'|'9'|'10'|'11'|'12'
)

```

**Parameters**

- **BackupInstanceMaxCount** (*integer*) – [REQUIRED]
- **Description** (*string*) –
- **InstanceBackupRuleName** (*string*) –
- **InstanceUniqueId** (*list*) – [REQUIRED]
  - (*string*) –
- **TimeSlotId** (*string*) – [REQUIRED]

**Return type** dict**Returns****Response Syntax**

```

{
    'InstanceBackupRule': {
        'AvailabilityZone': 'string',
        'BackupInstanceMaxCount': 123,
        'Description': 'string',
        'InstanceBackupRuleId': 'string',
        'InstanceBackupRuleName': 'string',
        'InstancesSet': [
            {
                'BackupInstancesSet': 'string',
                'InstanceId': 'string',
                'InstanceUniqueId': 'string'
            },
        ],
        'RegionName': 'string',
        'Status': 'string',
        'TimeSlotId': 'string'
    },
    'RequestId': 'string',
    'Return': True|False
}

```

**Response Structure**

- (*dict*) –
  - **InstanceBackupRule** (*dict*) –
    - \* **AvailabilityZone** (*string*) –
    - \* **BackupInstanceMaxCount** (*integer*) –
    - \* **Description** (*string*) –
    - \* **InstanceBackupRuleId** (*string*) –
    - \* **InstanceBackupRuleName** (*string*) –
    - \* **InstancesSet** (*list*) –
      - (*dict*) –
      - **BackupInstancesSet** (*string*) –
      - **InstanceId** (*string*) –
      - **InstanceUniqueId** (*string*) –
    - \* **RegionName** (*string*) –
    - \* **Status** (*string*) –



- \* **TimeSlotId** (*string*) –
- **RequestId** (*string*) –
- **Return** (*boolean*) –

*computing* / Client / create\_key\_pair

## create\_key\_pair

`computing.Client.create_key_pair(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.create_key_pair(
    Description='string',
    KeyName='string',
    Password='string'
)
```

### Parameters

- **Description** (*string*) –
- **KeyName** (*string*) – [REQUIRED]
- **Password** (*string*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```
{
    'Description': 'string',
    'KeyFingerprint': 'string',
    'KeyMaterial': 'string',
    'KeyName': 'string',
    'RequestId': 'string'
}
```

### Response Structure

- (*dict*) –
  - **Description** (*string*) –
  - **KeyFingerprint** (*string*) –
  - **KeyMaterial** (*string*) –
  - **KeyName** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / create\_load\_balancer

## create\_load\_balancer

`computing.Client.create_load_balancer(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.create_load_balancer(
    AccountingType='1'|'2',
    AvailabilityZones=[
```

(continues on next page)

(continued from previous page)

```

        'string',
    ],
    IpVersion='v4'|'v6',
    Listeners=[
        {
            'BalancingType': 123,
            'InstancePort': 123,
            'LoadBalancerPort': 123,
            'Protocol': 'HTTP'|'HTTPS'|'FTP'
        },
    ],
    LoadBalancerName='string',
    NetworkVolume=123,
    PolicyType='standard'|'ats'
)

```

**Parameters**

- **AccountingType** (*string*) –
- **AvailabilityZones** (*list*) –
  - (*string*) –
- **IpVersion** (*string*) –
- **Listeners** (*list*) –
  - (*dict*) –
    - \* **BalancingType** (*integer*) –
    - \* **InstancePort** (*integer*) –
    - \* **LoadBalancerPort** (*integer*) –
    - \* **Protocol** (*string*) –
- **LoadBalancerName** (*string*) – [REQUIRED]
- **NetworkVolume** (*integer*) –
- **PolicyType** (*string*) –

**Return type** dict**Returns****Response Syntax**

```

{
    'CreateLoadBalancerResult': {
        'DNSName': 'string'
    },
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}

```

**Response Structure**

- (*dict*) –
  - **CreateLoadBalancerResult** (*dict*) –
    - \* **DNSName** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*computing* / Client / create\_multi\_ip\_address\_group

## create\_multi\_ip\_address\_group

computing.Client.create\_multi\_ip\_address\_group(\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```

response = client.create_multi_ip_address_group(
    Description='string',
    IpAddressCount=123,
    MultiIpAddressGroupName='string',
    Placement={
        'AvailabilityZone': 'string'
    }
)

```

### Parameters

- **Description** (*string*) –
- **IpAddressCount** (*integer*) – [REQUIRED]
- **MultiIpAddressGroupName** (*string*) – [REQUIRED]
- **Placement** (*dict*) – [REQUIRED]
  - **AvailabilityZone** (*string*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```

{
    'MultiIpAddressGroup': {
        'AvailabilityZone': 'string',
        'CreateTime': 'string',
        'Description': 'string',
        'InstancesSet': 'string',
        'MultiIpAddressGroupId': 'string',
        'MultiIpAddressGroupName': 'string',
        'MultiIpAddressNetwork': {
            'DefaultGateway': 'string',
            'IpAddressesSet': 'string',
            'SubnetMask': 'string'
        },
        'Status': 'string'
    },
    'RequestId': 'string',
    'Return': True|False
}

```

### Response Structure

- (*dict*) –
  - **MultiIpAddressGroup** (*dict*) –
    - \* **AvailabilityZone** (*string*) –
    - \* **CreateTime** (*string*) –
    - \* **Description** (*string*) –
    - \* **InstancesSet** (*string*) –
    - \* **MultiIpAddressGroupId** (*string*) –
    - \* **MultiIpAddressGroupName** (*string*) –
    - \* **MultiIpAddressNetwork** (*dict*) –
      - **DefaultGateway** (*string*) –

- **IpAddressesSet** (*string*) –
- **SubnetMask** (*string*) –
- \* **Status** (*string*) –
- **RequestId** (*string*) –
- **Return** (*boolean*) –

*computing* / Client / create\_network\_interface

## create\_network\_interface

`computing.Client.create_network_interface(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.create_network_interface(
    Description='string',
    IpAddress='string',
    NiftyNetworkId='string',
    Placement={
        'AvailabilityZone': 'string'
    }
)
```

### Parameters

- **Description** (*string*) –
- **IpAddress** (*string*) –
- **NiftyNetworkId** (*string*) – [REQUIRED]
- **Placement** (*dict*) –
  - **AvailabilityZone** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
    'NetworkInterface': {
        'Association': {
            'AllocationId': 'string',
            'AssociationId': 'string',
            'IpOwnerId': 'string',
            'PublicDnsName': 'string',
            'PublicIp': 'string',
            'PublicIpV6': 'string'
        },
        'Attachment': {
            'AttachTime': 'string',
            'AttachmentId': 'string',
            'DeleteOnTermination': 'string',
            'DeviceIndex': 'string',
            'InstanceId': 'string',
            'InstanceOwnerId': 'string',
            'Status': 'string'
        },
        'AvailabilityZone': 'string',
        'Description': 'string',
        'GroupSet': [
```

(continues on next page)

(continued from previous page)

```

        {
            'GroupId': 'string'
        },
    ],
    'InterfaceType': 'string',
    'Ipv6AddressesSet': [
        {
            'Ipv6Address': 'string'
        },
    ],
    'MacAddress': 'string',
    'NetworkInterfaceId': 'string',
    'NiftyNetworkId': 'string',
    'NiftyNetworkName': 'string',
    'OwnerId': 'string',
    'PrivateDnsName': 'string',
    'PrivateIpAddress': 'string',
    'PrivateIpAddressV6': 'string',
    'PrivateIpAddressesSet': [
        {
            'Association': {
                'AllocationId': 'string',
                'AssociationId': 'string',
                'IpOwnerId': 'string',
                'PublicDnsName': 'string',
                'PublicIp': 'string',
                'PublicIpV6': 'string'
            },
            'Primary': 'string',
            'PrivateDnsName': 'string',
            'PrivateIpAddress': 'string'
        },
    ],
    'RequesterId': 'string',
    'RequesterManaged': 'string',
    'SourceDestCheck': 'string',
    'Status': 'string',
    'SubnetId': 'string',
    'TagSet': [
        {
            'Key': 'string',
            'Value': 'string'
        },
    ],
    'VpcId': 'string'
},
'RequestId': 'string',
'Return': True|False
}

```

**Response Structure**

- (dict) –
  - **NetworkInterface** (dict) –
    - \* **Association** (dict) –
      - **AllocationId** (string) –
      - **AssociationId** (string) –
      - **IpOwnerId** (string) –

- **PublicDnsName** (*string*) –
- **PublicIp** (*string*) –
- **PublicIpV6** (*string*) –
- \* **Attachment** (*dict*) –
  - **AttachTime** (*string*) –
  - **AttachmentId** (*string*) –
  - **DeleteOnTermination** (*string*) –
  - **DeviceIndex** (*string*) –
  - **InstanceId** (*string*) –
  - **InstanceOwnerId** (*string*) –
  - **Status** (*string*) –
- \* **AvailabilityZone** (*string*) –
- \* **Description** (*string*) –
- \* **GroupSet** (*list*) –
  - (*dict*) –
  - **GroupId** (*string*) –
- \* **InterfaceType** (*string*) –
- \* **Ipv6AddressesSet** (*list*) –
  - (*dict*) –
  - **Ipv6Address** (*string*) –
- \* **MacAddress** (*string*) –
- \* **NetworkInterfaceId** (*string*) –
- \* **NiftyNetworkId** (*string*) –
- \* **NiftyNetworkName** (*string*) –
- \* **OwnerId** (*string*) –
- \* **PrivateDnsName** (*string*) –
- \* **PrivateIpAddress** (*string*) –
- \* **PrivateIpAddressV6** (*string*) –
- \* **PrivateIpAddressesSet** (*list*) –
  - (*dict*) –
  - **Association** (*dict*) –
  - **AllocationId** (*string*) –
  - **AssociationId** (*string*) –
  - **IpOwnerId** (*string*) –
  - **PublicDnsName** (*string*) –
  - **PublicIp** (*string*) –
  - **PublicIpV6** (*string*) –
  - **Primary** (*string*) –
  - **PrivateDnsName** (*string*) –
  - **PrivateIpAddress** (*string*) –
- \* **RequesterId** (*string*) –
- \* **RequesterManaged** (*string*) –
- \* **SourceDestCheck** (*string*) –
- \* **Status** (*string*) –
- \* **SubnetId** (*string*) –
- \* **TagSet** (*list*) –
  - (*dict*) –
  - **Key** (*string*) –
  - **Value** (*string*) –
- \* **VpcId** (*string*) –
- **RequestId** (*string*) –
- **Return** (*boolean*) –

*computing* / Client / create\_remote\_access\_vpn\_gateway

## create\_remote\_access\_vpn\_gateway

computing.Client.**create\_remote\_access\_vpn\_gateway** (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```

response = client.create_remote_access_vpn_gateway(
    AccountingType=123,
    CACertificateId='string',
    CipherSuite=[
        'string',
    ],
    Description='string',
    NetworkInterface=[
        {
            'IpAddress': 'string',
            'NetworkId': 'string'
        },
    ],
    Placement={
        'AvailabilityZone': 'string'
    },
    PoolNetworkCidr='string',
    RemoteAccessVpnGatewayName='string',
    RemoteAccessVpnGatewayType='small' | 'medium' | 'large',
    SSLCertificateId='string'
)

```

### Parameters

- **AccountingType** (*integer*) –
- **CACertificateId** (*string*) –
- **CipherSuite** (*list*) – [REQUIRED]
  - (*string*) –
- **Description** (*string*) –
- **NetworkInterface** (*list*) – [REQUIRED]
  - (*dict*) –
    - \* **IpAddress** (*string*) – [REQUIRED]
    - \* **NetworkId** (*string*) – [REQUIRED]
- **Placement** (*dict*) –
  - **AvailabilityZone** (*string*) –
- **PoolNetworkCidr** (*string*) – [REQUIRED]
- **RemoteAccessVpnGatewayName** (*string*) –
- **RemoteAccessVpnGatewayType** (*string*) –
- **SSLCertificateId** (*string*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```

{
    'RemoteAccessVpnGateway': {
        'AccountingType': 'string',
        'AuthTypeSet': [
            {
                'AuthType': 'string'
            },
        ],
    },
}

```

(continues on next page)

(continued from previous page)

```

],
'AvailabilityZone': 'string',
'CaCertificateId': 'string',
'CipherSuiteSet': [
    {
        'CipherSuite': 'string'
    },
],
'ClientDownloadEndpoint': 'string',
'ClientTunnelMode': 'string',
'CreatedTime': 'string',
'Description': 'string',
'GroupSet': 'string',
'IsConfiguredNat': 'string',
'NetworkInterfaceSet': [
    {
        'Association': {
            'AllocationId': 'string',
            'AssociationId': 'string',
            'IpOwnerId': 'string',
            'PublicDnsName': 'string',
            'PublicIp': 'string',
            'PublicIpV6': 'string'
        },
        'Attachment': {
            'AttachTime': 'string',
            'AttachmentId': 'string',
            'DeleteOnTermination': 'string',
            'DeviceIndex': 'string',
            'InstanceId': 'string',
            'InstanceOwnerId': 'string',
            'Status': 'string'
        },
        'AvailabilityZone': 'string',
        'Description': 'string',
        'GroupSet': 'string',
        'InterfaceType': 'string',
        'Ipv6AddressesSet': 'string',
        'MacAddress': 'string',
        'NetworkInterfaceId': 'string',
        'NiftyNetworkId': 'string',
        'NiftyNetworkName': 'string',
        'OwnerId': 'string',
        'PrivateDnsName': 'string',
        'PrivateIpAddress': 'string',
        'PrivateIpAddressV6': 'string',
        'PrivateIpAddressesSet': 'string',
        'RequesterId': 'string',
        'RequesterManaged': 'string',
        'SourceDestCheck': 'string',
        'Status': 'string',
        'SubnetId': 'string',
        'TagSet': 'string',
        'VpcId': 'string'
    },
],
'NextMonthAccountingType': 'string',

```

(continues on next page)



(continued from previous page)

```

        'PoolNetworkCidr': 'string',
        'PoolNetworkGatewayIpAddress': 'string',
        'RemoteAccessVpnGatewayId': 'string',
        'RemoteAccessVpnGatewayName': 'string',
        'RemoteAccessVpnGatewayType': 'string',
        'RemoteUserSet': 'string',
        'RouteTableAssociationId': 'string',
        'RouteTableId': 'string',
        'SslCertificateId': 'string',
        'Status': 'string',
        'VersionInformation': {
            'IsLatest': 'string',
            'Version': 'string'
        }
    },
    'RequestId': 'string',
    'Return': True|False
}

```

### Response Structure

- (dict) –
  - RemoteAccessVpnGateway (dict) –
    - \* AccountingType (string) –
    - \* AuthTypeSet (list) –
      - (dict) –
      - AuthType (string) –
    - \* AvailabilityZone (string) –
    - \* CaCertificateId (string) –
    - \* CipherSuiteSet (list) –
      - (dict) –
      - CipherSuite (string) –
    - \* ClientDownloadEndpoint (string) –
    - \* ClientTunnelMode (string) –
    - \* CreatedTime (string) –
    - \* Description (string) –
    - \* GroupSet (string) –
    - \* IsConfiguredNat (string) –
    - \* NetworkInterfaceSet (list) –
      - (dict) –
      - Association (dict) –
      - AllocationId (string) –
      - AssociationId (string) –
      - IpOwnerId (string) –
      - PublicDnsName (string) –
      - PublicIp (string) –
      - PublicIpV6 (string) –
      - Attachment (dict) –
      - AttachTime (string) –
      - AttachmentId (string) –
      - DeleteOnTermination (string) –
      - DeviceIndex (string) –
      - InstanceId (string) –
      - InstanceOwnerId (string) –
      - Status (string) –

- **AvailabilityZone** (*string*) –
- **Description** (*string*) –
- **GroupSet** (*string*) –
- **InterfaceType** (*string*) –
- **Ipv6AddressesSet** (*string*) –
- **MacAddress** (*string*) –
- **NetworkInterfaceId** (*string*) –
- **NiftyNetworkId** (*string*) –
- **NiftyNetworkName** (*string*) –
- **OwnerId** (*string*) –
- **PrivateDnsName** (*string*) –
- **PrivateIpAddress** (*string*) –
- **PrivateIpAddressV6** (*string*) –
- **PrivateIpAddressesSet** (*string*) –
- **RequesterId** (*string*) –
- **RequesterManaged** (*string*) –
- **SourceDestCheck** (*string*) –
- **Status** (*string*) –
- **SubnetId** (*string*) –
- **TagSet** (*string*) –
- **VpcId** (*string*) –
- \* **NextMonthAccountingType** (*string*) –
- \* **PoolNetworkCidr** (*string*) –
- \* **PoolNetworkGatewayIpAddress** (*string*) –
- \* **RemoteAccessVpnGatewayId** (*string*) –
- \* **RemoteAccessVpnGatewayName** (*string*) –
- \* **RemoteAccessVpnGatewayType** (*string*) –
- \* **RemoteUserSet** (*string*) –
- \* **RouteTableAssociationId** (*string*) –
- \* **RouteTableId** (*string*) –
- \* **SslCertificateId** (*string*) –
- \* **Status** (*string*) –
- \* **VersionInformation** (*dict*) –
  - **IsLatest** (*string*) –
  - **Version** (*string*) –
- **RequestId** (*string*) –
- **Return** (*boolean*) –

*computing* / Client / `create_remote_access_vpn_gateway_users`

### `create_remote_access_vpn_gateway_users`

`computing.Client.create_remote_access_vpn_gateway_users` (*\*\*kwargs*)

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.create_remote_access_vpn_gateway_users(  
    RemoteAccessVpnGatewayId='string',  
    RemoteUser=[  
        {  
            'Description': 'string',  
            'Password': 'string',  
            'UserName': 'string'
```

(continues on next page)

(continued from previous page)

```

    },
]
)

```

**Parameters**

- **RemoteAccessVpnGatewayId** (*string*) – [REQUIRED]
- **RemoteUser** (*list*) – [REQUIRED]
  - (*dict*) –
    - \* **Description** (*string*) –
    - \* **Password** (*string*) – [REQUIRED]
    - \* **UserName** (*string*) – [REQUIRED]

**Return type** dict**Returns****Response Syntax**

```

{
    'RemoteAccessVpnGatewayId': 'string',
    'RemoteAccessVpnGatewayName': 'string',
    'RemoteUserSet': [
        {
            'Description': 'string',
            'UserName': 'string'
        },
    ],
    'RequestId': 'string',
    'Return': True|False
}

```

**Response Structure**

- (*dict*) –
  - **RemoteAccessVpnGatewayId** (*string*) –
  - **RemoteAccessVpnGatewayName** (*string*) –
  - **RemoteUserSet** (*list*) –
    - \* (*dict*) –
      - **Description** (*string*) –
      - **UserName** (*string*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / create\_route**create\_route**computing.Client.**create\_route** (\*\*kwargs)See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```

response = client.create_route(
    DestinationCidrBlock='string',
    GatewayId='string',
    InstanceId='string',
    IpAddress='string',
    NetworkId='string',

```

(continues on next page)

(continued from previous page)

```

NetworkInterfaceId='string',
NetworkName='string',
RouteTableId='string',
VpcPeeringConnectionId='string'
)

```

**Parameters**

- **DestinationCidrBlock** (*string*) – [REQUIRED]
- **GatewayId** (*string*) –
- **InstanceId** (*string*) –
- **IpAddress** (*string*) –
- **NetworkId** (*string*) –
- **NetworkInterfaceId** (*string*) –
- **NetworkName** (*string*) –
- **RouteTableId** (*string*) – [REQUIRED]
- **VpcPeeringConnectionId** (*string*) –

**Return type** dict**Returns****Response Syntax**

```

{
    'RequestId': 'string',
    'Return': True|False
}

```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / create\_route\_table**create\_route\_table**`computing.Client.create_route_table(**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```

response = client.create_route_table(
    VpcId='string'
)

```

**Parameters** **VpcId** (*string*) –**Return type** dict**Returns****Response Syntax**

```

{
    'RequestId': 'string',
    'RouteTable': {
        'AssociationSet': 'string',
        'ElasticLoadBalancerAssociationSet': 'string',

```

(continues on next page)

(continued from previous page)

```

        'RouteTableId': 'string',
        'TagSet': [
            {
                'Key': 'string',
                'Value': 'string'
            },
        ]
    }
}

```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **RouteTable** (*dict*) –
    - \* **AssociationSet** (*string*) –
    - \* **ElasticLoadBalancerAssociationSet** (*string*) –
    - \* **RouteTableId** (*string*) –
    - \* **TagSet** (*list*) –
      - (*dict*) –
      - **Key** (*string*) –
      - **Value** (*string*) –

*computing* / Client / create\_security\_group**create\_security\_group**`computing.Client.create_security_group(**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```

response = client.create_security_group(
    GroupDescription='string',
    GroupName='string',
    Placement={
        'AvailabilityZone': 'string'
    }
)

```

**Parameters**

- **GroupDescription** (*string*) –
- **GroupName** (*string*) – [REQUIRED]
- **Placement** (*dict*) –
  - **AvailabilityZone** (*string*) –

**Return type** `dict`**Returns****Response Syntax**

```

{
    'RequestId': 'string',
    'Return': True|False
}

```

**Response Structure**

- (*dict*) –

- **RequestId** (*string*) –
- **Return** (*boolean*) –

*computing* / Client / create\_ssl\_certificate

## create\_ssl\_certificate

`computing.Client.create_ssl_certificate(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.create_ssl_certificate(
    ApproverEmailAddress='string',
    CertAuthority=123,
    CertInfo={
        'CountryName': 'string',
        'EmailAddress': 'string',
        'LocationName': 'string',
        'OrganizationName': 'string',
        'OrganizationUnitName': 'string',
        'StateName': 'string'
    },
    Count=123,
    Fqdn='string',
    FqdnId='string',
    KeyLength=123,
    ValidityTerm=123
)
```

### Parameters

- **ApproverEmailAddress** (*string*) –
- **CertAuthority** (*integer*) –
- **CertInfo** (*dict*) –
  - **CountryName** (*string*) –
  - **EmailAddress** (*string*) –
  - **LocationName** (*string*) –
  - **OrganizationName** (*string*) –
  - **OrganizationUnitName** (*string*) –
  - **StateName** (*string*) –
- **Count** (*integer*) –
- **Fqdn** (*string*) –
- **FqdnId** (*string*) –
- **KeyLength** (*integer*) –
- **ValidityTerm** (*integer*) –

**Return type** dict

### Returns

### Response Syntax

```
{
    'ApproverEmailAddress': 'string',
    'CertAuthority': 'string',
    'CertState': 'string',
    'Fqdn': 'string',
    'FqdnId': 'string',
```

(continues on next page)

(continued from previous page)

```
{
  'RequestId': 'string',
  'ValidityTerm': 123
}
```

**Response Structure**

- (*dict*) –
  - **ApproverEmailAddress** (*string*) –
  - **CertAuthority** (*string*) –
  - **CertState** (*string*) –
  - **Fqdn** (*string*) –
  - **FqdnId** (*string*) –
  - **RequestId** (*string*) –
  - **ValidityTerm** (*integer*) –

*computing* / Client / create\_volume**create\_volume**`computing.Client.create_volume(**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.create_volume(
    AccountingType='1'|'2',
    Description='string',
    DiskType='2'|'3'|'4'|'5'|'6'|'7'|'8'|'9',
    InstanceId='string',
    InstanceUniqueId='string',
    Size=123,
    VolumeId='string'
)
```

**Parameters**

- **AccountingType** (*string*) –
- **Description** (*string*) –
- **DiskType** (*string*) –
- **InstanceId** (*string*) –
- **InstanceUniqueId** (*string*) –
- **Size** (*integer*) – **[REQUIRED]**
- **VolumeId** (*string*) –

**Return type** dict**Returns****Response Syntax**

```
{
  'AccountingType': 'string',
  'AvailabilityZone': 'string',
  'CreateTime': datetime(2015, 1, 1),
  'Description': 'string',
  'DiskType': 'string',
  'RequestId': 'string',
  'Size': 123,
  'SnapshotId': 'string',
```

(continues on next page)

(continued from previous page)

```

    'Status': 'string',
    'VolumeId': 'string',
    'VolumeUniqueId': 'string'
}

```

**Response Structure**

- (dict) –
  - **AccountingType** (string) –
  - **AvailabilityZone** (string) –
  - **CreateTime** (datetime) –
  - **Description** (string) –
  - **DiskType** (string) –
  - **RequestId** (string) –
  - **Size** (integer) –
  - **SnapshotId** (string) –
  - **Status** (string) –
  - **VolumeId** (string) –
  - **VolumeUniqueId** (string) –

*computing* / Client / create\_vpn\_connection

**create\_vpn\_connection**

`computing.Client.create_vpn_connection(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```

response = client.create_vpn_connection(
    Agreement=True|False,
    CustomerGatewayId='string',
    NiftyCustomerGatewayName='string',
    NiftyIpsecConfiguration={
        'DiffieHellmanGroup': 123,
        'EncapsulatingSecurityPayloadLifetime': 123,
        'EncryptionAlgorithm': 'AES128'|'AES256'|'3DES',
        'HashAlgorithm': 'SHA1'|'MD5'|'SHA256'|'SHA384'|'SHA512',
        'InternetKeyExchange': 'IKEv1'|'IKEv2',
        'InternetKeyExchangeLifetime': 123,
        'PreSharedKey': 'string'
    },
    NiftyTunnel={
        'DestinationPort': 'string',
        'Encapsulation': 'IP'|'UDP',
        'Mode': 'Unmanaged'|'Managed',
        'PeerSessionId': 'string',
        'PeerTunnelId': 'string',
        'SessionId': 'string',
        'SourcePort': 'string',
        'TunnelId': 'string',
        'Type': 'L2TPv3'
    },
    NiftyVpnConnectionDescription='string',
    NiftyVpnConnectionMtu='string',
    NiftyVpnGatewayName='string',

```

(continues on next page)



(continued from previous page)

```

Type='IPsec'|'L2TPv3 / IPsec'|'IPsec VTI',
VpnGatewayId='string'
)

```

**Parameters**

- **Agreement** (*boolean*) –
- **CustomerGatewayId** (*string*) –
- **NiftyCustomerGatewayName** (*string*) –
- **NiftyIpsecConfiguration** (*dict*) –
  - **DiffieHellmanGroup** (*integer*) –
  - **EncapsulatingSecurityPayloadLifetime** (*integer*) –
  - **EncryptionAlgorithm** (*string*) –
  - **HashAlgorithm** (*string*) –
  - **InternetKeyExchange** (*string*) –
  - **InternetKeyExchangeLifetime** (*integer*) –
  - **PreSharedKey** (*string*) –
- **NiftyTunnel** (*dict*) –
  - **DestinationPort** (*string*) –
  - **Encapsulation** (*string*) –
  - **Mode** (*string*) –
  - **PeerSessionId** (*string*) –
  - **PeerTunnelId** (*string*) –
  - **SessionId** (*string*) –
  - **SourcePort** (*string*) –
  - **TunnelId** (*string*) –
  - **Type** (*string*) –
- **NiftyVpnConnectionDescription** (*string*) –
- **NiftyVpnConnectionMtu** (*string*) –
- **NiftyVpnGatewayName** (*string*) –
- **Type** (*string*) – [REQUIRED]
- **VpnGatewayId** (*string*) –

**Return type** dict**Returns****Response Syntax**

```

{
  'RequestId': 'string',
  'VpnConnection': {
    'CreatedTime': datetime(2015, 1, 1),
    'CustomerGatewayId': 'string',
    'NiftyCustomerGatewayName': 'string',
    'NiftyIpsecConfiguration': {
      'DiffieHellmanGroup': 123,
      'EncapsulatingSecurityPayloadLifetime': 123,
      'EncryptionAlgorithm': 'string',
      'HashingAlgorithm': 'string',
      'InternetKeyExchange': 'string',
      'InternetKeyExchangeLifetime': 123,
      'Mtu': 'string',
      'PreSharedKey': 'string'
    },
    'NiftyTunnel': {
      'DestinationPort': 'string',

```

(continues on next page)

(continued from previous page)

```

        'Encapsulation': 'string',
        'Mode': 'string',
        'PeerSessionId': 'string',
        'PeerTunnelId': 'string',
        'SessionId': 'string',
        'SourcePort': 'string',
        'TunnelId': 'string',
        'Type': 'string'
    },
    'NiftyVpnConnectionDescription': 'string',
    'NiftyVpnGatewayName': 'string',
    'State': 'string',
    'TagSet': [
        {
            'Key': 'string',
            'Value': 'string'
        },
    ],
    'Type': 'string',
    'VgwTelemetry': [
        {
            'AcceptedRouteCount': 123,
            'LastStatusChange': datetime(2015, 1, 1),
            'OutsideIpAddress': 'string',
            'Status': 'string',
            'StatusMessage': 'string'
        },
    ],
    'VpnConnectionId': 'string',
    'VpnGatewayId': 'string'
}

```

### Response Structure

- (dict) –
  - **RequestId** (string) –
  - **VpnConnection** (dict) –
    - \* **CreatedTime** (datetime) –
    - \* **CustomerGatewayId** (string) –
    - \* **NiftyCustomerGatewayName** (string) –
    - \* **NiftyIpsecConfiguration** (dict) –
      - **DiffieHellmanGroup** (integer) –
      - **EncapsulatingSecurityPayloadLifetime** (integer) –
      - **EncryptionAlgorithm** (string) –
      - **HashingAlgorithm** (string) –
      - **InternetKeyExchange** (string) –
      - **InternetKeyExchangeLifetime** (integer) –
      - **Mtu** (string) –
      - **PreSharedKey** (string) –
    - \* **NiftyTunnel** (dict) –
      - **DestinationPort** (string) –
      - **Encapsulation** (string) –
      - **Mode** (string) –
      - **PeerSessionId** (string) –
      - **PeerTunnelId** (string) –

- **SessionId** (*string*) –
- **SourcePort** (*string*) –
- **TunnelId** (*string*) –
- **Type** (*string*) –
- \* **NiftyVpnConnectionDescription** (*string*) –
- \* **NiftyVpnGatewayName** (*string*) –
- \* **State** (*string*) –
- \* **TagSet** (*list*) –
  - (*dict*) –
  - **Key** (*string*) –
  - **Value** (*string*) –
- \* **Type** (*string*) –
- \* **VgwTelemetry** (*list*) –
  - (*dict*) –
  - **AcceptedRouteCount** (*integer*) –
  - **LastStatusChange** (*datetime*) –
  - **OutsideIpAddress** (*string*) –
  - **Status** (*string*) –
  - **StatusMessage** (*string*) –
- \* **VpnConnectionId** (*string*) –
- \* **VpnGatewayId** (*string*) –

*computing* / Client / create\_vpn\_gateway

## create\_vpn\_gateway

`computing.Client.create_vpn_gateway(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.create_vpn_gateway(
    AccountingType='1'|'2',
    NiftyNetwork={
        'IpAddress': 'string',
        'NetworkId': 'string',
        'NetworkName': 'string'
    },
    NiftyRedundancy=True|False,
    NiftyVpnGatewayDescription='string',
    NiftyVpnGatewayName='string',
    NiftyVpnGatewayType='small'|'medium'|'large',
    Placement={
        'AvailabilityZone': 'string'
    },
    SecurityGroup=[
        'string',
    ],
    Type='string'
)
```

### Parameters

- **AccountingType** (*string*) –
- **NiftyNetwork** (*dict*) –
  - **IpAddress** (*string*) –
  - **NetworkId** (*string*) –

- **NetworkName** (*string*) –
- **NiftyRedundancy** (*boolean*) –
- **NiftyVpnGatewayDescription** (*string*) –
- **NiftyVpnGatewayName** (*string*) –
- **NiftyVpnGatewayType** (*string*) –
- **Placement** (*dict*) –
  - **AvailabilityZone** (*string*) –
- **SecurityGroup** (*list*) –
  - (*string*) –
- **Type** (*string*) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'RequestId': 'string',
  'VpnGateway': {
    'AccountingType': 'string',
    'AvailabilityZone': 'string',
    'BackupInformation': {
      'IsBackup': True|False
    },
    'GroupSet': [
      {
        'GroupId': 'string'
      },
    ],
    'NetworkInterfaceSet': [
      {
        'IpAddress': 'string',
        'NetworkId': 'string',
        'NetworkName': 'string'
      },
    ],
    'NextMonthAccountingType': 'string',
    'NiftyRedundancy': True|False,
    'NiftyVpnGatewayDescription': 'string',
    'NiftyVpnGatewayName': 'string',
    'NiftyVpnGatewayType': 'string',
    'State': 'string',
    'VersionInformation': {
      'IsLatest': True|False,
      'Version': 'string'
    },
    'VpnGatewayId': 'string'
  }
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **VpnGateway** (*dict*) –
    - \* **AccountingType** (*string*) –
    - \* **AvailabilityZone** (*string*) –
    - \* **BackupInformation** (*dict*) –
      - **IsBackup** (*boolean*) –

- \* **GroupSet** (*list*) –
  - (*dict*) –
  - **GroupId** (*string*) –
- \* **NetworkInterfaceSet** (*list*) –
  - (*dict*) –
  - **IpAddress** (*string*) –
  - **NetworkId** (*string*) –
  - **NetworkName** (*string*) –
- \* **NextMonthAccountingType** (*string*) –
- \* **NiftyRedundancy** (*boolean*) –
- \* **NiftyVpnGatewayDescription** (*string*) –
- \* **NiftyVpnGatewayName** (*string*) –
- \* **NiftyVpnGatewayType** (*string*) –
- \* **State** (*string*) –
- \* **VersionInformation** (*dict*) –
  - **IsLatest** (*boolean*) –
  - **Version** (*string*) –
- \* **VpnGatewayId** (*string*) –

*computing* / Client / delete\_customer\_gateway

## delete\_customer\_gateway

`computing.Client.delete_customer_gateway(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_customer_gateway(
    CustomerGatewayId='string',
    NiftyCustomerGatewayName='string'
)
```

### Parameters

- **CustomerGatewayId** (*string*) –
- **NiftyCustomerGatewayName** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / delete\_dhcp\_options

## delete\_dhcp\_options

`computing.Client.delete_dhcp_options(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_dhcp_options(  
    DhcpOptionsId='string'  
)
```

**Parameters** `DhcpOptionsId` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'RequestId': 'string',  
    'Return': True|False  
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / delete\_image

## delete\_image

`computing.Client.delete_image(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_image(  
    ImageId='string'  
)
```

**Parameters** `ImageId` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'RequestId': 'string'  
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –

*computing* / Client / delete\_instance\_backup\_rule

## delete\_instance\_backup\_rule

`computing.Client.delete_instance_backup_rule(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_instance_backup_rule(
    InstanceBackupRuleId='string'
)
```

**Parameters** `InstanceBackupRuleId` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / delete\_iso\_image

## delete\_iso\_image

`computing.Client.delete_iso_image(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_iso_image(
    IsoImageId='string'
)
```

**Parameters** `IsoImageId` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / delete\_key\_pair

## delete\_key\_pair

`computing.Client.delete_key_pair(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_key_pair(  
    KeyName='string'  
)
```

**Parameters** `KeyName` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'RequestId': 'string',  
    'Return': True|False  
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / delete\_load\_balancer

## delete\_load\_balancer

`computing.Client.delete_load_balancer(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_load_balancer(  
    InstancePort=123,  
    LoadBalancerName='string',  
    LoadBalancerPort=123  
)
```

### Parameters

- **InstancePort** (*integer*) – [REQUIRED]
- **LoadBalancerName** (*string*) – [REQUIRED]
- **LoadBalancerPort** (*integer*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'DeleteLoadBalancerResult': 'string',  
    'ResponseMetadata': {  
        'RequestId': 'string'  
    }  
}
```



**Response Structure**

- *(dict)* –
  - **DeleteLoadBalancerResult** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*computing* / Client / delete\_multi\_ip\_address\_group

**delete\_multi\_ip\_address\_group**

`computing.Client.delete_multi_ip_address_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.delete_multi_ip_address_group(
    MultiIpAddressGroupId='string'
)
```

**Parameters** `MultiIpAddressGroupId` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

**Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- *(dict)* –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / delete\_network\_interface

**delete\_network\_interface**

`computing.Client.delete_network_interface(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.delete_network_interface(
    NetworkInterfaceId='string'
)
```

**Parameters** `NetworkInterfaceId` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

**Response Syntax**

```
{
    'RequestId': 'string',
```

(continues on next page)

(continued from previous page)

```
'Return': True|False
}
```

**Response Structure**

- (dict) –
  - **RequestId** (string) –
  - **Return** (boolean) –

*computing* / Client / delete\_remote\_access\_vpn\_gateway**delete\_remote\_access\_vpn\_gateway**`computing.Client.delete_remote_access_vpn_gateway(**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.delete_remote_access_vpn_gateway(
    RemoteAccessVpnGatewayId='string'
)
```

**Parameters** `RemoteAccessVpnGatewayId` (string) – [REQUIRED]**Return type** dict**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (dict) –
  - **RequestId** (string) –
  - **Return** (boolean) –

*computing* / Client / delete\_remote\_access\_vpn\_gateway\_connections**delete\_remote\_access\_vpn\_gateway\_connections**`computing.Client.delete_remote_access_vpn_gateway_connections(**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.delete_remote_access_vpn_gateway_connections(
    Connection=[
        {
            'ConnectionId': 'string'
        },
    ],
    RemoteAccessVpnGatewayId='string'
)
```

**Parameters**

- **Connection** (*list*) – [REQUIRED]
  - (*dict*) –
    - \* **ConnectionId** (*string*) – [REQUIRED]
- **RemoteAccessVpnGatewayId** (*string*) – [REQUIRED]

Return type `dict`

Returns

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / `delete_remote_access_vpn_gateway_users`

### delete\_remote\_access\_vpn\_gateway\_users

`computing.Client.delete_remote_access_vpn_gateway_users (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.delete_remote_access_vpn_gateway_users(
    RemoteAccessVpnGatewayId='string',
    RemoteUser=[
        {
            'UserName': 'string'
        },
    ],
)
```

#### Parameters

- **RemoteAccessVpnGatewayId** (*string*) – [REQUIRED]
- **RemoteUser** (*list*) – [REQUIRED]
  - (*dict*) –
    - \* **UserName** (*string*) – [REQUIRED]

Return type `dict`

Returns

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / `delete_route`

## delete\_route

`computing.Client.delete_route(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_route(  
    DestinationCidrBlock='string',  
    RouteTableId='string'  
)
```

### Parameters

- **DestinationCidrBlock** (*string*) – [REQUIRED]
- **RouteTableId** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'RequestId': 'string',  
    'Return': True|False  
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / delete\_route\_table

## delete\_route\_table

`computing.Client.delete_route_table(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_route_table(  
    RouteTableId='string'  
)
```

**Parameters** **RouteTableId** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'RequestId': 'string',  
    'Return': True|False  
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / delete\_security\_group

## delete\_security\_group

`computing.Client.delete_security_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_security_group(
    GroupName='string'
)
```

**Parameters** `GroupName` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / delete\_ssl\_certificate

## delete\_ssl\_certificate

`computing.Client.delete_ssl_certificate(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_ssl_certificate(
    FqdnId='string'
)
```

**Parameters** `FqdnId` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / delete\_volume

## delete\_volume

`computing.Client.delete_volume(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_volume(  
    VolumeId='string'  
)
```

**Parameters** `VolumeId` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'RequestId': 'string',  
    'Return': True|False  
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / delete\_vpn\_connection

## delete\_vpn\_connection

`computing.Client.delete_vpn_connection(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_vpn_connection(  
    Agreement=True|False,  
    VpnConnectionId='string'  
)
```

### Parameters

- **Agreement** (*boolean*) –
- **VpnConnectionId** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'RequestId': 'string',  
    'Return': True|False  
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / delete\_vpn\_gateway

## delete\_vpn\_gateway

`computing.Client.delete_vpn_gateway(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_vpn_gateway(
    NiftyVpnGatewayName='string',
    VpnGatewayId='string'
)
```

### Parameters

- **NiftyVpnGatewayName** (*string*) –
- **VpnGatewayId** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / deregister\_instances\_from\_load\_balancer

## deregister\_instances\_from\_load\_balancer

`computing.Client.deregister_instances_from_load_balancer(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.deregister_instances_from_load_balancer(
    InstancePort=123,
    Instances=[
        {
            'InstanceId': 'string'
        },
    ],
    LoadBalancerName='string',
    LoadBalancerPort=123
)
```

### Parameters

- **InstancePort** (*integer*) – [REQUIRED]
- **Instances** (*list*) – [REQUIRED]
  - (*dict*) –
    - \* **InstanceId** (*string*) – [REQUIRED]
- **LoadBalancerName** (*string*) – [REQUIRED]
- **LoadBalancerPort** (*integer*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'DeregisterInstancesFromLoadBalancerResult': {
    'Instances': [
      {
        'InstanceId': 'string',
        'InstanceUniqueId': 'string'
      },
    ]
  },
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

#### Response Structure

- (*dict*) –
  - **DeregisterInstancesFromLoadBalancerResult** (*dict*) –
    - \* **Instances** (*list*) –
      - (*dict*) –
      - **InstanceId** (*string*) –
      - **InstanceUniqueId** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*computing* / Client / deregister\_instances\_from\_security\_group

### deregister\_instances\_from\_security\_group

`computing.Client.deregister_instances_from_security_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.deregister_instances_from_security_group(
    GroupName='string',
    InstanceId=[
        'string',
    ]
)
```

#### Parameters

- **GroupName** (*string*) – [REQUIRED]
- **InstanceId** (*list*) – [REQUIRED]
  - (*string*) –

**Return type** dict



## Returns

### Response Syntax

```
{
  'InstancesSet': [
    {
      'InstanceId': 'string'
    },
  ],
  'RequestId': 'string'
}
```

### Response Structure

- (dict) –
  - **InstancesSet** (list) –
    - \* (dict) –
      - **InstanceId** (string) –
  - **RequestId** (string) –

*computing* / Client / describe\_addresses

## describe\_addresses

`computing.Client.describe_addresses (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_addresses(
    AllocationId=[
        'string',
    ],
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'string'
        },
    ],
    PrivateIpAddress=[
        'string',
    ],
    PublicIp=[
        'string',
    ]
)
```

### Parameters

- **AllocationId** (list) –
  - (string) –
- **Filter** (list) –
  - (dict) –
    - \* **ListOfRequestValue** (list) –
      - (string) –
    - \* **Name** (string) –

- **PrivateIpAddress** (*list*) –
  - (*string*) –
- **PublicIp** (*list*) –
  - (*string*) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'AddressesSet': [
    {
      'AvailabilityZone': 'string',
      'Description': 'string',
      'InstanceId': 'string',
      'InstanceUniqueId': 'string',
      'PrivateIpAddress': 'string',
      'PublicIp': 'string'
    },
  ],
  'RequestId': 'string'
}
```

#### Response Structure

- (*dict*) –
  - **AddressesSet** (*list*) –
    - \* (*dict*) –
      - **AvailabilityZone** (*string*) –
      - **Description** (*string*) –
      - **InstanceId** (*string*) –
      - **InstanceUniqueId** (*string*) –
      - **PrivateIpAddress** (*string*) –
      - **PublicIp** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / describe\_associated\_users

### describe\_associated\_users

`computing.Client.describe_associated_users (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.describe_associated_users(
    FunctionName='LB'
)
```

**Parameters** **FunctionName** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'DescribeAssociatedUsersResult': {
    'Users': [
```

(continues on next page)

(continued from previous page)

```

        {
            'UserId': 'string'
        },
    ],
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}

```

**Response Structure**

- (*dict*) –
  - **DescribeAssociatedUsersResult** (*dict*) –
    - \* **Users** (*list*) –
      - (*dict*) –
        - **UserId** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*computing* / Client / describe\_availability\_zones

**describe\_availability\_zones**

`computing.Client.describe_availability_zones(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```

response = client.describe_availability_zones(
    ZoneName=[
        'string',
    ]
)

```

**Parameters** **ZoneName** (*list*) –

- (*string*) –

**Return type** `dict`

**Returns**

**Response Syntax**

```

{
    'AvailabilityZoneInfo': [
        {
            'IsDefault': True|False,
            'MessageSet': 'string',
            'RegionName': 'string',
            'SecurityGroupSupported': True|False,
            'ZoneName': 'string',
            'ZoneState': 'string'
        },
    ],
    'RequestId': 'string'
}

```

**Response Structure**

- *(dict)* –
  - **AvailabilityZoneInfo** (*list*) –
    - \* *(dict)* –
      - **IsDefault** (*boolean*) –
      - **MessageSet** (*string*) –
      - **RegionName** (*string*) –
      - **SecurityGroupSupported** (*boolean*) –
      - **ZoneName** (*string*) –
      - **ZoneState** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / describe\_customer\_gateways

## describe\_customer\_gateways

`computing.Client.describe_customer_gateways(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_customer_gateways(
    CustomerGatewayId=[
        'string',
    ],
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'customer-gateway-id'|'nifty-customer-gateway-name'|'ip-
↪address'|'state'|'nifty-customer-gateway-description'
        },
    ],
    NiftyCustomerGatewayName=[
        'string',
    ]
)
```

### Parameters

- **CustomerGatewayId** (*list*) –
  - (*string*) –
- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **NiftyCustomerGatewayName** (*list*) –
  - (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
    'CustomerGatewaySet': [
        {
```

(continues on next page)

(continued from previous page)

```

        'CreatedTime': datetime(2015, 1, 1),
        'CustomerGatewayId': 'string',
        'IpAddress': 'string',
        'NiftyCustomerGatewayDescription': 'string',
        'NiftyCustomerGatewayName': 'string',
        'NiftyLanSideCidrBlock': 'string',
        'NiftyLanSideIpAddress': 'string',
        'State': 'string',
        'TagSet': [
            {
                'Key': 'string',
                'Value': 'string'
            },
        ],
    ],
    'RequestId': 'string'
}

```

**Response Structure**

- (dict) –
  - **CustomerGatewaySet** (list) –
    - \* (dict) –
      - **CreatedTime** (datetime) –
      - **CustomerGatewayId** (string) –
      - **IpAddress** (string) –
      - **NiftyCustomerGatewayDescription** (string) –
      - **NiftyCustomerGatewayName** (string) –
      - **NiftyLanSideCidrBlock** (string) –
      - **NiftyLanSideIpAddress** (string) –
      - **State** (string) –
      - **TagSet** (list) –
      - (dict) –
        - **Key** (string) –
        - **Value** (string) –
  - **RequestId** (string) –

*computing* / Client / describe\_dhcp\_options**describe\_dhcp\_options**`computing.Client.describe_dhcp_options(**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```

response = client.describe_dhcp_options(
    DhcpOptionsId=[
        'string',
    ],
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
        },
    ],
)

```

(continues on next page)

(continued from previous page)

```

        'Name': 'dhcp-options-id'|'key'|'value'
    },
]
)

```

**Parameters**

- **DhcpOptionsId** (*list*) –
  - (*string*) –
- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –

**Return type** dict**Returns****Response Syntax**

```

{
    'DhcpOptionsSet': [
        {
            'DhcpConfigurationSet': [
                {
                    'Key': 'string',
                    'ValueSet': [
                        {
                            'Value': 'string'
                        },
                    ],
                },
            ],
            'DhcpOptionsId': 'string'
        },
    ],
    'RequestId': 'string'
}

```

**Response Structure**

- (*dict*) –
  - **DhcpOptionsSet** (*list*) –
    - \* (*dict*) –
      - **DhcpConfigurationSet** (*list*) –
        - (*dict*) –
        - **Key** (*string*) –
        - **ValueSet** (*list*) –
          - (*dict*) –
          - **Value** (*string*) –
        - **DhcpOptionsId** (*string*) –
    - **RequestId** (*string*) –

*computing* / Client / describe\_images

## describe\_images

computing.Client.**describe\_images** (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```

response = client.describe_images(
    ExecutableBy=[
        'string',
    ],
    ImageId=[
        'string',
    ],
    ImageName=[
        'string',
    ],
    Owner=[
        'string',
    ]
)

```

### Parameters

- **ExecutableBy** (*list*) –
  - (*string*) –
- **ImageId** (*list*) –
  - (*string*) –
- **ImageName** (*list*) –
  - (*string*) –
- **Owner** (*list*) –
  - (*string*) –

**Return type** dict

### Returns

### Response Syntax

```

{
    'ImagesSet': [
        {
            'Architecture': 'string',
            'BlockDeviceMapping': [
                {
                    'DeviceName': 'string',
                    'Ebs': {
                        'DeleteOnTermination': 'string',
                        'DiskType': 'string',
                        'SnapshotId': 'string',
                        'VolumeSize': 123
                    }
                }
            ],
            'Description': 'string',
            'DetailDescription': 'string',
            'ImageId': 'string',
            'ImageLocation': 'string',
            'ImageOwnerAlias': 'string',
            'ImageOwnerId': 'string',

```

(continues on next page)

(continued from previous page)

```

        'ImageState': 'string',
        'ImageType': 'string',
        'IsPublic': True|False,
        'KernelId': 'string',
        'LaunchTime': datetime(2015, 1, 1),
        'Name': 'string',
        'NiftyContactUrl': 'string',
        'NiftyDistributionIds': [
            {
                'DistributionId': 'string'
            },
        ],
        'NiftyImageSize': 'string',
        'NiftyIsAllowedDistribution': True|False,
        'Placement': {
            'AvailabilityZone': 'string',
            'RegionName': 'string'
        },
        'Platform': 'string',
        'ProductCodes': [
            {
                'ProductCode': 'string'
            },
        ],
        'RamdiskId': 'string',
        'Redistributable': True|False,
        'RootDeviceName': 'string',
        'RootDeviceType': 'string',
        'StateReason': {
            'Code': 'string',
            'Message': 'string'
        }
    },
    'RequestId': 'string'
}

```

**Response Structure**

- (dict) –
  - **ImagesSet** (list) –
    - \* (dict) –
      - **Architecture** (string) –
      - **BlockDeviceMapping** (list) –
      - (dict) –
      - **DeviceName** (string) –
      - **Ebs** (dict) –
      - **DeleteOnTermination** (string) –
      - **DiskType** (string) –
      - **SnapshotId** (string) –
      - **VolumeSize** (integer) –
      - **Description** (string) –
      - **DetailDescription** (string) –
      - **ImageId** (string) –
      - **ImageLocation** (string) –
      - **ImageOwnerAlias** (string) –
      - **ImageOwnerId** (string) –



- **ImageState** (*string*) –
- **ImageType** (*string*) –
- **IsPublic** (*boolean*) –
- **KernelId** (*string*) –
- **LaunchTime** (*datetime*) –
- **Name** (*string*) –
- **NiftyContactUrl** (*string*) –
- **NiftyDistributionIds** (*list*) –
- (*dict*) –
- **DistributionId** (*string*) –
- **NiftyImageSize** (*string*) –
- **NiftyIsAllowedDistribution** (*boolean*) –
- **Placement** (*dict*) –
- **AvailabilityZone** (*string*) –
- **RegionName** (*string*) –
- **Platform** (*string*) –
- **ProductCodes** (*list*) –
- (*dict*) –
- **ProductCode** (*string*) –
- **RamdiskId** (*string*) –
- **Redistributable** (*boolean*) –
- **RootDeviceName** (*string*) –
- **RootDeviceType** (*string*) –
- **StateReason** (*dict*) –
- **Code** (*string*) –
- **Message** (*string*) –
- **RequestId** (*string*) –

*computing* / Client / describe\_instance\_attribute

## describe\_instance\_attribute

`computing.Client.describe_instance_attribute(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_instance_attribute(
    Attribute='instanceType'|'disableApiTermination'|'blockDeviceMapping'|
    ↳ 'accountingType'|'nextMonthAccountingType'|'loadbalancing'|'copyInfo'|
    ↳ 'autoscaling'|'ipType'|'niftyPrivateIpType'|'groupId'|'description'|
    ↳ 'networkInterfaceSet'|'elasticloadbalancing',
    InstanceId='string'
)
```

### Parameters

- **Attribute** (*string*) –
- **InstanceId** (*string*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```
{
    'AccountingType': {
```

(continues on next page)

(continued from previous page)

```

        'Value': 'string'
    },
    'Autoscaling': {
        'AutoScalingGroupName': 'string',
        'ExpireTime': datetime(2015, 1, 1)
    },
    'BlockDeviceMapping': [
        {
            'DeviceName': 'string',
            'Ebs': {
                'AttachTime': 'string',
                'DeleteOnTermination': 'string',
                'Status': 'string',
                'VolumeId': 'string',
                'VolumeUniqueId': 'string'
            }
        },
    ],
    'CopyInfo': {
        'Value': 'string'
    },
    'Description': {
        'Value': 'string'
    },
    'DisableApiTermination': {
        'Value': True|False
    },
    'GroupId': {
        'Value': 'string'
    },
    'InstanceId': 'string',
    'InstanceType': {
        'Value': 'string'
    },
    'InstanceUniqueId': 'string',
    'IpType': {
        'Value': 'string'
    },
    'Loadbalancing': [
        {
            'InstancePort': 123,
            'LoadBalancerName': 'string',
            'LoadBalancerPort': 123,
            'State': 'string'
        },
    ],
    'NetworkInterfaceSet': [
        {
            'Association': {
                'IpOwnerId': 'string',
                'PublicDnsName': 'string',
                'PublicIp': 'string',
                'PublicIpV6': 'string'
            },
            'Attachment': {
                'AttachTime': 'string',
                'AttachmentId': 'string',

```

(continues on next page)

(continued from previous page)

```

        'DeleteOnTermination': 'string',
        'DeviceIndex': 'string',
        'Status': 'string'
    },
    'Description': 'string',
    'GroupSet': 'string',
    'MacAddress': 'string',
    'MultiIpAddressesSet': [
        {
            'IpAddress': 'string'
        },
    ],
    'NetworkInterfaceId': 'string',
    'NiftyNetworkId': 'string',
    'NiftyNetworkName': 'string',
    'OwnerId': 'string',
    'PrivateDnsName': 'string',
    'PrivateIpAddress': 'string',
    'PrivateIpAddressV6': 'string',
    'PrivateIpAddressesSet': 'string',
    'SourceDestCheck': 'string',
    'Status': 'string',
    'SubnetId': 'string',
    'VpcId': 'string'
},
],
'NextMonthAccountingType': {
    'Value': 'string'
},
'NiftyElasticLoadBalancing': [
    {
        'ElasticLoadBalancerId': 'string',
        'ElasticLoadBalancerName': 'string',
        'ElasticLoadBalancerPort': 123,
        'InstancePort': 123,
        'Protocol': 'string'
    },
],
'NiftyPrivateIpType': {
    'Value': 'string'
},
'RequestId': 'string'
}

```

### Response Structure

- *(dict)* –
  - **AccountingType** (*dict*) –
    - \* **Value** (*string*) –
  - **Autoscaling** (*dict*) –
    - \* **AutoScalingGroupName** (*string*) –
    - \* **ExpireTime** (*datetime*) –
  - **BlockDeviceMapping** (*list*) –
    - \* (*dict*) –
      - **DeviceName** (*string*) –
      - **Ebs** (*dict*) –
      - **AttachTime** (*string*) –

- **DeleteOnTermination** (*string*) –
  - **Status** (*string*) –
  - **VolumeId** (*string*) –
  - **VolumeUniqueId** (*string*) –
- **CopyInfo** (*dict*) –
  - \* **Value** (*string*) –
- **Description** (*dict*) –
  - \* **Value** (*string*) –
- **DisableApiTermination** (*dict*) –
  - \* **Value** (*boolean*) –
- **GroupId** (*dict*) –
  - \* **Value** (*string*) –
- **InstanceId** (*string*) –
- **InstanceType** (*dict*) –
  - \* **Value** (*string*) –
- **InstanceUniqueId** (*string*) –
- **IpType** (*dict*) –
  - \* **Value** (*string*) –
- **Loadbalancing** (*list*) –
  - \* (*dict*) –
    - **InstancePort** (*integer*) –
    - **LoadBalancerName** (*string*) –
    - **LoadBalancerPort** (*integer*) –
    - **State** (*string*) –
- **NetworkInterfaceSet** (*list*) –
  - \* (*dict*) –
    - **Association** (*dict*) –
    - **IpOwnerId** (*string*) –
    - **PublicDnsName** (*string*) –
    - **PublicIp** (*string*) –
    - **PublicIpV6** (*string*) –
    - **Attachment** (*dict*) –
    - **AttachTime** (*string*) –
    - **AttachmentId** (*string*) –
    - **DeleteOnTermination** (*string*) –
    - **DeviceIndex** (*string*) –
    - **Status** (*string*) –
    - **Description** (*string*) –
    - **GroupSet** (*string*) –
    - **MacAddress** (*string*) –
    - **MultiIpAddressesSet** (*list*) –
    - (*dict*) –
      - **IpAddress** (*string*) –
      - **NetworkInterfaceId** (*string*) –
      - **NiftyNetworkId** (*string*) –
      - **NiftyNetworkName** (*string*) –
      - **OwnerId** (*string*) –
      - **PrivateDnsName** (*string*) –
      - **PrivateIpAddress** (*string*) –
      - **PrivateIpAddressV6** (*string*) –
      - **PrivateIpAddressesSet** (*string*) –
      - **SourceDestCheck** (*string*) –
      - **Status** (*string*) –
      - **SubnetId** (*string*) –

- **VpcId** (*string*) –
- **NextMonthAccountingType** (*dict*) –
  - \* **Value** (*string*) –
- **NiftyElasticLoadBalancing** (*list*) –
  - \* (*dict*) –
    - **ElasticLoadBalancerId** (*string*) –
    - **ElasticLoadBalancerName** (*string*) –
    - **ElasticLoadBalancerPort** (*integer*) –
    - **InstancePort** (*integer*) –
    - **Protocol** (*string*) –
- **NiftyPrivateIpType** (*dict*) –
  - \* **Value** (*string*) –
- **RequestId** (*string*) –

*computing* / Client / describe\_instance\_backup\_rule\_activities

## describe\_instance\_backup\_rule\_activities

`computing.Client.describe_instance_backup_rule_activities(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_instance_backup_rule_activities(
    Duration=123,
    EndDateTime='string',
    InstanceBackupRuleId='string',
    MaxRecords=123
)
```

### Parameters

- **Duration** (*integer*) –
- **EndDateTime** (*string*) –
- **InstanceBackupRuleId** (*string*) – [REQUIRED]
- **MaxRecords** (*integer*) –

**Return type** dict

### Returns

### Response Syntax

```
{
  'ActivitiesSet': [
    {
      'BackupInstanceUniqueId': 'string',
      'Detail': 'string',
      'EndDateTime': 'string',
      'InstanceUniqueId': 'string',
      'Operation': 'string',
      'StartDateTime': 'string',
      'Status': 'string'
    },
  ],
  'InstanceBackupRuleId': 'string',
  'InstanceBackupRuleName': 'string',
  'RequestId': 'string'
}
```

**Response Structure**

- *(dict)* –
  - **ActivitiesSet** (*list*) –
    - \* *(dict)* –
      - **BackupInstanceUniqueId** (*string*) –
      - **Detail** (*string*) –
      - **EndTime** (*string*) –
      - **InstanceUniqueId** (*string*) –
      - **Operation** (*string*) –
      - **StartTime** (*string*) –
      - **Status** (*string*) –
  - **InstanceBackupRuleId** (*string*) –
  - **InstanceBackupRuleName** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / describe\_instance\_backup\_rules

**describe\_instance\_backup\_rules**

`computing.Client.describe_instance_backup_rules(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.describe_instance_backup_rules(
    InstanceBackupRuleId=[
        'string',
    ]
)
```

**Parameters** **InstanceBackupRuleId** (*list*) –

- (*string*) –

**Return type** dict

**Returns**

**Response Syntax**

```
{
    'InstanceBackupRulesSet': [
        {
            'AvailabilityZone': 'string',
            'BackupInstanceMaxCount': 123,
            'Description': 'string',
            'InstanceBackupRuleId': 'string',
            'InstanceBackupRuleName': 'string',
            'InstancesSet': [
                {
                    'BackupInstancesSet': [
                        {
                            'BackupInstanceCreateTime': 'string',
                            'BackupInstanceUniqueId': 'string',
                            'Status': 'string'
                        },
                    ],
                    'InstanceId': 'string',
                    'InstanceUniqueId': 'string'
                },
            ],
        },
    ],
}
```

(continues on next page)

(continued from previous page)

```

        },
    ],
    'RegionName': 'string',
    'Status': 'string',
    'TimeSlotId': 'string'
},
],
'RequestId': 'string'
}

```

**Response Structure**

- (dict) –
  - **InstanceBackupRulesSet** (list) –
    - \* (dict) –
      - **AvailabilityZone** (string) –
      - **BackupInstanceMaxCount** (integer) –
      - **Description** (string) –
      - **InstanceBackupRuleId** (string) –
      - **InstanceBackupRuleName** (string) –
      - **InstancesSet** (list) –
      - (dict) –
      - **BackupInstancesSet** (list) –
      - (dict) –
      - **BackupInstanceCreateTime** (string) –
      - **BackupInstanceUniqueId** (string) –
      - **Status** (string) –
      - **InstanceId** (string) –
      - **InstanceUniqueId** (string) –
      - **RegionName** (string) –
      - **Status** (string) –
      - **TimeSlotId** (string) –
  - **RequestId** (string) –

*computing* / Client / describe\_instance\_health

**describe\_instance\_health**

`computing.Client.describe_instance_health(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```

response = client.describe_instance_health(
    InstancePort=123,
    Instances=[
        {
            'InstanceId': 'string'
        },
    ],
    LoadBalancerName='string',
    LoadBalancerPort=123
)

```

**Parameters**

- **InstancePort** (integer) – [REQUIRED]

- **Instances** (*list*) –
  - (*dict*) –
    - \* **InstanceId** (*string*) –
- **LoadBalancerName** (*string*) – [REQUIRED]
- **LoadBalancerPort** (*integer*) – [REQUIRED]

Return type `dict`

Returns

#### Response Syntax

```
{
  'DescribeInstanceHealthResult': {
    'InstanceStates': [
      {
        'Description': 'string',
        'InstanceId': 'string',
        'InstanceUniqueId': 'string',
        'ReasonCode': 'string',
        'State': 'string'
      },
    ]
  },
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

#### Response Structure

- (*dict*) –
  - **DescribeInstanceHealthResult** (*dict*) –
    - \* **InstanceStates** (*list*) –
      - (*dict*) –
        - **Description** (*string*) –
        - **InstanceId** (*string*) –
        - **InstanceUniqueId** (*string*) –
        - **ReasonCode** (*string*) –
        - **State** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*computing* / Client / `describe_instances`

### `describe_instances`

`computing.Client.describe_instances` (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.describe_instances(
    InstanceId=[
        'string',
    ],
    Tenancy=[
        'string',
    ]
)
```



**Parameters**

- **InstanceId** (*list*) –  
– (*string*) –
- **Tenancy** (*list*) –  
– (*string*) –

**Return type** dict**Returns****Response Syntax**

```
{
  'RequestId': 'string',
  'ReservationSet': [
    {
      'GroupSet': [
        {
          'GroupId': 'string'
        }
      ],
      'InstancesSet': [
        {
          'AccountingType': 'string',
          'AmiLaunchIndex': 'string',
          'Architecture': 'string',
          'Autoscaling': {
            'AutoScalingGroupName': 'string',
            'ExpireTime': datetime(2015, 1, 1)
          },
          'BlockDeviceMapping': [
            {
              'DeviceName': 'string',
              'Ebs': {
                'AttachTime': 'string',
                'DeleteOnTermination': 'string',
                'Status': 'string',
                'VolumeId': 'string',
                'VolumeUniqueId': 'string'
              }
            }
          ],
          'CopyInfo': 'string',
          'Description': 'string',
          'DnsName': 'string',
          'HotAdd': 'string',
          'ImageId': 'string',
          'ImageName': 'string',
          'InstanceBackupRule': {
            'InstanceBackupRuleId': 'string',
            'InstanceBackupRuleName': 'string'
          },
          'InstanceId': 'string',
          'InstanceLifecycle': 'string',
          'InstanceState': {
            'Code': 123,
            'Name': 'string'
          },
          'InstanceType': 'string',
          'InstanceUniqueId': 'string',

```

(continues on next page)

(continued from previous page)

```

'IpAddress': 'string',
'IpAddressV6': 'string',
'IpType': 'string',
'IsoImageSet': [
    {
        'IsoImageId': 'string',
        'IsoImageName': 'string'
    },
],
'KernelId': 'string',
'KeyName': 'string',
'LaunchTime': datetime(2015, 1, 1),
'Loadbalancing': [
    {
        'InstancePort': 123,
        'LoadBalancerName': 'string',
        'LoadBalancerPort': 123,
        'State': 'string'
    },
],
'Monitoring': {
    'State': 'string'
},
'MultiIpAddressGroup': {
    'MultiIpAddressGroupId': 'string',
    'MultiIpAddressGroupName': 'string'
},
'NetworkInterfaceSet': [
    {
        'Association': {
            'IpOwnerId': 'string',
            'PublicDnsName': 'string',
            'PublicIp': 'string',
            'PublicIpV6': 'string'
        },
        'Attachment': {
            'AttachTime': 'string',
            'AttachmentId': 'string',
            'DeleteOnTermination': 'string',
            'DeviceIndex': 'string',
            'Status': 'string'
        },
        'Description': 'string',
        'GroupSet': [
            {
                'GroupId': 'string'
            },
        ],
        'MacAddress': 'string',
        'MultiIpAddressesSet': [
            {
                'IpAddress': 'string'
            },
        ],
        'NetworkInterfaceId': 'string',
        'NiftyNetworkId': 'string',
        'NiftyNetworkName': 'string',

```

(continues on next page)

(continued from previous page)

```

        'OwnerId': 'string',
        'PrivateDnsName': 'string',
        'PrivateIpAddress': 'string',
        'PrivateIpAddressV6': 'string',
        'PrivateIpAddressesSet': [
            {
                'Association': {
                    'IpOwnerId': 'string',
                    'PublicDnsName': 'string',
                    'PublicIp': 'string',
                    'PublicIpV6': 'string'
                },
                'Primary': True|False,
                'PrivateDnsName': 'string',
                'PrivateIpAddress': 'string'
            },
        ],
        'SourceDestCheck': 'string',
        'Status': 'string',
        'SubnetId': 'string',
        'VpcId': 'string'
    },
],
'NextMonthAccountingType': 'string',
'NiftyElasticLoadBalancing': [
    {
        'ElasticLoadBalancerId': 'string',
        'ElasticLoadBalancerName': 'string',
        'ElasticLoadBalancerPort': 123,
        'InstancePort': 123,
        'Protocol': 'string'
    },
],
'NiftyPrivateIpType': 'string',
'NiftyPrivateNetworkType': 'string',
'NiftySnapshotting': [
    {
        'State': 'string'
    },
],
'Placement': {
    'AvailabilityZone': 'string'
},
'Platform': 'string',
'PrivateDnsName': 'string',
'PrivateIpAddress': 'string',
'PrivateIpAddressV6': 'string',
'ProductCodes': [
    {
        'ProductCode': 'string'
    },
],
'RamdiskId': 'string',
'Reason': 'string',
'RootDeviceName': 'string',
'RootDeviceType': 'string',
'SpotInstanceRequestId': 'string',

```

(continues on next page)

(continued from previous page)

```

        'StateReason': {
            'Code': 'string',
            'Message': 'string'
        },
        'SubnetId': 'string',
        'Tenancy': 'string',
        'VmTools': {
            'State': 'string',
            'Version': 'string'
        },
        'VpcId': 'string'
    },
    ],
    'OwnerId': 'string',
    'ReservationId': 'string'
},
]
}

```

### Response Structure

- (dict) –
  - **RequestId** (string) –
  - **ReservationSet** (list) –
    - \* (dict) –
      - **GroupSet** (list) –
      - (dict) –
      - **GroupId** (string) –
      - **InstancesSet** (list) –
      - (dict) –
      - **AccountingType** (string) –
      - **AmiLaunchIndex** (string) –
      - **Architecture** (string) –
      - **Autoscaling** (dict) –
      - **AutoScalingGroupName** (string) –
      - **ExpireTime** (datetime) –
      - **BlockDeviceMapping** (list) –
      - (dict) –
      - **DeviceName** (string) –
      - **Ebs** (dict) –
      - **AttachTime** (string) –
      - **DeleteOnTermination** (string) –
      - **Status** (string) –
      - **VolumeId** (string) –
      - **VolumeUniqueId** (string) –
      - **CopyInfo** (string) –
      - **Description** (string) –
      - **DnsName** (string) –
      - **HotAdd** (string) –
      - **ImageId** (string) –
      - **ImageName** (string) –
      - **InstanceBackupRule** (dict) –
      - **InstanceBackupRuleId** (string) –
      - **InstanceBackupRuleName** (string) –
      - **InstanceId** (string) –

- **InstanceLifecycle** (*string*) –
- **InstanceState** (*dict*) –
- **Code** (*integer*) –
- **Name** (*string*) –
- **InstanceType** (*string*) –
- **InstanceUniqueId** (*string*) –
- **IpAddress** (*string*) –
- **IpAddressV6** (*string*) –
- **IpType** (*string*) –
- **IsoImageSet** (*list*) –
- (*dict*) –
- **IsoImageId** (*string*) –
- **IsoImageName** (*string*) –
- **KernelId** (*string*) –
- **KeyName** (*string*) –
- **LaunchTime** (*datetime*) –
- **Loadbalancing** (*list*) –
- (*dict*) –
- **InstancePort** (*integer*) –
- **LoadBalancerName** (*string*) –
- **LoadBalancerPort** (*integer*) –
- **State** (*string*) –
- **Monitoring** (*dict*) –
- **State** (*string*) –
- **MultiIpAddressGroup** (*dict*) –
- **MultiIpAddressGroupId** (*string*) –
- **MultiIpAddressGroupName** (*string*) –
- **NetworkInterfaceSet** (*list*) –
- (*dict*) –
- **Association** (*dict*) –
- **IpOwnerId** (*string*) –
- **PublicDnsName** (*string*) –
- **PublicIp** (*string*) –
- **PublicIpV6** (*string*) –
- **Attachment** (*dict*) –
- **AttachTime** (*string*) –
- **AttachmentId** (*string*) –
- **DeleteOnTermination** (*string*) –
- **DeviceIndex** (*string*) –
- **Status** (*string*) –
- **Description** (*string*) –
- **GroupSet** (*list*) –
- (*dict*) –
- **GroupId** (*string*) –
- **MacAddress** (*string*) –
- **MultiIpAddressesSet** (*list*) –
- (*dict*) –
- **IpAddress** (*string*) –
- **NetworkInterfaceId** (*string*) –
- **NiftyNetworkId** (*string*) –
- **NiftyNetworkName** (*string*) –
- **OwnerId** (*string*) –
- **PrivateDnsName** (*string*) –
- **PrivateIpAddress** (*string*) –

- **PrivateIpAddressV6** (*string*) –
- **PrivateIpAddressesSet** (*list*) –
- (*dict*) –
- **Association** (*dict*) –
- **IpOwnerId** (*string*) –
- **PublicDnsName** (*string*) –
- **PublicIp** (*string*) –
- **PublicIpV6** (*string*) –
- **Primary** (*boolean*) –
- **PrivateDnsName** (*string*) –
- **PrivateIpAddress** (*string*) –
- **SourceDestCheck** (*string*) –
- **Status** (*string*) –
- **SubnetId** (*string*) –
- **VpcId** (*string*) –
- **NextMonthAccountingType** (*string*) –
- **NiftyElasticLoadBalancing** (*list*) –
- (*dict*) –
- **ElasticLoadBalancerId** (*string*) –
- **ElasticLoadBalancerName** (*string*) –
- **ElasticLoadBalancerPort** (*integer*) –
- **InstancePort** (*integer*) –
- **Protocol** (*string*) –
- **NiftyPrivateIpType** (*string*) –
- **NiftyPrivateNetworkType** (*string*) –
- **NiftySnapshotting** (*list*) –
- (*dict*) –
- **State** (*string*) –
- **Placement** (*dict*) –
- **AvailabilityZone** (*string*) –
- **Platform** (*string*) –
- **PrivateDnsName** (*string*) –
- **PrivateIpAddress** (*string*) –
- **PrivateIpAddressV6** (*string*) –
- **ProductCodes** (*list*) –
- (*dict*) –
- **ProductCode** (*string*) –
- **RamdiskId** (*string*) –
- **Reason** (*string*) –
- **RootDeviceName** (*string*) –
- **RootDeviceType** (*string*) –
- **SpotInstanceRequestId** (*string*) –
- **StateReason** (*dict*) –
- **Code** (*string*) –
- **Message** (*string*) –
- **SubnetId** (*string*) –
- **Tenancy** (*string*) –
- **VmTools** (*dict*) –
- **State** (*string*) –
- **Version** (*string*) –
- **VpcId** (*string*) –
- **OwnerId** (*string*) –
- **ReservationId** (*string*) –

*computing* / Client / describe\_iso\_images

## describe\_iso\_images

computing.Client.**describe\_iso\_images** (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_iso_images(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'iso-image-id'|'iso-image-name'|'availability-zone'|
→ 'description'
        },
    ],
    IsoImageId='string'
)
```

### Parameters

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
  - **IsoImageId** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
    'IsoImagesSet': [
        {
            'AvailabilityZone': 'string',
            'CreatedTime': 'string',
            'Description': 'string',
            'ExpiredTime': 'string',
            'InstancesSet': [
                {
                    'InstanceId': 'string',
                    'InstanceUniqueId': 'string'
                },
            ],
            'IsoImageId': 'string',
            'IsoImageName': 'string',
            'IsoImageSize': 'string',
            'Status': 'string'
        },
    ],
    'RequestId': 'string'
}
```

### Response Structure

- (*dict*) –
  - **IsoImagesSet** (*list*) –
    - \* (*dict*) –
      - **AvailabilityZone** (*string*) –
      - **CreatedTime** (*string*) –
      - **Description** (*string*) –
      - **ExpiredTime** (*string*) –
      - **InstancesSet** (*list*) –
      - (*dict*) –
      - **InstanceId** (*string*) –
      - **InstanceUniqueId** (*string*) –
      - **IsoImageId** (*string*) –
      - **IsoImageName** (*string*) –
      - **IsoImageSize** (*string*) –
      - **Status** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / describe\_key\_pairs

## describe\_key\_pairs

`computing.Client.describe_key_pairs(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_key_pairs(  
    KeyName=[  
        'string',  
    ]  
)
```

**Parameters** **KeyName** (*list*) –

- (*string*) –

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'KeySet': [  
        {  
            'Description': 'string',  
            'InstancesSet': [  
                {  
                    'InstanceId': 'string',  
                    'RegionName': 'string'  
                },  
            ],  
            'KeyFingerprint': 'string',  
            'KeyName': 'string'  
        },  
    ],  
    'RequestId': 'string'  
}
```

### Response Structure



- *(dict)* –
  - **KeySet** (*list*) –
    - \* *(dict)* –
      - **Description** (*string*) –
      - **InstancesSet** (*list*) –
      - *(dict)* –
      - **InstanceId** (*string*) –
      - **RegionName** (*string*) –
      - **KeyFingerprint** (*string*) –
      - **KeyName** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / describe\_load\_balancers

## describe\_load\_balancers

`computing.Client.describe_load_balancers(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_load_balancers(
    LoadBalancerNames=[
        {
            'InstancePort': 123,
            'LoadBalancerName': 'string',
            'LoadBalancerPort': 123
        },
    ],
    Owner='self' | 'other' | 'all'
)
```

### Parameters

- **LoadBalancerNames** (*list*) –
  - *(dict)* –
    - \* **InstancePort** (*integer*) –
    - \* **LoadBalancerName** (*string*) –
    - \* **LoadBalancerPort** (*integer*) –
- **Owner** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
    'DescribeLoadBalancersResult': {
        'LoadBalancerDescriptions': [
            {
                'AccountingType': 'string',
                'AvailabilityZones': [
                    'string',
                ],
                'CreatedTime': datetime(2015, 1, 1),
                'DNSName': 'string',
                'Description': 'string',
                'Filter': {
```

(continues on next page)

(continued from previous page)

```

        'FilterType': 'string',
        'IPAddresses': [
            {
                'IPAddress': 'string'
            },
        ],
    },
    'HealthCheck': {
        'HealthyThreshold': 123,
        'InstanceStates': [
            {
                'Description': 'string',
                'InstanceId': 'string',
                'InstanceUniqueId': 'string',
                'ReasonCode': 'string',
                'State': 'string'
            },
        ],
        'Interval': 123,
        'Target': 'string',
        'Timeout': 123,
        'UnhealthyThreshold': 123
    },
    'Instances': [
        {
            'InstanceId': 'string',
            'InstanceUniqueId': 'string'
        },
    ],
    'ListenerDescriptions': [
        {
            'Listener': {
                'BalancingType': 123,
                'InstancePort': 123,
                'LoadBalancerPort': 123,
                'Protocol': 'string',
                'SSLCertificateId': 'string',
                'SSLPolicy': {
                    'SSLPolicyId': 'string',
                    'SSLPolicyName': 'string'
                }
            },
        },
    ],
    'LoadBalancerName': 'string',
    'NetworkVolume': 123,
    'NextMonthAccountingType': 'string',
    'Option': {
        'SessionStickinessPolicy': {
            'Enabled': True|False,
            'ExpirationPeriod': 123
        },
        'SorryPage': {
            'Enabled': True|False,
            'StatusCode': 123
        }
    },

```

(continues on next page)

(continued from previous page)

```

        'Policies': {
            'AppCookieStickinessPolicies': [
                {
                    'CookieName': 'string',
                    'PolicyName': 'string'
                },
            ],
            'LBCookieStickinessPolicies': [
                {
                    'CookieExpirationPeriod': 'string',
                    'PolicyName': 'string'
                },
            ],
        },
        'PolicyType': 'string'
    },
]
},
'ResponseMetadata': {
    'RequestId': 'string'
}
}

```

**Response Structure**

- (dict) –
  - **DescribeLoadBalancersResult** (dict) –
    - \* **LoadBalancerDescriptions** (list) –
      - (dict) –
      - **AccountingType** (string) –
      - **AvailabilityZones** (list) –
      - (string) –
      - **CreatedTime** (datetime) –
      - **DNSName** (string) –
      - **Description** (string) –
      - **Filter** (dict) –
      - **FilterType** (string) –
      - **IPAddresses** (list) –
      - (dict) –
      - **IPAddress** (string) –
      - **HealthCheck** (dict) –
      - **HealthyThreshold** (integer) –
      - **InstanceStates** (list) –
      - (dict) –
      - **Description** (string) –
      - **InstanceId** (string) –
      - **InstanceUniqueId** (string) –
      - **ReasonCode** (string) –
      - **State** (string) –
      - **Interval** (integer) –
      - **Target** (string) –
      - **Timeout** (integer) –
      - **UnhealthyThreshold** (integer) –
      - **Instances** (list) –
      - (dict) –

- **InstanceId** (*string*) –
- **InstanceUniqueId** (*string*) –
- **ListenerDescriptions** (*list*) –
- (*dict*) –
- **Listener** (*dict*) –
- **BalancingType** (*integer*) –
- **InstancePort** (*integer*) –
- **LoadBalancerPort** (*integer*) –
- **Protocol** (*string*) –
- **SSLCertificateId** (*string*) –
- **SSLPolicy** (*dict*) –
- **SSLPolicyId** (*string*) –
- **SSLPolicyName** (*string*) –
- **LoadBalancerName** (*string*) –
- **NetworkVolume** (*integer*) –
- **NextMonthAccountingType** (*string*) –
- **Option** (*dict*) –
- **SessionStickinessPolicy** (*dict*) –
- **Enabled** (*boolean*) –
- **ExpirationPeriod** (*integer*) –
- **SorryPage** (*dict*) –
- **Enabled** (*boolean*) –
- **StatusCode** (*integer*) –
- **Policies** (*dict*) –
- **AppCookieStickinessPolicies** (*list*) –
- (*dict*) –
- **CookieName** (*string*) –
- **PolicyName** (*string*) –
- **LBCookieStickinessPolicies** (*list*) –
- (*dict*) –
- **CookieExpirationPeriod** (*string*) –
- **PolicyName** (*string*) –
- **PolicyType** (*string*) –
- **ResponseMetadata** (*dict*) –
  - \* **RequestId** (*string*) –

*computing* / Client / describe\_multi\_ip\_address\_groups

## describe\_multi\_ip\_address\_groups

`computing.Client.describe_multi_ip_address_groups (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_multi_ip_address_groups(  
    MultiIpAddressGroupId=[  
        'string',  
    ]  
)
```

**Parameters** **MultiIpAddressGroupId** (*list*) –

- (*string*) –

**Return type** dict

**Returns**

## Response Syntax

```
{
  'MultiIpAddressGroupsSet': [
    {
      'AvailabilityZone': 'string',
      'CreateTime': 'string',
      'Description': 'string',
      'InstancesSet': [
        {
          'InstanceId': 'string',
          'InstanceUniqueId': 'string'
        },
      ],
      'MultiIpAddressGroupId': 'string',
      'MultiIpAddressGroupName': 'string',
      'MultiIpAddressNetwork': {
        'DefaultGateway': 'string',
        'IpAddressesSet': [
          {
            'IpAddress': 'string'
          },
        ],
        'SubnetMask': 'string'
      },
      'Status': 'string'
    },
  ],
  'RequestId': 'string'
}
```

## Response Structure

- (dict) –
  - **MultiIpAddressGroupsSet** (list) –
    - \* (dict) –
      - **AvailabilityZone** (string) –
      - **CreateTime** (string) –
      - **Description** (string) –
      - **InstancesSet** (list) –
      - (dict) –
        - **InstanceId** (string) –
        - **InstanceUniqueId** (string) –
      - **MultiIpAddressGroupId** (string) –
      - **MultiIpAddressGroupName** (string) –
      - **MultiIpAddressNetwork** (dict) –
        - **DefaultGateway** (string) –
        - **IpAddressesSet** (list) –
          - (dict) –
            - **IpAddress** (string) –
          - **SubnetMask** (string) –
        - **Status** (string) –
    - **RequestId** (string) –

*computing* / Client / describe\_network\_interfaces

## describe\_network\_interfaces

computing.Client.**describe\_network\_interfaces** (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_network_interfaces(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'attachment.instance-id'|'availability-zone'|'description'|
↪ 'network-interface-id'|'nifty-network-id'|'nifty-network-name'
        },
    ],
    NetworkInterfaceId=[
        'string',
    ]
)
```

### Parameters

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **NetworkInterfaceId** (*list*) –
  - (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
    'NetworkInterfaceSet': [
        {
            'Association': {
                'AllocationId': 'string',
                'AssociationId': 'string',
                'IpOwnerId': 'string',
                'PublicDnsName': 'string',
                'PublicIp': 'string',
                'PublicIpV6': 'string'
            },
            'Attachment': {
                'AttachTime': 'string',
                'AttachmentId': 'string',
                'DeleteOnTermination': 'string',
                'DeviceIndex': 'string',
                'InstanceId': 'string',
                'InstanceOwnerId': 'string',
                'Status': 'string'
            },
            'AvailabilityZone': 'string',
            'Description': 'string',
            'GroupSet': [
```

(continues on next page)

(continued from previous page)

```

        {
            'GroupId': 'string'
        },
    ],
    'InterfaceType': 'string',
    'Ipv6AddressesSet': [
        {
            'Ipv6Address': 'string'
        },
    ],
    'MacAddress': 'string',
    'NetworkInterfaceId': 'string',
    'NiftyNetworkId': 'string',
    'NiftyNetworkName': 'string',
    'OwnerId': 'string',
    'PrivateDnsName': 'string',
    'PrivateIpAddress': 'string',
    'PrivateIpAddressV6': 'string',
    'PrivateIpAddressesSet': [
        {
            'Association': {
                'AllocationId': 'string',
                'AssociationId': 'string',
                'IpOwnerId': 'string',
                'PublicDnsName': 'string',
                'PublicIp': 'string',
                'PublicIpV6': 'string'
            },
            'Primary': 'string',
            'PrivateDnsName': 'string',
            'PrivateIpAddress': 'string'
        },
    ],
    'RequesterId': 'string',
    'RequesterManaged': 'string',
    'SourceDestCheck': 'string',
    'Status': 'string',
    'SubnetId': 'string',
    'TagSet': [
        {
            'Key': 'string',
            'Value': 'string'
        },
    ],
    'VpcId': 'string'
    },
    'RequestId': 'string'
}

```

**Response Structure**

- (dict) –
  - **NetworkInterfaceSet** (list) –
    - \* (dict) –
      - **Association** (dict) –
      - **AllocationId** (string) –
      - **AssociationId** (string) –

- **IpOwnerId** (*string*) –
- **PublicDnsName** (*string*) –
- **PublicIp** (*string*) –
- **PublicIpV6** (*string*) –
- **Attachment** (*dict*) –
- **AttachTime** (*string*) –
- **AttachmentId** (*string*) –
- **DeleteOnTermination** (*string*) –
- **DeviceIndex** (*string*) –
- **InstanceId** (*string*) –
- **InstanceOwnerId** (*string*) –
- **Status** (*string*) –
- **AvailabilityZone** (*string*) –
- **Description** (*string*) –
- **GroupSet** (*list*) –
- (*dict*) –
- **GroupId** (*string*) –
- **InterfaceType** (*string*) –
- **Ipv6AddressesSet** (*list*) –
- (*dict*) –
- **Ipv6Address** (*string*) –
- **MacAddress** (*string*) –
- **NetworkInterfaceId** (*string*) –
- **NiftyNetworkId** (*string*) –
- **NiftyNetworkName** (*string*) –
- **OwnerId** (*string*) –
- **PrivateDnsName** (*string*) –
- **PrivateIpAddress** (*string*) –
- **PrivateIpAddressV6** (*string*) –
- **PrivateIpAddressesSet** (*list*) –
- (*dict*) –
- **Association** (*dict*) –
- **AllocationId** (*string*) –
- **AssociationId** (*string*) –
- **IpOwnerId** (*string*) –
- **PublicDnsName** (*string*) –
- **PublicIp** (*string*) –
- **PublicIpV6** (*string*) –
- **Primary** (*string*) –
- **PrivateDnsName** (*string*) –
- **PrivateIpAddress** (*string*) –
- **RequesterId** (*string*) –
- **RequesterManaged** (*string*) –
- **SourceDestCheck** (*string*) –
- **Status** (*string*) –
- **SubnetId** (*string*) –
- **TagSet** (*list*) –
- (*dict*) –
- **Key** (*string*) –
- **Value** (*string*) –
- **VpcId** (*string*) –
- **RequestId** (*string*) –

*computing* / Client / describe\_regions



## describe\_regions

computing.Client.**describe\_regions** (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```

response = client.describe_regions(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'string'
        },
    ],
    RegionName=[
        'string',
    ]
)

```

### Parameters

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
  - **RegionName** (*list*) –
    - (*string*) –

**Return type** dict

### Returns

### Response Syntax

```

{
    'RegionInfo': [
        {
            'IsDefault': True|False,
            'MessageSet': [
                {
                    'Message': 'string'
                },
            ],
            'RegionEndpoint': 'string',
            'RegionName': 'string'
        },
    ],
    'RequestId': 'string'
}

```

### Response Structure

- (*dict*) –
  - **RegionInfo** (*list*) –
    - \* (*dict*) –
      - **IsDefault** (*boolean*) –
      - **MessageSet** (*list*) –
        - (*dict*) –
        - **Message** (*string*) –

- **RegionEndpoint** (*string*) –
- **RegionName** (*string*) –
- **RequestId** (*string*) –

*computing* / Client / describe\_remote\_access\_vpn\_gateway\_activities

## describe\_remote\_access\_vpn\_gateway\_activities

`computing.Client.describe_remote_access_vpn_gateway_activities(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_remote_access_vpn_gateway_activities(
    RemoteAccessVpnGatewayId='string'
)
```

**Parameters** `RemoteAccessVpnGatewayId` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
    'Log': 'string',
    'RemoteAccessVpnGatewayId': 'string',
    'RemoteAccessVpnGatewayName': 'string',
    'RequestId': 'string'
}
```

### Response Structure

- (*dict*) –
  - **Log** (*string*) –
  - **RemoteAccessVpnGatewayId** (*string*) –
  - **RemoteAccessVpnGatewayName** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / describe\_remote\_access\_vpn\_gateway\_client\_config

## describe\_remote\_access\_vpn\_gateway\_client\_config

`computing.Client.describe_remote_access_vpn_gateway_client_config(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_remote_access_vpn_gateway_client_config(
    RemoteAccessVpnGatewayId='string'
)
```

**Parameters** `RemoteAccessVpnGatewayId` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
    'Encoding': 'string',
    'FileData': 'string',
    'RequestId': 'string'
}
```

**Response Structure**

- (dict) –
  - **Encoding** (*string*) –
  - **FileData** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / `describe_remote_access_vpn_gateway_connections`

**describe\_remote\_access\_vpn\_gateway\_connections**

`computing.Client.describe_remote_access_vpn_gateway_connections(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.describe_remote_access_vpn_gateway_connections(
    RemoteAccessVpnGatewayId='string'
)
```

**Parameters** `RemoteAccessVpnGatewayId` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

**Response Syntax**

```
{
    'RemoteAccessVpnGatewayConnection': {
        'ConnectionCount': 123,
        'ConnectionSet': [
            {
                'AssignedIpAddress': 'string',
                'ClientIpAddress': 'string',
                'ConnectionId': 'string',
                'RemoteUserName': 'string',
                'StartTime': 'string'
            },
        ],
    },
    'RemoteAccessVpnGatewayId': 'string',
    'RemoteAccessVpnGatewayName': 'string',
    'RequestId': 'string'
}
```

**Response Structure**

- (dict) –
  - **RemoteAccessVpnGatewayConnection** (*dict*) –
    - \* **ConnectionCount** (*integer*) –
    - \* **ConnectionSet** (*list*) –
      - (*dict*) –
      - **AssignedIpAddress** (*string*) –
      - **ClientIpAddress** (*string*) –

- **ConnectionId** (*string*) –
- **RemoteUserName** (*string*) –
- **StartTime** (*string*) –
- **RemoteAccessVpnGatewayId** (*string*) –
- **RemoteAccessVpnGatewayName** (*string*) –
- **RequestId** (*string*) –

*computing* / Client / describe\_remote\_access\_vpn\_gateways

## describe\_remote\_access\_vpn\_gateways

`computing.Client.describe_remote_access_vpn_gateways(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_remote_access_vpn_gateways(
    RemoteAccessVpnGatewayId=[
        'string',
    ]
)
```

**Parameters** **RemoteAccessVpnGatewayId** (*list*) –

- (*string*) –

**Return type** dict

**Returns**

### Response Syntax

```
{
    'RemoteAccessVpnGatewaySet': [
        {
            'AccountingType': 'string',
            'AuthTypeSet': [
                {
                    'AuthType': 'string'
                },
            ],
            'AvailabilityZone': 'string',
            'CaCertificateId': 'string',
            'CipherSuiteSet': [
                {
                    'CipherSuite': 'string'
                },
            ],
            'ClientDownloadEndpoint': 'string',
            'ClientTunnelMode': 'string',
            'CreatedTime': 'string',
            'Description': 'string',
            'GroupSet': 'string',
            'IsConfiguredNat': 'string',
            'NetworkInterfaceSet': [
                {
                    'Association': {
                        'AllocationId': 'string',
                        'AssociationId': 'string',
                        'IpOwnerId': 'string',

```

(continues on next page)

(continued from previous page)

```

        'PublicDnsName': 'string',
        'PublicIp': 'string',
        'PublicIpV6': 'string'
    },
    'Attachment': {
        'AttachTime': 'string',
        'AttachmentId': 'string',
        'DeleteOnTermination': 'string',
        'DeviceIndex': 'string',
        'InstanceId': 'string',
        'InstanceOwnerId': 'string',
        'Status': 'string'
    },
    'AvailabilityZone': 'string',
    'Description': 'string',
    'GroupSet': 'string',
    'InterfaceType': 'string',
    'Ipv6AddressesSet': 'string',
    'MacAddress': 'string',
    'NetworkInterfaceId': 'string',
    'NiftyNetworkId': 'string',
    'NiftyNetworkName': 'string',
    'OwnerId': 'string',
    'PrivateDnsName': 'string',
    'PrivateIpAddress': 'string',
    'PrivateIpAddressV6': 'string',
    'PrivateIpAddressesSet': 'string',
    'RequesterId': 'string',
    'RequesterManaged': 'string',
    'SourceDestCheck': 'string',
    'Status': 'string',
    'SubnetId': 'string',
    'TagSet': 'string',
    'VpcId': 'string'
},
],
'NextMonthAccountingType': 'string',
'PoolNetworkCidr': 'string',
'PoolNetworkGatewayIpAddress': 'string',
'RemoteAccessVpnGatewayId': 'string',
'RemoteAccessVpnGatewayName': 'string',
'RemoteAccessVpnGatewayType': 'string',
'RemoteUserSet': [
    {
        'Description': 'string',
        'UserName': 'string'
    },
],
],
'RouteTableAssociationId': 'string',
'RouteTableId': 'string',
'SslCertificateId': 'string',
'Status': 'string',
'VersionInformation': {
    'IsLatest': 'string',
    'Version': 'string'
}
}
}

```

(continues on next page)

(continued from previous page)

```
    ],  
    'RequestId': 'string'  
}
```

**Response Structure**

- (dict) –
  - **RemoteAccessVpnGatewaySet** (list) –
    - \* (dict) –
      - **AccountingType** (string) –
      - **AuthTypeSet** (list) –
      - (dict) –
      - **AuthType** (string) –
      - **AvailabilityZone** (string) –
      - **CaCertificateId** (string) –
      - **CipherSuiteSet** (list) –
      - (dict) –
      - **CipherSuite** (string) –
      - **ClientDownloadEndpoint** (string) –
      - **ClientTunnelMode** (string) –
      - **CreatedTime** (string) –
      - **Description** (string) –
      - **GroupSet** (string) –
      - **IsConfiguredNat** (string) –
      - **NetworkInterfaceSet** (list) –
      - (dict) –
      - **Association** (dict) –
      - **AllocationId** (string) –
      - **AssociationId** (string) –
      - **IpOwnerId** (string) –
      - **PublicDnsName** (string) –
      - **PublicIp** (string) –
      - **PublicIpV6** (string) –
      - **Attachment** (dict) –
      - **AttachTime** (string) –
      - **AttachmentId** (string) –
      - **DeleteOnTermination** (string) –
      - **DeviceIndex** (string) –
      - **InstanceId** (string) –
      - **InstanceOwnerId** (string) –
      - **Status** (string) –
      - **AvailabilityZone** (string) –
      - **Description** (string) –
      - **GroupSet** (string) –
      - **InterfaceType** (string) –
      - **Ipv6AddressesSet** (string) –
      - **MacAddress** (string) –
      - **NetworkInterfaceId** (string) –
      - **NiftyNetworkId** (string) –
      - **NiftyNetworkName** (string) –
      - **OwnerId** (string) –
      - **PrivateDnsName** (string) –
      - **PrivateIpAddress** (string) –
      - **PrivateIpAddressV6** (string) –

- **PrivateIpAddressesSet** (*string*) –
  - **RequesterId** (*string*) –
  - **RequesterManaged** (*string*) –
  - **SourceDestCheck** (*string*) –
  - **Status** (*string*) –
  - **SubnetId** (*string*) –
  - **TagSet** (*string*) –
  - **VpcId** (*string*) –
  - **NextMonthAccountingType** (*string*) –
  - **PoolNetworkCidr** (*string*) –
  - **PoolNetworkGatewayIpAddress** (*string*) –
  - **RemoteAccessVpnGatewayId** (*string*) –
  - **RemoteAccessVpnGatewayName** (*string*) –
  - **RemoteAccessVpnGatewayType** (*string*) –
  - **RemoteUserSet** (*list*) –
  - (*dict*) –
  - **Description** (*string*) –
  - **UserName** (*string*) –
  - **RouteTableAssociationId** (*string*) –
  - **RouteTableId** (*string*) –
  - **SslCertificateId** (*string*) –
  - **Status** (*string*) –
  - **VersionInformation** (*dict*) –
  - **IsLatest** (*string*) –
  - **Version** (*string*) –
- **RequestId** (*string*) –

*computing* / Client / describe\_resources

## describe\_resources

`computing.Client.describe_resources()`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_resources()
```

**Return type** dict

**Returns**

### Response Syntax

```
{
    'RequestId': 'string',
    'ResourceInfo': {
        'AddDiskCount': 123,
        'AddDiskTotalSize': 123,
        'AutoScaleCount': 123,
        'CustomizeImageCount': 123,
        'DynamicIpCount': 123,
        'ElasticIpItemSet': [
            {
                'Count': 123,
                'Type': 'string'
```

(continues on next page)

(continued from previous page)

```

    },
],
'ElasticLoadBalancerCount': 123,
'InstanceBackupRuleCount': 123,
'InstanceItemSet': [
    {
        'Count': 123,
        'Type': 'string'
    },
],
'LoadBalancerCount': 123,
'MigrationHubItemSet': [
    {
        'Count': 123,
        'Type': 'string'
    },
],
'MonitoringRuleCount': 123,
'MultiIpAddressItemSet': [
    {
        'Count': 123,
        'Type': 'string'
    },
],
'NetworkFlowAmount': 123,
'NetworkInterfaceItemSet': [
    {
        'Count': 123,
        'Type': 'string'
    },
],
'NiftyMultiAccountCount': 123,
'PremiumSupportSet': [
    {
        'SupportName': 'string'
    },
],
'PrivateLanClassicCount': 123,
'PrivateLanCount': 123,
'RemoteAccessVpnGatewaySet': [
    {
        'Count': 123,
        'Type': 'string'
    },
],
'RouterItemSet': [
    {
        'Count': 123,
        'Type': 'string'
    },
],
'SecurityGroupCount': 123,
'SslCertCount': 123,
'VpnGatewayItemSet': [
    {
        'Count': 123,
        'Type': 'string'
    },
],

```

(continues on next page)



(continued from previous page)

```

    },
  ],
}
}

```

**Response Structure**

- *(dict)* –
  - **RequestId** (*string*) –
  - **ResourceInfo** (*dict*) –
    - \* **AddDiskCount** (*integer*) –
    - \* **AddDiskTotalSize** (*integer*) –
    - \* **AutoScaleCount** (*integer*) –
    - \* **CustomizeImageCount** (*integer*) –
    - \* **DynamicIpCount** (*integer*) –
    - \* **ElasticIpItemSet** (*list*) –
      - *(dict)* –
      - **Count** (*integer*) –
      - **Type** (*string*) –
    - \* **ElasticLoadBalancerCount** (*integer*) –
    - \* **InstanceBackupRuleCount** (*integer*) –
    - \* **InstanceItemSet** (*list*) –
      - *(dict)* –
      - **Count** (*integer*) –
      - **Type** (*string*) –
    - \* **LoadBalancerCount** (*integer*) –
    - \* **MigrationHubItemSet** (*list*) –
      - *(dict)* –
      - **Count** (*integer*) –
      - **Type** (*string*) –
    - \* **MonitoringRuleCount** (*integer*) –
    - \* **MultiIpAddressItemSet** (*list*) –
      - *(dict)* –
      - **Count** (*integer*) –
      - **Type** (*string*) –
    - \* **NetworkFlowAmount** (*integer*) –
    - \* **NetworkInterfaceItemSet** (*list*) –
      - *(dict)* –
      - **Count** (*integer*) –
      - **Type** (*string*) –
    - \* **NiftyMultiAccountCount** (*integer*) –
    - \* **PremiumSupportSet** (*list*) –
      - *(dict)* –
      - **SupportName** (*string*) –
    - \* **PrivateLanClassicCount** (*integer*) –
    - \* **PrivateLanCount** (*integer*) –
    - \* **RemoteAccessVpnGatewaySet** (*list*) –
      - *(dict)* –
      - **Count** (*integer*) –
      - **Type** (*string*) –
    - \* **RouterItemSet** (*list*) –
      - *(dict)* –
      - **Count** (*integer*) –
      - **Type** (*string*) –

- \* **SecurityGroupCount** (*integer*) –
- \* **SslCertCount** (*integer*) –
- \* **VpnGatewayItemSet** (*list*) –
  - (*dict*) –
  - **Count** (*integer*) –
  - **Type** (*string*) –

*computing* / Client / describe\_route\_tables

## describe\_route\_tables

`computing.Client.describe_route_tables(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_route_tables(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'association.route-table-association-id'|'association.route-
↪table-id'|'association.router-id'|'association.router-name'|'association.main'|
↪'route-table-id'|'route.destination-cidr-block'|'route.gateway-id'|'route.vpc-
↪peering-connection-id'|'route.origin'|'route.state'|'route.ip-address'|'route.
↪network-id'
        },
    ],
    RouteTableId=[
        'string',
    ]
)
```

### Parameters

- **Filter** (*list*) –
  - (*dict*) –
  - \* **ListOfRequestValue** (*list*) –
    - (*string*) –
  - \* **Name** (*string*) –
- **RouteTableId** (*list*) –
  - (*string*) –

**Return type** dict

**Returns**

### Response Syntax

```
{
    'RequestId': 'string',
    'RouteTableSet': [
        {
            'AssociationSet': [
                {
                    'Main': True|False,
                    'RouteTableAssociationId': 'string',
                    'RouteTableId': 'string',
                    'RouterId': 'string',
```

(continues on next page)

(continued from previous page)

```

        'RouterName': 'string'
    },
],
'ElasticLoadBalancerAssociationSet': [
    {
        'ElasticLoadBalancerId': 'string',
        'ElasticLoadBalancerName': 'string',
        'Main': True|False,
        'RouteTableAssociationId': 'string',
        'RouteTableId': 'string'
    },
],
'PropagatingVgwSet': [
    {
        'GatewayId': 'string',
        'NiftyGatewayName': 'string',
        'RouteTableAssociationId': 'string'
    },
],
'RouteSet': [
    {
        'DestinationCidrBlock': 'string',
        'IpAddress': 'string',
        'NetworkId': 'string',
        'NetworkName': 'string',
        'Origin': 'string',
        'Priority': 'string',
        'State': 'string',
        'VpcPeeringConnectionId': 'string'
    },
],
'RouteTableId': 'string',
'TagSet': [
    {
        'Key': 'string',
        'Value': 'string'
    },
],
]
}

```

**Response Structure**

- (dict) –
  - **RequestId** (string) –
  - **RouteTableSet** (list) –
    - \* (dict) –
      - **AssociationSet** (list) –
      - (dict) –
      - **Main** (boolean) –
      - **RouteTableAssociationId** (string) –
      - **RouteTableId** (string) –
      - **RouterId** (string) –
      - **RouterName** (string) –
      - **ElasticLoadBalancerAssociationSet** (list) –
      - (dict) –

- **ElasticLoadBalancerId** (*string*) –
- **ElasticLoadBalancerName** (*string*) –
- **Main** (*boolean*) –
- **RouteTableAssociationId** (*string*) –
- **RouteTableId** (*string*) –
- **PropagatingVgwSet** (*list*) –
- (*dict*) –
- **GatewayId** (*string*) –
- **NiftyGatewayName** (*string*) –
- **RouteTableAssociationId** (*string*) –
- **RouteSet** (*list*) –
- (*dict*) –
- **DestinationCidrBlock** (*string*) –
- **IpAddress** (*string*) –
- **NetworkId** (*string*) –
- **NetworkName** (*string*) –
- **Origin** (*string*) –
- **Priority** (*string*) –
- **State** (*string*) –
- **VpcPeeringConnectionId** (*string*) –
- **RouteTableId** (*string*) –
- **TagSet** (*list*) –
- (*dict*) –
- **Key** (*string*) –
- **Value** (*string*) –

*computing* / Client / describe\_security\_activities

## describe\_security\_activities

`computing.Client.describe_security_activities(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_security_activities(  
    ActivityDate='string',  
    GroupName='string',  
    Range={  
        'All': True|False,  
        'EndNumber': 123,  
        'StartNumber': 123  
    }  
)
```

### Parameters

- **ActivityDate** (*string*) –
- **GroupName** (*string*) – [REQUIRED]
- **Range** (*dict*) –
  - **All** (*boolean*) –
  - **EndNumber** (*integer*) –
  - **StartNumber** (*integer*) –

**Return type** dict

### Returns

### Response Syntax

```
{
    'GroupName': 'string',
    'Log': 'string',
    'RequestId': 'string'
}
```

**Response Structure**

- *(dict)* –
  - **GroupName** (*string*) –
  - **Log** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / describe\_security\_groups

**describe\_security\_groups**

`computing.Client.describe_security_groups(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.describe_security_groups(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'description'|'group-name'
        },
    ],
    GroupName=[
        'string',
    ]
)
```

**Parameters**

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **GroupName** (*list*) –
  - (*string*) –

**Return type** dict

**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'SecurityGroupInfo': [
        {
            'AvailabilityZone': 'string',
            'GroupDescription': 'string',
            'GroupLogFilterBroadcast': True|False,
            'GroupLogFilterNetBios': True|False,
```

(continues on next page)

(continued from previous page)

```

        'GroupLogLimit': 123,
        'GroupName': 'string',
        'GroupRuleLimit': 123,
        'GroupStatus': 'string',
        'InstanceUniqueIdsSet': [
            {
                'InstanceUniqueId': 'string'
            },
        ],
        'InstancesSet': [
            {
                'InstanceId': 'string'
            },
        ],
        'IpPermissions': [
            {
                'AddDatetime': datetime(2015, 1, 1),
                'Description': 'string',
                'FromPort': 123,
                'Groups': [
                    {
                        'GroupName': 'string',
                        'UserId': 'string'
                    },
                ],
                'InOut': 'string',
                'IpProtocol': 'string',
                'IpRanges': [
                    {
                        'CidrIp': 'string'
                    },
                ],
                'ToPort': 123
            },
        ],
        'OwnerId': 'string',
        'RouterSet': [
            {
                'RouterId': 'string',
                'RouterName': 'string'
            },
        ],
        'VpnGatewaySet': [
            {
                'NiftyVpnGatewayName': 'string',
                'VpnGatewayId': 'string'
            },
        ],
    ],
},
]
}

```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **SecurityGroupInfo** (*list*) –
    - \* (*dict*) –

- **AvailabilityZone** (*string*) –
- **GroupDescription** (*string*) –
- **GroupLogFilterBroadcast** (*boolean*) –
- **GroupLogFilterNetBios** (*boolean*) –
- **GroupLogLimit** (*integer*) –
- **GroupName** (*string*) –
- **GroupRuleLimit** (*integer*) –
- **GroupStatus** (*string*) –
- **InstanceUniqueIdsSet** (*list*) –
- (*dict*) –
- **InstanceUniqueId** (*string*) –
- **InstancesSet** (*list*) –
- (*dict*) –
- **InstanceId** (*string*) –
- **IpPermissions** (*list*) –
- (*dict*) –
- **AddDatetime** (*datetime*) –
- **Description** (*string*) –
- **FromPort** (*integer*) –
- **Groups** (*list*) –
- (*dict*) –
- **GroupName** (*string*) –
- **UserId** (*string*) –
- **InOut** (*string*) –
- **IpProtocol** (*string*) –
- **IpRanges** (*list*) –
- (*dict*) –
- **CidrIp** (*string*) –
- **ToPort** (*integer*) –
- **OwnerId** (*string*) –
- **RouterSet** (*list*) –
- (*dict*) –
- **RouterId** (*string*) –
- **RouterName** (*string*) –
- **VpnGatewaySet** (*list*) –
- (*dict*) –
- **NiftyVpnGatewayName** (*string*) –
- **VpnGatewayId** (*string*) –

*computing* / Client / describe\_service\_status

## describe\_service\_status

`computing.Client.describe_service_status(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_service_status(
    FromDate='string',
    ToDate='string'
)
```

### Parameters

- **FromDate** (*string*) –

- **ToDate** (*string*) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'RequestId': 'string',
  'ServiceStatusSet': [
    {
      'ControlPanelStatus': 'string',
      'Date': 'string',
      'DiskStatus': 'string',
      'InstanceStatus': 'string',
      'NetworkStatus': 'string',
      'StorageStatus': 'string'
    },
  ]
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **ServiceStatusSet** (*list*) –
    - \* (*dict*) –
      - **ControlPanelStatus** (*string*) –
      - **Date** (*string*) –
      - **DiskStatus** (*string*) –
      - **InstanceStatus** (*string*) –
      - **NetworkStatus** (*string*) –
      - **StorageStatus** (*string*) –

*computing* / Client / describe\_ssl\_certificate\_attribute

### describe\_ssl\_certificate\_attribute

`computing.Client.describe_ssl_certificate_attribute(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.describe_ssl_certificate_attribute(
    Attribute='certAuthority'|'count'|'certState'|'period'|'validityTerm'|
    → 'keyLength'|'uploadState'|'description'|'certInfo'|'caState',
    FqdnId='string'
)
```

#### Parameters

- **Attribute** (*string*) –
- **FqdnId** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'CaState': {
```

(continues on next page)



(continued from previous page)

```

        'Value': True|False
    },
    'CertAuthority': {
        'Value': 'string'
    },
    'CertInfo': {
        'CountryName': 'string',
        'EmailAddress': 'string',
        'LocationName': 'string',
        'OrganizationName': 'string',
        'OrganizationUnitName': 'string',
        'StateName': 'string'
    },
    'CertState': {
        'Value': 'string'
    },
    'Count': {
        'Value': 123
    },
    'Description': {
        'Value': 'string'
    },
    'Fqdn': 'string',
    'FqdnId': 'string',
    'KeyLength': {
        'Value': 123
    },
    'Period': {
        'EndDate': datetime(2015, 1, 1),
        'StartDate': datetime(2015, 1, 1),
        'ValidityTerm': 123
    },
    'RequestId': 'string',
    'UploadState': {
        'Value': True|False
    }
}

```

### Response Structure

- (dict) –
  - **CaState** (dict) –
    - \* **Value** (boolean) –
  - **CertAuthority** (dict) –
    - \* **Value** (string) –
  - **CertInfo** (dict) –
    - \* **CountryName** (string) –
    - \* **EmailAddress** (string) –
    - \* **LocationName** (string) –
    - \* **OrganizationName** (string) –
    - \* **OrganizationUnitName** (string) –
    - \* **StateName** (string) –
  - **CertState** (dict) –
    - \* **Value** (string) –
  - **Count** (dict) –
    - \* **Value** (integer) –
  - **Description** (dict) –

- \* **Value** (*string*) –
- **Fqdn** (*string*) –
- **FqdnId** (*string*) –
- **KeyLength** (*dict*) –
- \* **Value** (*integer*) –
- **Period** (*dict*) –
- \* **EndDate** (*datetime*) –
- \* **StartDate** (*datetime*) –
- \* **ValidityTerm** (*integer*) –
- **RequestId** (*string*) –
- **UploadState** (*dict*) –
- \* **Value** (*boolean*) –

*computing* / Client / describe\_ssl\_certificates

## describe\_ssl\_certificates

`computing.Client.describe_ssl_certificates(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_ssl_certificates(
    Fqdn=[
        'string',
    ],
    FqdnId=[
        'string',
    ]
)
```

### Parameters

- **Fqdn** (*list*) –
  - (*string*) –
- **FqdnId** (*list*) –
  - (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
    'CertsSet': [
        {
            'CaState': True|False,
            'CertAuthority': 'string',
            'CertInfo': {
                'CountryName': 'string',
                'EmailAddress': 'string',
                'LocationName': 'string',
                'OrganizationName': 'string',
                'OrganizationUnitName': 'string',
                'StateName': 'string'
            },
            'CertState': 'string',
            'Count': 123,
```

(continues on next page)

(continued from previous page)

```

        'Description': 'string',
        'Fqdn': 'string',
        'FqdnId': 'string',
        'KeyLength': 123,
        'Period': {
            'EndDate': datetime(2015, 1, 1),
            'StartDate': datetime(2015, 1, 1),
            'ValidityTerm': 123
        },
        'UploadState': True|False
    },
    'RequestId': 'string'
}

```

**Response Structure**

- (dict) –
  - CertsSet (list) –
    - \* (dict) –
      - CaState (boolean) –
      - CertAuthority (string) –
      - CertInfo (dict) –
      - CountryName (string) –
      - EmailAddress (string) –
      - LocationName (string) –
      - OrganizationName (string) –
      - OrganizationUnitName (string) –
      - StateName (string) –
      - CertState (string) –
      - Count (integer) –
      - Description (string) –
      - Fqdn (string) –
      - FqdnId (string) –
      - KeyLength (integer) –
      - Period (dict) –
      - EndDate (datetime) –
      - StartDate (datetime) –
      - ValidityTerm (integer) –
      - UploadState (boolean) –
  - RequestId (string) –

*computing* / Client / describe\_uploads**describe\_uploads**`computing.Client.describe_uploads (**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```

response = client.describe_uploads(
    ConversionTaskId=[
        'string',

```

(continues on next page)

(continued from previous page)

```
]
)
```

**Parameters** `ConversionTaskId` (*list*) –

- (*string*) –

**Return type** dict**Returns****Response Syntax**

```
{
  'Uploads': [
    {
      'ConversionTaskId': 123,
      'ExpirationTime': 'string',
      'ImportInstance': {
        'AvailabilityZone': 'string',
        'Image': {
          'Format': 'string',
          'Size': 123
        },
        'InstanceId': 'string',
        'InstanceUniqueId': 'string'
      }
    },
  ]
}
```

**Response Structure**

- (*dict*) –
  - **Uploads** (*list*) –
    - \* (*dict*) –
      - **ConversionTaskId** (*integer*) –
      - **ExpirationTime** (*string*) –
      - **ImportInstance** (*dict*) –
      - **AvailabilityZone** (*string*) –
      - **Image** (*dict*) –
      - **Format** (*string*) –
      - **Size** (*integer*) –
      - **InstanceId** (*string*) –
      - **InstanceUniqueId** (*string*) –

*computing* / Client / describe\_usage**describe\_usage**`computing.Client.describe_usage(**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.describe_usage(
    IsCharge=True|False,
    Region='string',
```

(continues on next page)

(continued from previous page)

```
YearMonth='string'
)
```

**Parameters**

- **IsCharge** (*boolean*) –
- **Region** (*string*) –
- **YearMonth** (*string*) –

**Return type** dict**Returns****Response Syntax**

```
{
  'AutoScaleInfo': {
    'AutoScaleCount': {
      'Charge': 123,
      'Type': 'string',
      'Unit': 'string',
      'Value': 123
    },
    'RunningScaleOutInstanceSet': [
      {
        'Charge': 123,
        'Type': 'string',
        'Unit': 'string',
        'Value': 123
      },
    ],
    'RunningScaleOutOsSet': [
      {
        'Charge': 123,
        'Type': 'string',
        'Unit': 'string',
        'Value': 123
      },
    ],
    'StoppedScaleOutInstanceSet': [
      {
        'Charge': 123,
        'Type': 'string',
        'Unit': 'string',
        'Value': 123
      },
    ],
    'StoppedScaleOutOsSet': [
      {
        'Charge': 123,
        'Type': 'string',
        'Unit': 'string',
        'Value': 123
      },
    ],
  },
  'ChargeDetailInfo': {
    'ChargeDetail': {
      'Charge': 123,
```

(continues on next page)

(continued from previous page)

```

        'Value': 123
    },
    'CopyInfo': {
        'InstanceCopy': {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
            'Value': 123
        }
    },
    'ElasticIpInfo': {
        'ElasticIpSet': [
            {
                'Charge': 123,
                'Type': 'string',
                'Unit': 'string',
                'Value': 123
            }
        ]
    },
    'ElasticLoadBalancerInfo': {
        'Vip': [
            {
                'Charge': 123,
                'Type': 'string',
                'Unit': 'string',
                'Value': 123
            }
        ],
        'VipMeasuredRate': [
            {
                'Charge': 123,
                'Type': 'string',
                'Unit': 'string',
                'Value': 123
            }
        ]
    },
    'ExtraChargeInfo': {
        'ExtraChargeMonthlyRateSet': [
            {
                'Charge': 123,
                'Type': 'string',
                'Unit': 'string',
                'Value': 123
            }
        ]
    },
    'ImageInfo': {
        'CreateImage': {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
            'Value': 123
        },
        'KeepImageSet': [

```

(continues on next page)

(continued from previous page)

```

        {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
            'Value': 123
        },
    ],
},
'InstanceBackupInfo': {
    'InstanceBackupBaseSet': [
        {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
            'Value': 123
        },
    ],
    'InstanceBackupImportInstanceDiskSet': {
        'Charge': 123,
        'Unit': 'string',
        'Value': 123
    },
    'InstanceBackupVolumeSet': {
        'Charge': 123,
        'Unit': 'string',
        'Value': 123
    }
},
'InstanceInfo': {
    'DynamicIpMeasuredRate': {
        'Charge': 123,
        'Type': 'string',
        'Unit': 'string',
        'Value': 123
    },
    'DynamicIpMonthlyRate': {
        'Charge': 123,
        'Type': 'string',
        'Unit': 'string',
        'Value': 123
    },
    'InstanceMonthlyRateSet': [
        {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
            'Value': 123
        },
    ],
    'MultiIpMonthlyRate': {
        'Charge': 123,
        'Type': 'string',
        'Unit': 'string',
        'Value': 123
    },
    'OsMeasuredRate': [
        {

```

(continues on next page)

(continued from previous page)

```

        'Charge': 123,
        'Type': 'string',
        'Unit': 'string',
        'Value': 123
    },
],
'OsMonthlyRate': [
    {
        'Charge': 123,
        'Type': 'string',
        'Unit': 'string',
        'Value': 123
    },
],
'RunningInstanceMeasuredRateSet': [
    {
        'Charge': 123,
        'Type': 'string',
        'Unit': 'string',
        'Value': 123
    },
],
'StoppedInstanceMeasuredRateSet': [
    {
        'Charge': 123,
        'Type': 'string',
        'Unit': 'string',
        'Value': 123
    },
],
],
'InternetVpnInfo': {
    'InternetVpnInitial': {
        'Charge': 123,
        'Type': 'string',
        'Unit': 'string',
        'Value': 123
    },
    'InternetVpnMonthlyRateSet': [
        {
            'Charge': 123,
            'Type': 'string',
            'Value': 123
        },
    ]
},
'LicenseInfo': {
    'LicenseMonthlyRateSet': [
        {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
            'Value': 123
        },
    ],
},
'LiveMigrationInfo': {

```

(continues on next page)



(continued from previous page)

```

    'LiveMigrationHubMonthlyRateSet': [
        {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
            'Value': 123
        },
    ],
    'LiveMigrationOperationMonthlyRateSet': [
        {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
            'Value': 123
        },
    ],
],
'LoadBalancerInfo': {
    'OptionSet': [
        {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
            'Value': 123
        },
    ],
    'VipMeasuredRateSet': [
        {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
            'Value': 123
        },
    ],
    'VipSet': [
        {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
            'Value': 123
        },
    ],
},
'MultiAccountInfo': {
    'MultiAccount': {
        'Charge': 123,
        'Unit': 'string',
        'Value': 123
    }
},
'MultiIpAddressInfo': {
    'MultiIpAddressMonthlyRateSet': [
        {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
            'Value': 123
        },
    ],

```

(continues on next page)

(continued from previous page)

```

        },
    ],
},
'NetworkInfo': {
    'NetworkFlowSet': [
        {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
            'Value': 123
        },
    ],
},
'NetworkInterfaceInfo': {
    'NetworkInterfaceMonthlyRateSet': [
        {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
            'Value': 123
        },
    ],
},
'OptionCommonInfo': [
    {
        'OptionName': 'string',
        'OptionSet': [
            {
                'Charge': 123,
                'Type': 'string',
                'Unit': 'string',
                'Value': 123
            },
        ],
    },
],
'OptionInfo': [
    {
        'OptionName': 'string',
        'OptionSet': [
            {
                'Charge': 123,
                'Type': 'string',
                'Unit': 'string',
                'Value': 123
            },
        ],
    },
],
'OsOptionChargeInfo': {
    'OsOptionChargeMonthlyRateSet': [
        {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
            'Value': 123
        },
    ],
},

```

(continues on next page)

(continued from previous page)

```

    ],
    'PatternAuthInfo': {
        'PatternAuthSet': [
            {
                'Charge': 123,
                'Type': 'string',
                'Unit': 'string',
                'Value': 123
            },
        ],
    },
    'PremiumSupportInfo': {
        'PremiumSupportSet': [
            {
                'Charge': 123,
                'Type': 'string',
                'Unit': 'string',
                'Value': 123
            },
        ],
    },
    'PrivateLanInfo': {
        'PrivateLan': {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
            'Value': 123
        }
    },
    'PrivateNetworkInfo': {
        'PrivateNetworkMeasuredRate': {
            'Charge': 123,
            'Unit': 'string',
            'Value': 123
        },
        'PrivateNetworkMonthlyRate': {
            'Charge': 123,
            'Value': 123
        }
    },
    'RemoteAccessVpnGatewayInfo': {
        'RemoteAccessVpnGatewayMeasuredRateSet': [
            {
                'Charge': 123,
                'Type': 'string',
                'Unit': 'string',
                'Value': 123
            },
        ],
        'RemoteAccessVpnGatewayMonthlyRateSet': [
            {
                'Charge': 123,
                'Type': 'string',
                'Unit': 'string',
                'Value': 123
            },
        ],
    },

```

(continues on next page)

(continued from previous page)

```

    ],
    'RequestId': 'string',
    'RouterInfo': {
        'RouterMeasuredRateSet': [
            {
                'Charge': 123,
                'Type': 'string',
                'Unit': 'string',
                'Value': 123
            },
        ],
        'RouterMonthlyRateSet': [
            {
                'Charge': 123,
                'Type': 'string',
                'Unit': 'string',
                'Value': 123
            },
        ],
    },
    'SecurityGroupInfo': {
        'OptionSet': [
            {
                'Charge': 123,
                'Type': 'string',
                'Unit': 'string',
                'Value': 123
            },
        ],
        'SecurityGroupApplyTime': {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
            'Value': 123
        }
    },
    'SnapshotInfo': {
        'Snapshot': {
            'Charge': 123,
            'Unit': 'string',
            'Value': 123
        }
    },
    'SslCertInfo': {
        'CreateSslCertSet': [
            {
                'Charge': 123,
                'Type': 'string',
                'Unit': 'string',
                'Value': 123
            },
        ],
    },
    'VolumeInfo': {
        'ImportInstanceDiskMeasuredRate': {
            'Charge': 123,

```

(continues on next page)

(continued from previous page)

```

        'Type': 'string',
        'Unit': 'string',
        'Value': 123
    },
    'ImportInstanceDiskMonthlyRate': {
        'Charge': 123,
        'Type': 'string',
        'Unit': 'string',
        'Value': 123
    },
    'VolumeMeasuredRateSet': [
        {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
            'Value': 123
        },
    ],
    'VolumeSet': [
        {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
            'Value': 123
        },
    ],
    ],
    'VpnGatewayInfo': {
        'VpnGatewayMeasuredRateSet': [
            {
                'Charge': 123,
                'Type': 'string',
                'Unit': 'string',
                'Value': 123
            },
        ],
        'VpnGatewayMonthlyRateSet': [
            {
                'Charge': 123,
                'Type': 'string',
                'Unit': 'string',
                'Value': 123
            },
        ],
    ],
    'YearMonth': 'string'
}

```

**Response Structure**

- (dict) –
  - **AutoScaleInfo** (dict) –
    - \* **AutoScaleCount** (dict) –
      - **Charge** (integer) –
      - **Type** (string) –
      - **Unit** (string) –
      - **Value** (integer) –
    - \* **RunningScaleOutInstanceSet** (list) –

- *(dict)* –
- **Charge** (*integer*) –
- **Type** (*string*) –
- **Unit** (*string*) –
- **Value** (*integer*) –
- \* **RunningScaleOutOsSet** (*list*) –
  - *(dict)* –
  - **Charge** (*integer*) –
  - **Type** (*string*) –
  - **Unit** (*string*) –
  - **Value** (*integer*) –
- \* **StoppedScaleOutInstanceSet** (*list*) –
  - *(dict)* –
  - **Charge** (*integer*) –
  - **Type** (*string*) –
  - **Unit** (*string*) –
  - **Value** (*integer*) –
- \* **StoppedScaleOutOsSet** (*list*) –
  - *(dict)* –
  - **Charge** (*integer*) –
  - **Type** (*string*) –
  - **Unit** (*string*) –
  - **Value** (*integer*) –
- **ChargeDetailInfo** (*dict*) –
  - \* **ChargeDetail** (*dict*) –
    - **Charge** (*integer*) –
    - **Value** (*integer*) –
- **CopyInfo** (*dict*) –
  - \* **InstanceCopy** (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **ElasticIpInfo** (*dict*) –
  - \* **ElasticIpSet** (*list*) –
    - *(dict)* –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **ElasticLoadBalancerInfo** (*dict*) –
  - \* **Vip** (*list*) –
    - *(dict)* –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
  - \* **VipMeasuredRate** (*list*) –
    - *(dict)* –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **ExtraChargeInfo** (*dict*) –

- \* **ExtraChargeMonthlyRateSet** (*list*) –
  - (*dict*) –
  - **Charge** (*integer*) –
  - **Type** (*string*) –
  - **Unit** (*string*) –
  - **Value** (*integer*) –
- **ImageInfo** (*dict*) –
  - \* **CreateImage** (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
  - \* **KeepImageSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **InstanceBackupInfo** (*dict*) –
  - \* **InstanceBackupBaseSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
  - \* **InstanceBackupImportInstanceDiskSet** (*dict*) –
    - **Charge** (*integer*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
  - \* **InstanceBackupVolumeSet** (*dict*) –
    - **Charge** (*integer*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **InstanceInfo** (*dict*) –
  - \* **DynamicIpMeasuredRate** (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
  - \* **DynamicIpMonthlyRate** (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
  - \* **InstanceMonthlyRateSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
  - \* **MultiIpMonthlyRate** (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –

- **Value** (*integer*) –
- \* **OsMeasuredRate** (*list*) –
  - (*dict*) –
  - **Charge** (*integer*) –
  - **Type** (*string*) –
  - **Unit** (*string*) –
  - **Value** (*integer*) –
- \* **OsMonthlyRate** (*list*) –
  - (*dict*) –
  - **Charge** (*integer*) –
  - **Type** (*string*) –
  - **Unit** (*string*) –
  - **Value** (*integer*) –
- \* **RunningInstanceMeasuredRateSet** (*list*) –
  - (*dict*) –
  - **Charge** (*integer*) –
  - **Type** (*string*) –
  - **Unit** (*string*) –
  - **Value** (*integer*) –
- \* **StoppedInstanceMeasuredRateSet** (*list*) –
  - (*dict*) –
  - **Charge** (*integer*) –
  - **Type** (*string*) –
  - **Unit** (*string*) –
  - **Value** (*integer*) –
- **InternetVpnInfo** (*dict*) –
  - \* **InternetVpnInitial** (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
  - \* **InternetVpnMonthlyRateSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Value** (*integer*) –
- **LicenseInfo** (*dict*) –
  - \* **LicenseMonthlyRateSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **LiveMigrationInfo** (*dict*) –
  - \* **LiveMigrationHubMonthlyRateSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
  - \* **LiveMigrationOperationMonthlyRateSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –



- **Unit** (*string*) –
  - **Value** (*integer*) –
- **LoadBalancerInfo** (*dict*) –
  - \* **OptionSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
  - \* **VipMeasuredRateSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
  - \* **VipSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **MultiAccountInfo** (*dict*) –
  - \* **MultiAccount** (*dict*) –
    - **Charge** (*integer*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **MultiIpAddressInfo** (*dict*) –
  - \* **MultiIpAddressMonthlyRateSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **NetworkInfo** (*dict*) –
  - \* **NetworkFlowSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **NetworkInterfaceInfo** (*dict*) –
  - \* **NetworkInterfaceMonthlyRateSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **OptionCommonInfo** (*list*) –
  - \* (*dict*) –
    - **OptionName** (*string*) –
    - **OptionSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –

- **Unit** (*string*) –
- **Value** (*integer*) –
- **OptionInfo** (*list*) –
  - \* (*dict*) –
    - **OptionName** (*string*) –
    - **OptionSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **OsOptionChargeInfo** (*dict*) –
  - \* **OsOptionChargeMonthlyRateSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **PatternAuthInfo** (*dict*) –
  - \* **PatternAuthSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **PremiumSupportInfo** (*dict*) –
  - \* **PremiumSupportSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **PrivateLanInfo** (*dict*) –
  - \* **PrivateLan** (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **PrivateNetworkInfo** (*dict*) –
  - \* **PrivateNetworkMeasuredRate** (*dict*) –
    - **Charge** (*integer*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
  - \* **PrivateNetworkMonthlyRate** (*dict*) –
    - **Charge** (*integer*) –
    - **Value** (*integer*) –
- **RemoteAccessVpnGatewayInfo** (*dict*) –
  - \* **RemoteAccessVpnGatewayMeasuredRateSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
  - \* **RemoteAccessVpnGatewayMonthlyRateSet** (*list*) –

- *(dict)* –
- **Charge** (*integer*) –
- **Type** (*string*) –
- **Unit** (*string*) –
- **Value** (*integer*) –
- **RequestId** (*string*) –
- **RouterInfo** (*dict*) –
  - \* **RouterMeasuredRateSet** (*list*) –
    - *(dict)* –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
  - \* **RouterMonthlyRateSet** (*list*) –
    - *(dict)* –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **SecurityGroupInfo** (*dict*) –
  - \* **OptionSet** (*list*) –
    - *(dict)* –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
  - \* **SecurityGroupApplyTime** (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **SnapshotInfo** (*dict*) –
  - \* **SnapShot** (*dict*) –
    - **Charge** (*integer*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **SslCertInfo** (*dict*) –
  - \* **CreateSslCertSet** (*list*) –
    - *(dict)* –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **VolumeInfo** (*dict*) –
  - \* **ImportInstanceDiskMeasuredRate** (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
  - \* **ImportInstanceDiskMonthlyRate** (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –

- \* **VolumeMeasuredRateSet** (*list*) –
  - (*dict*) –
  - **Charge** (*integer*) –
  - **Type** (*string*) –
  - **Unit** (*string*) –
  - **Value** (*integer*) –
- \* **VolumeSet** (*list*) –
  - (*dict*) –
  - **Charge** (*integer*) –
  - **Type** (*string*) –
  - **Unit** (*string*) –
  - **Value** (*integer*) –
- **VpnGatewayInfo** (*dict*) –
  - \* **VpnGatewayMeasuredRateSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
  - \* **VpnGatewayMonthlyRateSet** (*list*) –
    - (*dict*) –
    - **Charge** (*integer*) –
    - **Type** (*string*) –
    - **Unit** (*string*) –
    - **Value** (*integer*) –
- **YearMonth** (*string*) –

*computing* / Client / describe\_user\_activities

## describe\_user\_activities

`computing.Client.describe_user_activities(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_user_activities(  
    Range={  
        'EndNumber': 123,  
        'StartNumber': 123  
    },  
    YearMonth='string'  
)
```

### Parameters

- **Range** (*dict*) –
  - **EndNumber** (*integer*) –
  - **StartNumber** (*integer*) –
- **YearMonth** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
  'RequestId': 'string',
  'UserActivitiesSet': [
    {
      'CategoryName': 'string',
      'DateTime': datetime(2015, 1, 1),
      'IpAddress': 'string',
      'Operation': 'string',
      'Operator': 'string',
      'Result': True|False,
      'ServiceId': 'string',
      'Uuid': 'string'
    },
  ]
}
```

**Response Structure**

- (dict) –
  - **RequestId** (string) –
  - **UserActivitiesSet** (list) –
    - \* (dict) –
      - **CategoryName** (string) –
      - **DateTime** (datetime) –
      - **IpAddress** (string) –
      - **Operation** (string) –
      - **Operator** (string) –
      - **Result** (boolean) –
      - **ServiceId** (string) –
      - **Uuid** (string) –

*computing* / Client / describe\_volumes

**describe\_volumes**

`computing.Client.describe_volumes(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.describe_volumes(
    VolumeId=[
        'string',
    ]
)
```

**Parameters** **VolumeId** (list) –

- (string) –

**Return type** dict

**Returns**

**Response Syntax**

```
{
  'RequestId': 'string',
  'VolumeSet': [
    {
```

(continues on next page)

(continued from previous page)

```

        'AccountingType': 'string',
        'AttachmentSet': [
            {
                'AttachTime': 'string',
                'DeleteOnTermination': 'string',
                'Device': 'string',
                'InstanceId': 'string',
                'InstanceUniqueId': 'string',
                'Status': 'string',
                'VolumeId': 'string',
                'VolumeUniqueId': 'string'
            },
        ],
        'AvailabilityZone': 'string',
        'CreateTime': datetime(2015, 1, 1),
        'Description': 'string',
        'DiskType': 'string',
        'NextMonthAccountingType': 'string',
        'Size': 'string',
        'SnapshotId': 'string',
        'Status': 'string',
        'VolumeId': 'string',
        'VolumeUniqueId': 'string'
    },
]
}

```

**Response Structure**

- *(dict)* –
  - **RequestId** (*string*) –
  - **VolumeSet** (*list*) –
    - \* *(dict)* –
      - **AccountingType** (*string*) –
      - **AttachmentSet** (*list*) –
      - *(dict)* –
      - **AttachTime** (*string*) –
      - **DeleteOnTermination** (*string*) –
      - **Device** (*string*) –
      - **InstanceId** (*string*) –
      - **InstanceUniqueId** (*string*) –
      - **Status** (*string*) –
      - **VolumeId** (*string*) –
      - **VolumeUniqueId** (*string*) –
      - **AvailabilityZone** (*string*) –
      - **CreateTime** (*datetime*) –
      - **Description** (*string*) –
      - **DiskType** (*string*) –
      - **NextMonthAccountingType** (*string*) –
      - **Size** (*string*) –
      - **SnapshotId** (*string*) –
      - **Status** (*string*) –
      - **VolumeId** (*string*) –
      - **VolumeUniqueId** (*string*) –

*computing* / Client / describe\_vpn\_connections

## describe\_vpn\_connections

computing.Client.**describe\_vpn\_connections** (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```

response = client.describe_vpn_connections(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'customer-gateway-configuration'|'customer-gateway-id'|'nifty-
→customer-gateway-name'|'state'|'option.static-routes-only'|'route.destination-
→cidr-block'|'type'|'vpn-connection-id'|'vpn-gateway-id'|'nifty-vpn-gateway-name
→'|'nifty-vpn-connection-description'|'nifty-internet-key-exchange'
        },
    ],
    VpnConnectionId=[
        'string',
    ]
)

```

### Parameters

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
  - **VpnConnectionId** (*list*) –
    - (*string*) –

**Return type** dict

### Returns

### Response Syntax

```

{
    'RequestId': 'string',
    'VpnConnectionSet': [
        {
            'CreatedTime': datetime(2015, 1, 1),
            'CustomerGatewayConfiguration': 'string',
            'CustomerGatewayId': 'string',
            'NiftyCustomerGatewayName': 'string',
            'NiftyIpssecConfiguration': {
                'DiffieHellmanGroup': 123,
                'EncapsulatingSecurityPayloadLifetime': 123,
                'EncryptionAlgorithm': 'string',
                'HashingAlgorithm': 'string',
                'InternetKeyExchange': 'string',
                'InternetKeyExchangeLifetime': 123,
                'Mtu': 'string',
                'PreSharedKey': 'string'
            },
            'NiftyTunnel': {
                'DestinationPort': 'string',
                'Encapsulation': 'string',

```

(continues on next page)

(continued from previous page)

```

        'Mode': 'string',
        'PeerSessionId': 'string',
        'PeerTunnelId': 'string',
        'SessionId': 'string',
        'SourcePort': 'string',
        'TunnelId': 'string',
        'Type': 'string'
    },
    'NiftyVpnConnectionDescription': 'string',
    'NiftyVpnGatewayName': 'string',
    'State': 'string',
    'TagSet': [
        {
            'Key': 'string',
            'Value': 'string'
        },
    ],
    'Type': 'string',
    'VgwTelemetry': [
        {
            'AcceptedRouteCount': 123,
            'LastStatusChange': datetime(2015, 1, 1),
            'OutsideIpAddress': 'string',
            'Status': 'string',
            'StatusMessage': 'string'
        },
    ],
    'VpnConnectionId': 'string',
    'VpnGatewayId': 'string'
    },
]
}

```

### Response Structure

- (dict) –
  - **RequestId** (string) –
  - **VpnConnectionSet** (list) –
    - \* (dict) –
      - **CreatedTime** (datetime) –
      - **CustomerGatewayConfiguration** (string) –
      - **CustomerGatewayId** (string) –
      - **NiftyCustomerGatewayName** (string) –
      - **NiftyIpsecConfiguration** (dict) –
      - **DiffieHellmanGroup** (integer) –
      - **EncapsulatingSecurityPayloadLifetime** (integer) –
      - **EncryptionAlgorithm** (string) –
      - **HashingAlgorithm** (string) –
      - **InternetKeyExchange** (string) –
      - **InternetKeyExchangeLifetime** (integer) –
      - **Mtu** (string) –
      - **PreSharedKey** (string) –
      - **NiftyTunnel** (dict) –
      - **DestinationPort** (string) –
      - **Encapsulation** (string) –
      - **Mode** (string) –



- **PeerSessionId** (*string*) –
- **PeerTunnelId** (*string*) –
- **SessionId** (*string*) –
- **SourcePort** (*string*) –
- **TunnelId** (*string*) –
- **Type** (*string*) –
- **NiftyVpnConnectionDescription** (*string*) –
- **NiftyVpnGatewayName** (*string*) –
- **State** (*string*) –
- **TagSet** (*list*) –
- (*dict*) –
- **Key** (*string*) –
- **Value** (*string*) –
- **Type** (*string*) –
- **VgwTelemetry** (*list*) –
- (*dict*) –
- **AcceptedRouteCount** (*integer*) –
- **LastStatusChange** (*datetime*) –
- **OutsideIpAddress** (*string*) –
- **Status** (*string*) –
- **StatusMessage** (*string*) –
- **VpnConnectionId** (*string*) –
- **VpnGatewayId** (*string*) –

*computing* / Client / describe\_vpn\_gateways

## describe\_vpn\_gateways

`computing.Client.describe_vpn_gateways` (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_vpn_gateways(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'attachment.state'|'attachment.vpc-id'|'availability-zone'|
↪ 'state'|'type'|'vpn-gateway-id'|'nifty-vpn-gateway-name'|'nifty-vpn-gateway-type'
↪ '| 'nifty-vpn-gateway-description'|'nifty-vpn-gateway-accountingType'|'ip-address'
↪ '| 'latest-version-information'|'version'
        },
    ],
    NiftyVpnGatewayName=[
        'string',
    ],
    VpnGatewayId=[
        'string',
    ]
)
```

### Parameters

- **Filter** (*list*) –
- (*dict*) –

- \* **ListOfRequestValue** (*list*) –
  - (*string*) –
- \* **Name** (*string*) –
- **NiftyVpnGatewayName** (*list*) –
  - (*string*) –
- **VpnGatewayId** (*list*) –
  - (*string*) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'RequestId': 'string',
  'VpnGatewaySet': [
    {
      'AccountingType': 'string',
      'Attachments': 'string',
      'AvailabilityZone': 'string',
      'BackupInformation': {
        'ExpirationDate': datetime(2015, 1, 1),
        'IsBackup': True|False
      },
      'CreatedTime': datetime(2015, 1, 1),
      'GroupSet': [
        {
          'GroupId': 'string'
        },
      ],
    },
  ],
  'NetworkInterfaceSet': [
    {
      'CidrBlock': 'string',
      'Descriprion': 'string',
      'DeviceIndex': 'string',
      'IpAddress': 'string',
      'NetworkId': 'string',
      'NetworkName': 'string'
    },
  ],
  'NextMonthAccountingType': 'string',
  'NiftyRedundancy': True|False,
  'NiftyVpnGatewayDescription': 'string',
  'NiftyVpnGatewayName': 'string',
  'NiftyVpnGatewayType': 'string',
  'RouteTableAssociationId': 'string',
  'RouteTableId': 'string',
  'State': 'string',
  'TagSet': [
    {
      'Key': 'string',
      'Value': 'string'
    },
  ],
  'VersionInformation': {
    'IsLatest': True|False,
    'Version': 'string'
  },
  'VpnGatewayId': 'string'
}
```

(continues on next page)

(continued from previous page)

```

    },
  ]
}

```

**Response Structure**

- *(dict)* –
  - **RequestId** (*string*) –
  - **VpnGatewaySet** (*list*) –
    - \* *(dict)* –
      - **AccountingType** (*string*) –
      - **Attachments** (*string*) –
      - **AvailabilityZone** (*string*) –
      - **BackupInformation** (*dict*) –
      - **ExpirationDate** (*datetime*) –
      - **IsBackup** (*boolean*) –
      - **CreatedTime** (*datetime*) –
      - **GroupSet** (*list*) –
      - *(dict)* –
      - **GroupId** (*string*) –
      - **NetworkInterfaceSet** (*list*) –
      - *(dict)* –
      - **CidrBlock** (*string*) –
      - **Descripription** (*string*) –
      - **DeviceIndex** (*string*) –
      - **IpAddress** (*string*) –
      - **NetworkId** (*string*) –
      - **NetworkName** (*string*) –
      - **NextMonthAccountingType** (*string*) –
      - **NiftyRedundancy** (*boolean*) –
      - **NiftyVpnGatewayDescription** (*string*) –
      - **NiftyVpnGatewayName** (*string*) –
      - **NiftyVpnGatewayType** (*string*) –
      - **RouteTableAssociationId** (*string*) –
      - **RouteTableId** (*string*) –
      - **State** (*string*) –
      - **TagSet** (*list*) –
      - *(dict)* –
      - **Key** (*string*) –
      - **Value** (*string*) –
      - **VersionInformation** (*dict*) –
      - **IsLatest** (*boolean*) –
      - **Version** (*string*) –
      - **VpnGatewayId** (*string*) –

*computing* / Client / detach\_iso\_image

**detach\_iso\_image**

`computing.Client.detach_iso_image(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.detach_iso_image(  
    InstanceUniqueId='string',  
    IsoImageId='string'  
)
```

**Parameters**

- **InstanceUniqueId** (*string*) – [REQUIRED]
- **IsoImageId** (*string*) – [REQUIRED]

**Return type** dict**Returns****Response Syntax**

```
{  
    'RequestId': 'string',  
    'Return': True|False  
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / detach\_network\_interface**detach\_network\_interface**`computing.Client.detach_network_interface(**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.detach_network_interface(  
    AttachmentId='string',  
    NiftyReboot='force'|'true'|'false'  
)
```

**Parameters**

- **AttachmentId** (*string*) – [REQUIRED]
- **NiftyReboot** (*string*) –

**Return type** dict**Returns****Response Syntax**

```
{  
    'RequestId': 'string',  
    'Return': True|False  
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / detach\_volume

## detach\_volume

computing.Client.**detach\_volume** (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.detach_volume(
    Agreement=True|False,
    Device='string',
    Force=True|False,
    InstanceId='string',
    VolumeId='string'
)
```

### Parameters

- **Agreement** (*boolean*) –
- **Device** (*string*) –
- **Force** (*boolean*) –
- **InstanceId** (*string*) –
- **VolumeId** (*string*) – [REQUIRED]

Return type dict

### Returns

### Response Syntax

```
{
    'AttachTime': 'string',
    'Device': 'string',
    'InstanceId': 'string',
    'InstanceUniqueId': 'string',
    'RequestId': 'string',
    'Status': 'string',
    'VolumeId': 'string',
    'VolumeUniqueId': 'string'
}
```

### Response Structure

- (*dict*) –
  - **AttachTime** (*string*) –
  - **Device** (*string*) –
  - **InstanceId** (*string*) –
  - **InstanceUniqueId** (*string*) –
  - **RequestId** (*string*) –
  - **Status** (*string*) –
  - **VolumeId** (*string*) –
  - **VolumeUniqueId** (*string*) –

*computing* / Client / disassociate\_address

## disassociate\_address

computing.Client.**disassociate\_address** (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.disassociate_address(  
    AssociationId='string',  
    NiftyReboot='force'|'true'|'false',  
    PrivateIpAddress='string',  
    PublicIp='string'  
)
```

**Parameters**

- **AssociationId** (*string*) –
- **NiftyReboot** (*string*) –
- **PrivateIpAddress** (*string*) –
- **PublicIp** (*string*) –

**Return type** dict**Returns****Response Syntax**

```
{  
    'RequestId': 'string',  
    'Return': True|False  
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / `disassociate_multi_ip_address_group`**disassociate\_multi\_ip\_address\_group**`computing.Client.disassociate_multi_ip_address_group(**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.disassociate_multi_ip_address_group(  
    InstanceUniqueId='string',  
    MultiIpAddressGroupId='string',  
    NiftyReboot='force'|'true'|'false'  
)
```

**Parameters**

- **InstanceUniqueId** (*string*) – [REQUIRED]
- **MultiIpAddressGroupId** (*string*) – [REQUIRED]
- **NiftyReboot** (*string*) –

**Return type** dict**Returns****Response Syntax**

```
{  
    'RequestId': 'string',  
    'Return': True|False  
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / disassociate\_route\_table

## disassociate\_route\_table

`computing.Client.disassociate_route_table(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.disassociate_route_table(
    Agreement=True|False,
    AssociationId='string'
)
```

### Parameters

- **Agreement** (*boolean*) –
- **AssociationId** (*string*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / dissociate\_users

## dissociate\_users

`computing.Client.dissociate_users(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.dissociate_users(
    FunctionName='LB',
    Users=[
        {
            'UserId': 'string'
        },
    ]
)
```

### Parameters

- **FunctionName** (*string*) – [REQUIRED]
- **Users** (*list*) – [REQUIRED]

- (dict) –
  - \* **UserId** (string) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'DissociateUsersResult': {
    'Users': [
      {
        'UserId': 'string'
      },
    ]
  },
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

#### Response Structure

- (dict) –
  - **DissociateUsersResult** (dict) –
    - \* **Users** (list) –
      - (dict) –
        - **UserId** (string) –
  - **ResponseMetadata** (dict) –
    - \* **RequestId** (string) –

*computing* / Client / download\_ssl\_certificate

### download\_ssl\_certificate

`computing.Client.download_ssl_certificate(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.download_ssl_certificate(
    FileType='1'|'2'|'3',
    FqdnId='string'
)
```

#### Parameters

- **FileType** (string) – [REQUIRED]
- **FqdnId** (string) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'FileData': 'string',
  'Fqdn': 'string',
  'FqdnId': 'string',
  'RequestId': 'string'
}
```



**Response Structure**

- (*dict*) –
  - **FileData** (*string*) –
  - **Fqdn** (*string*) –
  - **FqdnId** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / `extend_volume_size`

**extend\_volume\_size**

`computing.Client.extend_volume_size(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.extend_volume_size(
    NiftyReboot='force'|'true'|'false',
    VolumeId='string'
)
```

**Parameters**

- **NiftyReboot** (*string*) –
- **VolumeId** (*string*) – [REQUIRED]

**Return type** dict

**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': 'string'
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*string*) –

*computing* / Client / `get_paginator`

**get\_paginator**

`computing.Client.get_paginator(operation_name)`

Create a paginator for an operation.

**Parameters** **operation\_name** (*string*) – The operation name. This is the same name as the method name on the client. For example, if the method name is `create_foo`, and you'd normally invoke the operation as `client.create_foo(**kwargs)`, if the `create_foo` operation can be paginated, you can use the call `client.get_paginator("create_foo")`.

**Raises** **OperationNotPageableError** – Raised if the operation is not pageable. You can use the `client.can_paginate` method to check if an operation is pageable.

**Return type** L{botocore.paginator.Paginator}

**Returns** A paginator object.

*computing* / Client / `get_waiter`

## get\_waiter

`computing.Client.get_waiter(waiter_name)`

Returns an object that can wait for some condition.

**Parameters** `waiter_name` (*str*) – The name of the waiter to get. See the waiters section of the service docs for a list of available waiters.

**Returns** The specified waiter object.

**Return type** `botocore.waiter.Waiter`

*computing* / *Client* / *import\_instance*

## import\_instance

`computing.Client.import_instance(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```

response = client.import_instance(
    AccountingType='1'|'2',
    Architecture='string',
    Description='string',
    DiskImage=[
        {
            'Description': 'string',
            'RequestImage': {
                'Bytes': 123,
                'Format': 'string',
                'ImportManifestUrl': 'string'
            },
            'RequestVolume': {
                'Size': 123
            }
        },
    ],
    InstanceId='string',
    InstanceInitiatedShutdownBehavior='string',
    InstanceType='e-mini'|'h2-mini'|'mini'|'c-small'|'e-small'|'h2-small'|'small'|
    ↪ 'c-small2'|'e-small2'|'h2-small2'|'small2'|'c-small4'|'e-small4'|'h2-small4'|
    ↪ 'small4'|'e-small8'|'h2-small8'|'small8'|'e-small16'|'h2-small16'|'small16'|'c-
    ↪ medium'|'e-medium'|'h2-medium'|'medium'|'c-medium4'|'e-medium4'|'h2-medium4'|
    ↪ 'medium4'|'c-medium8'|'e-medium8'|'h2-medium8'|'medium8'|'e-medium16'|'h2-
    ↪ medium16'|'e-medium16'|'h2-medium16'|'h2-medium24'|'medium24'|'c-large'|'e-large'|
    ↪ 'h2-large'|'large'|'c-large8'|'e-large8'|'h2-large8'|'large8'|'e-large16'|'h2-
    ↪ large16'|'large16'|'e-large24'|'h2-large24'|'large24'|'e-large32'|'h2-large32'|
    ↪ 'large32'|'e-extra-large8'|'h2-extra-large8'|'extra-large8'|'e-extra-large16'|
    ↪ 'h2-extra-large16'|'extra-large16'|'e-extra-large24'|'h2-extra-large24'|'extra-
    ↪ large24'|'e-extra-large32'|'h2-extra-large32'|'extra-large32'|'e-extra-large48'|
    ↪ 'h2-extra-large48'|'extra-large48'|'e-double-large16'|'h2-double-large16'|
    ↪ 'double-large16'|'e-double-large24'|'h2-double-large24'|'double-large24'|'e-
    ↪ double-large32'|'h2-double-large32'|'double-large32'|'e-double-large48'|'h2-
    ↪ double-large48'|'double-large48'|'e-double-large64'|'h2-double-large64'|'double-
    ↪ large64'|'e-double-large96'|'h2-double-large96'|'double-large96'|'h2-triple-
    ↪ large32'|'triple-large32'|'h2-triple-large48'|'triple-large48'|'h2-triple-
    ↪ large64'|'triple-large64'|'h2-triple-large96'|'triple-large96'|'h2-triple-
    ↪ large128'|'triple-large128'|'h2-quad-large64'|'quad-large64'|'h2-quad-large96'|
    ↪ 'quad-large96'|'h2-quad-large128'|'quad-large128'|'h2-septa-large128'|'septa-
    ↪ large128',

```

(continues on next page)

(continued from previous page)

```

IpType='static'|'elastic'|'none',
Monitoring={
    'Enabled': 'string'
},
NetworkInterface=[
    {
        'DeviceIndex': 123,
        'IpAddress': 'string',
        'ListOfRequestSecurityGroupId': [
            'string',
        ],
        'NetworkId': 'string',
        'NetworkName': 'string'
    },
],
Ovf='string',
Placement={
    'AvailabilityZone': 'string'
},
Platform='string',
PrivateIpAddress='string',
PublicIp='string',
SecurityGroup=[
    'string',
],
SubnetId='string',
UserData={
    'Content': 'string'
}
)

```

### Parameters

- **AccountingType** (*string*) –
- **Architecture** (*string*) –
- **Description** (*string*) –
- **DiskImage** (*list*) –
  - (*dict*) –
    - \* **Description** (*string*) –
    - \* **RequestImage** (*dict*) –
      - **Bytes** (*integer*) –
      - **Format** (*string*) –
      - **ImportManifestUrl** (*string*) –
    - \* **RequestVolume** (*dict*) –
      - **Size** (*integer*) –
- **InstanceId** (*string*) –
- **InstanceInitiatedShutdownBehavior** (*string*) –
- **InstanceType** (*string*) –
- **IpType** (*string*) –
- **Monitoring** (*dict*) –
  - **Enabled** (*string*) –
- **NetworkInterface** (*list*) –
  - (*dict*) –
    - \* **DeviceIndex** (*integer*) –
    - \* **IpAddress** (*string*) –
    - \* **ListOfRequestSecurityGroupId** (*list*) –

- (string) –
- \* **NetworkId** (string) –
- \* **NetworkName** (string) –
- **Ovf** (string) – **[REQUIRED]**
- **Placement** (dict) –
  - **AvailabilityZone** (string) –
- **Platform** (string) –
- **PrivateIpAddress** (string) –
- **PublicIp** (string) –
- **SecurityGroup** (list) –
  - (string) –
- **SubnetId** (string) –
- **UserData** (dict) –
  - **Content** (string) –

**Return type** dict

**Returns**

### Response Syntax

```
{
  'ConversionTask': {
    'ConversionTaskId': 'string',
    'ExpirationTime': 'string',
    'ImportInstance': {
      'Description': 'string',
      'InstanceId': 'string',
      'InstanceUniqueId': 'string',
      'Volumes': [
        {
          'AvailabilityZone': 'string',
          'BytesConverted': 123,
          'Image': {
            'Format': 'string',
            'Size': 123
          },
          'Status': 'string'
        },
      ],
    },
  },
  'NetworkInterfaceSet': [
    {
      'Association': {
        'IpOwnerId': 'string',
        'PublicDnsName': 'string'
      },
      'Attachment': {
        'AttachTime': 'string',
        'AttachmentID': 'string',
        'DeleteOnTermination': 'string',
        'DeviceIndex': 'string',
        'Status': 'string'
      },
      'Description': 'string',
      'GroupSet': 'string',
      'NetworkInterfaceId': 'string',
      'NiftyNetworkId': 'string',
      'NiftyNetworkName': 'string',
```

(continues on next page)

(continued from previous page)

```

        'OwnerId': 'string',
        'PrivateDnsName': 'string',
        'PrivateIpAddressesSet': 'string',
        'SourceDestCheck': 'string',
        'Status': 'string',
        'SubnetId': 'string',
        'VpcId': 'string'
    },
],
'State': 'string'
}

```

**Response Structure**

- *(dict)* –
  - **ConversionTask** (*dict*) –
    - \* **ConversionTaskId** (*string*) –
    - \* **ExpirationTime** (*string*) –
    - \* **ImportInstance** (*dict*) –
      - **Description** (*string*) –
      - **InstanceId** (*string*) –
      - **InstanceUniqueId** (*string*) –
      - **Volumes** (*list*) –
      - (*dict*) –
      - **AvailabilityZone** (*string*) –
      - **BytesConverted** (*integer*) –
      - **Image** (*dict*) –
      - **Format** (*string*) –
      - **Size** (*integer*) –
      - **Status** (*string*) –
    - \* **NetworkInterfaceSet** (*list*) –
      - (*dict*) –
      - **Association** (*dict*) –
      - **IpOwnerId** (*string*) –
      - **PublicDnsName** (*string*) –
      - **Attachment** (*dict*) –
      - **AttachTime** (*string*) –
      - **AttachmentID** (*string*) –
      - **DeleteOnTermination** (*string*) –
      - **DeviceIndex** (*string*) –
      - **Status** (*string*) –
      - **Description** (*string*) –
      - **GroupSet** (*string*) –
      - **NetworkInterfaceId** (*string*) –
      - **NiftyNetworkId** (*string*) –
      - **NiftyNetworkName** (*string*) –
      - **OwnerId** (*string*) –
      - **PrivateDnsName** (*string*) –
      - **PrivateIpAddressesSet** (*string*) –
      - **SourceDestCheck** (*string*) –
      - **Status** (*string*) –
      - **SubnetId** (*string*) –
      - **VpcId** (*string*) –
    - \* **State** (*string*) –

*computing* / Client / import\_key\_pair

## import\_key\_pair

`computing.Client.import_key_pair(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.import_key_pair(
    Description='string',
    KeyName='string',
    PublicKeyMaterial='string'
)
```

### Parameters

- **Description** (*string*) –
- **KeyName** (*string*) – [REQUIRED]
- **PublicKeyMaterial** (*string*) – [REQUIRED]

**Return type** dict

### Returns

#### Response Syntax

```
{
    'KeyFingerprint': 'string',
    'KeyName': 'string',
    'RequestId': 'string'
}
```

### Response Structure

- (*dict*) –
  - **KeyFingerprint** (*string*) –
  - **KeyName** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / increase\_multi\_ip\_address\_count

## increase\_multi\_ip\_address\_count

`computing.Client.increase_multi_ip_address_count(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.increase_multi_ip_address_count(
    IpAddressCount=123,
    MultiIpAddressGroupId='string'
)
```

### Parameters

- **IpAddressCount** (*integer*) – [REQUIRED]
- **MultiIpAddressGroupId** (*string*) – [REQUIRED]

**Return type** dict

### Returns

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (dict) –
  - **RequestId** (string) –
  - **Return** (boolean) –

*computing* / Client / modify\_image\_attribute

### modify\_image\_attribute

`computing.Client.modify_image_attribute(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.modify_image_attribute(
    Attribute='description'|'imageName'|'niftyContactUrl'|'detailDescription',
    ImageId='string',
    LaunchPermission={
        'ListOfRequestAdd': [
            {
                'Group': 'string',
                'UserId': 'string'
            },
        ],
        'ListOfRequestRemove': [
            {
                'Group': 'string',
                'UserId': 'string'
            },
        ],
    },
    ProductCode=[
        'string',
    ],
    Value='string'
)
```

#### Parameters

- **Attribute** (string) –
- **ImageId** (string) – [REQUIRED]
- **LaunchPermission** (dict) –
  - **ListOfRequestAdd** (list) –
    - \* (dict) –
      - **Group** (string) –
      - **UserId** (string) –
  - **ListOfRequestRemove** (list) –
    - \* (dict) –
      - **Group** (string) –
      - **UserId** (string) –
- **ProductCode** (list) –
  - (string) –

- **Value** (*string*) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / modify\_instance\_attribute

### modify\_instance\_attribute

`computing.Client.modify_instance_attribute(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.modify_instance_attribute(
    Attribute='instanceType'|'disableApiTermination'|'instanceName'|'description'|
    ↪ 'ipType'|'groupId'|'accountingType',
    Force=True|False,
    InstanceId='string',
    NiftyReboot='force'|'true'|'false',
    Tenancy='default'|'dedicated',
    Value='string'
)
```

#### Parameters

- **Attribute** (*string*) – [REQUIRED]
- **Force** (*boolean*) –
- **InstanceId** (*string*) – [REQUIRED]
- **NiftyReboot** (*string*) –
- **Tenancy** (*string*) –
- **Value** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / modify\_instance\_backup\_rule\_attribute



## modify\_instance\_backup\_rule\_attribute

`computing.Client.modify_instance_backup_rule_attribute(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.modify_instance_backup_rule_attribute(
    BackupInstanceMaxCount=123,
    Description='string',
    InstanceBackupRuleId='string',
    InstanceBackupRuleName='string',
    TimeSlotId='1'|'2'|'3'|'4'|'5'|'6'|'7'|'8'|'9'|'10'|'11'|'12'
)
```

### Parameters

- **BackupInstanceMaxCount** (*integer*) –
- **Description** (*string*) –
- **InstanceBackupRuleId** (*string*) – [REQUIRED]
- **InstanceBackupRuleName** (*string*) –
- **TimeSlotId** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / `modify_multi_ip_address_group_attribute`

## modify\_multi\_ip\_address\_group\_attribute

`computing.Client.modify_multi_ip_address_group_attribute(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.modify_multi_ip_address_group_attribute(
    Description='string',
    MultiIpAddressGroupId='string',
    MultiIpAddressGroupName='string'
)
```

### Parameters

- **Description** (*string*) –
- **MultiIpAddressGroupId** (*string*) – [REQUIRED]
- **MultiIpAddressGroupName** (*string*) –

**Return type** dict

## Returns

### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / `modify_network_interface_attribute`

## `modify_network_interface_attribute`

`computing.Client.modify_network_interface_attribute(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.modify_network_interface_attribute(
    Description='string',
    IpAddress='string',
    NetworkInterfaceId='string'
)
```

### Parameters

- **Description** (*string*) –
- **IpAddress** (*string*) –
- **NetworkInterfaceId** (*string*) – [REQUIRED]

**Return type** dict

### Returns

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / `modify_remote_access_vpn_gateway_attribute`

## `modify_remote_access_vpn_gateway_attribute`

`computing.Client.modify_remote_access_vpn_gateway_attribute(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.modify_remote_access_vpn_gateway_attribute(
    AccountingType='1'|'2',
    ClientTunnelMode='split'|'full',
    Description='string',
    RemoteAccessVpnGatewayId='string',
    RemoteAccessVpnGatewayName='string',
    RemoteAccessVpnGatewayType='small'|'medium'|'large'
)
```

**Parameters**

- **AccountingType** (*string*) –
- **ClientTunnelMode** (*string*) –
- **Description** (*string*) –
- **RemoteAccessVpnGatewayId** (*string*) – [REQUIRED]
- **RemoteAccessVpnGatewayName** (*string*) –
- **RemoteAccessVpnGatewayType** (*string*) –

**Return type** dict

**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / modify\_remote\_access\_vpn\_gateway\_user\_attribute

**modify\_remote\_access\_vpn\_gateway\_user\_attribute**

`computing.Client.modify_remote_access_vpn_gateway_user_attribute` (*\*\*kwargs*)

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.modify_remote_access_vpn_gateway_user_attribute(
    Description='string',
    Password='string',
    RemoteAccessVpnGatewayId='string',
    UserName='string'
)
```

**Parameters**

- **Description** (*string*) –
- **Password** (*string*) –
- **RemoteAccessVpnGatewayId** (*string*) – [REQUIRED]
- **UserName** (*string*) – [REQUIRED]

**Return type** dict

**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / `modify_ssl_certificate_attribute`

**modify\_ssl\_certificate\_attribute**

`computing.Client.modify_ssl_certificate_attribute(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.modify_ssl_certificate_attribute(
    Description={
        'Value': 'string'
    },
    FqdnId='string'
)
```

**Parameters**

- **Description** (*dict*) –
  - **Value** (*string*) –
- **FqdnId** (*string*) – [REQUIRED]

**Return type** `dict`

**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / `modify_volume_attribute`

**modify\_volume\_attribute**

`computing.Client.modify_volume_attribute(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.modify_volume_attribute(
    Attribute='accountingType'|'volumeName'|'description',
    Value='string',
    VolumeId='string'
)
```

**Parameters**

- **Attribute** (*string*) –
- **Value** (*string*) –
- **VolumeId** (*string*) – [REQUIRED]

**Return type** dict**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_associate\_image**nifty\_associate\_image**computing.Client.**nifty\_associate\_image** (\*\*kwargs)See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.nifty_associate_image(
    DistributionId=[
        'string',
    ],
    ImageId='string',
    IsPublic=True|False,
    IsRedistribute=True|False
)
```

**Parameters**

- **DistributionId** (*list*) –
  - (*string*) –
- **ImageId** (*string*) – [REQUIRED]
- **IsPublic** (*boolean*) – [REQUIRED]
- **IsRedistribute** (*boolean*) –

**Return type** dict**Returns****Response Syntax**

```
{
    'RequestId': 'string',
```

(continues on next page)

(continued from previous page)

```
'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_associate\_nat\_table**nifty\_associate\_nat\_table**`computing.Client.nifty_associate_nat_table(**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.nifty_associate_nat_table(
    Agreement=True|False,
    NatTableId='string',
    RouterId='string',
    RouterName='string'
)
```

**Parameters**

- **Agreement** (*boolean*) –
- **NatTableId** (*string*) – [REQUIRED]
- **RouterId** (*string*) –
- **RouterName** (*string*) –

**Return type** dict**Returns****Response Syntax**

```
{
    'AssociationId': 'string',
    'RequestId': 'string'
}
```

**Response Structure**

- (*dict*) –
  - **AssociationId** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / nifty\_associate\_route\_table\_with\_elastic\_load\_balancer**nifty\_associate\_route\_table\_with\_elastic\_load\_balancer**`computing.Client.nifty_associate_route_table_with_elastic_load_balancer(**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.nifty_associate_route_table_with_elastic_load_balancer(
    ElasticLoadBalancerId='string',
    RouteTableId='string'
)
```

**Parameters**

- **ElasticLoadBalancerId** (*string*) – [REQUIRED]
- **RouteTableId** (*string*) – [REQUIRED]

**Return type** dict**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_associate\_route\_table\_with\_vpn\_gateway**nifty\_associate\_route\_table\_with\_vpn\_gateway**`computing.Client.nifty_associate_route_table_with_vpn_gateway(**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.nifty_associate_route_table_with_vpn_gateway(
    Agreement=True|False,
    NiftyVpnGatewayName='string',
    RouteTableId='string',
    VpnGatewayId='string'
)
```

**Parameters**

- **Agreement** (*boolean*) –
- **NiftyVpnGatewayName** (*string*) –
- **RouteTableId** (*string*) – [REQUIRED]
- **VpnGatewayId** (*string*) –

**Return type** dict**Returns****Response Syntax**

```
{
    'AssociationId': 'string',
    'RequestId': 'string'
}
```

**Response Structure**

- (*dict*) –
  - **AssociationId** (*string*) –

– **RequestId** (*string*) –

*computing* / Client / nifty\_configure\_elastic\_load\_balancer\_health\_check

## nifty\_configure\_elastic\_load\_balancer\_health\_check

`computing.Client.nifty_configure_elastic_load_balancer_health_check(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_configure_elastic_load_balancer_health_check(
    ElasticLoadBalancerId='string',
    ElasticLoadBalancerName='string',
    ElasticLoadBalancerPort=123,
    HealthCheck={
        'Interval': 123,
        'ListOfRequestExpectation': [
            {
                'HttpCode': 'string'
            },
        ],
        'Path': 'string',
        'Target': 'string',
        'UnhealthyThreshold': 123
    },
    InstancePort=123,
    Protocol='TCP' | 'UDP' | 'HTTP' | 'HTTPS'
)
```

### Parameters

- **ElasticLoadBalancerId** (*string*) –
- **ElasticLoadBalancerName** (*string*) –
- **ElasticLoadBalancerPort** (*integer*) – [REQUIRED]
- **HealthCheck** (*dict*) – [REQUIRED]
  - **Interval** (*integer*) – [REQUIRED]
  - **ListOfRequestExpectation** (*list*) –
    - \* (*dict*) –
      - **HttpCode** (*string*) –
  - **Path** (*string*) –
  - **Target** (*string*) – [REQUIRED]
  - **UnhealthyThreshold** (*integer*) – [REQUIRED]
- **InstancePort** (*integer*) – [REQUIRED]
- **Protocol** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
    'NiftyConfigureElasticLoadBalancerHealthCheckResult': {
        'HealthCheck': {
            'Expectation': [
                {
                    'HttpCode': 'string'
                },
            ],
        },
    },
}
```

(continues on next page)



(continued from previous page)

```

        ],
        'Interval': 123,
        'Path': 'string',
        'Target': 'string',
        'UnhealthyThreshold': 123
    }
},
'ResponseMetadata': {
    'RequestId': 'string'
}
}

```

**Response Structure**

- (dict) –
  - **NiftyConfigureElasticLoadBalancerHealthCheckResult** (dict) –
    - \* **HealthCheck** (dict) –
      - **Expectation** (list) –
      - (dict) –
      - **HttpCode** (string) –
      - **Interval** (integer) –
      - **Path** (string) –
      - **Target** (string) –
      - **UnhealthyThreshold** (integer) –
  - **ResponseMetadata** (dict) –
    - \* **RequestId** (string) –

*computing* / Client / nifty\_create\_alarm**nifty\_create\_alarm**computing.Client.**nifty\_create\_alarm**(\*\*kwargs)See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```

response = client.nifty_create_alarm(
    AlarmCondition='and'|'or',
    Description='string',
    ElasticLoadBalancerName=[
        'string',
    ],
    ElasticLoadBalancerPort=[
        123,
    ],
    ElasticLoadBalancerProtocol=[
        'string',
    ],
    EmailAddress=[
        'string',
    ],
    FunctionName='Server'|'LoadBalancer'|'DiskPartition'|'ElasticLoadBalancer',
    InstanceId=[
        'string',
    ],
    LoadBalancerName=[

```

(continues on next page)

(continued from previous page)

```

        'string',
    ],
    LoadBalancerPort=[
        123,
    ],
    Partition=[
        'string',
    ],
    Rule=[
        {
            'BreachDuration': 123,
            'DataType': 'string',
            'Threshold': 123.0,
            'UpperLowerCondition': 'upper' | 'lower'
        },
    ],
    RuleName='string',
    Zone='string'
)

```

**Parameters**

- **AlarmCondition** (*string*) –
- **Description** (*string*) –
- **ElasticLoadBalancerName** (*list*) –  
– (*string*) –
- **ElasticLoadBalancerPort** (*list*) –  
– (*integer*) –
- **ElasticLoadBalancerProtocol** (*list*) –  
– (*string*) –
- **EmailAddress** (*list*) – [REQUIRED]  
– (*string*) –
- **FunctionName** (*string*) – [REQUIRED]
- **InstanceId** (*list*) –  
– (*string*) –
- **LoadBalancerName** (*list*) –  
– (*string*) –
- **LoadBalancerPort** (*list*) –  
– (*integer*) –
- **Partition** (*list*) –  
– (*string*) –
- **Rule** (*list*) – [REQUIRED]  
– (*dict*) –  
    \* **BreachDuration** (*integer*) – [REQUIRED]  
    \* **DataType** (*string*) – [REQUIRED]  
    \* **Threshold** (*float*) –  
    \* **UpperLowerCondition** (*string*) –
- **RuleName** (*string*) –
- **Zone** (*string*) –

**Return type** dict**Returns****Response Syntax**

```
{
```

(continues on next page)

(continued from previous page)

```

    'RequestId': 'string',
    'Return': True|False
}

```

**Response Structure**

- (dict) –
  - **RequestId** (string) –
  - **Return** (boolean) –

*computing* / Client / nifty\_create\_auto\_scaling\_group

**nifty\_create\_auto\_scaling\_group**

`computing.Client.nifty_create_auto_scaling_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```

response = client.nifty_create_auto_scaling_group(
    AutoScalingGroupName='string',
    ChangeInCapacity=123,
    DefaultCooldown=123,
    Description='string',
    ImageId='string',
    InstanceLifecycleLimit=123,
    InstanceType='e-mini'|'h2-mini'|'mini'|'c-small'|'e-small'|'h2-small'|'small'|
↪ 'c-small12'|'e-small12'|'h2-small12'|'small12'|'c-small14'|'e-small14'|'h2-small14'|
↪ 'small14'|'e-small18'|'h2-small18'|'small18'|'e-small116'|'h2-small116'|'small116'|'c-
↪ medium'|'e-medium'|'h2-medium'|'medium'|'c-medium4'|'e-medium4'|'h2-medium4'|
↪ 'medium4'|'c-medium8'|'e-medium8'|'h2-medium8'|'medium8'|'e-medium16'|'h2-
↪ medium16'|'c-medium16'|'e-medium24'|'h2-medium24'|'medium24'|'c-large'|'e-large'|
↪ 'h2-large'|'large'|'c-large8'|'e-large8'|'h2-large8'|'large8'|'e-large16'|'h2-
↪ large16'|'large16'|'e-large24'|'h2-large24'|'large24'|'e-large32'|'h2-large32'|
↪ 'large32'|'e-extra-large8'|'h2-extra-large8'|'extra-large8'|'e-extra-large16'|
↪ 'h2-extra-large16'|'extra-large16'|'e-extra-large24'|'h2-extra-large24'|'extra-
↪ large24'|'e-extra-large32'|'h2-extra-large32'|'extra-large32'|'e-extra-large48'|
↪ 'h2-extra-large48'|'extra-large48'|'e-double-large16'|'h2-double-large16'|
↪ 'double-large16'|'e-double-large24'|'h2-double-large24'|'double-large24'|'e-
↪ double-large32'|'h2-double-large32'|'double-large32'|'e-double-large48'|'h2-
↪ double-large48'|'double-large48'|'e-double-large64'|'h2-double-large64'|'double-
↪ large64'|'e-double-large96'|'h2-double-large96'|'double-large96'|'h2-triple-
↪ large32'|'triple-large32'|'h2-triple-large48'|'triple-large48'|'h2-triple-
↪ large64'|'triple-large64'|'h2-triple-large96'|'triple-large96'|'h2-triple-
↪ large128'|'triple-large128'|'h2-quad-large64'|'quad-large64'|'h2-quad-large96'|
↪ 'quad-large96'|'h2-quad-large128'|'quad-large128'|'h2-septa-large128'|'septa-
↪ large128',
    LoadBalancers=[
        {
            'InstancePort': 123,
            'LoadBalancerPort': 123,
            'Name': 'string'
        },
    ],
    MaxSize=123,
    MinSize=123,
    Scaleout=123,

```

(continues on next page)

(continued from previous page)

```

ScaleoutCondition='or'|'and',
ScalingSchedule=[
    {
        'RequestDDay': {
            'EndingDDay': 'string',
            'StartingDDay': 'string'
        },
        'RequestDay': {
            'SetFriday': '0'|'1',
            'SetMonday': '0'|'1',
            'SetSaturday': '0'|'1',
            'SetSunday': '0'|'1',
            'SetThursday': '0'|'1',
            'SetTuesday': '0'|'1',
            'SetWednesday': '0'|'1'
        },
        'RequestMonth': {
            'EndingMonth': 'string',
            'StartingMonth': 'string'
        },
        'RequestTimeZone': {
            'EndingTimeZone': 'string',
            'StartingTimeZone': 'string'
        }
    },
],
ScalingTrigger=[
    {
        'BreachDuration': 123,
        'Resource': 'Server-cpu'|'Server-memory'|'Server-network'|
↪ 'LoadBalancer-network',
        'UpperThreshold': 123.0
    },
],
SecurityGroup=[
    'string',
]
)

```

### Parameters

- **AutoScalingGroupName** (*string*) – [REQUIRED]
- **ChangeInCapacity** (*integer*) – [REQUIRED]
- **DefaultCooldown** (*integer*) –
- **Description** (*string*) –
- **ImageId** (*string*) – [REQUIRED]
- **InstanceLifecycleLimit** (*integer*) –
- **InstanceType** (*string*) –
- **LoadBalancers** (*list*) –
  - (*dict*) –
    - \* **InstancePort** (*integer*) –
    - \* **LoadBalancerPort** (*integer*) –
    - \* **Name** (*string*) –
- **MaxSize** (*integer*) – [REQUIRED]
- **MinSize** (*integer*) – [REQUIRED]
- **Scaleout** (*integer*) –
- **ScaleoutCondition** (*string*) – [REQUIRED]

- **ScalingSchedule** (*list*) –
  - (*dict*) –
    - \* **RequestDDay** (*dict*) –
      - **EndingDDay** (*string*) –
      - **StartingDDay** (*string*) –
    - \* **RequestDay** (*dict*) –
      - **SetFriday** (*string*) –
      - **SetMonday** (*string*) –
      - **SetSaturday** (*string*) –
      - **SetSunday** (*string*) –
      - **SetThursday** (*string*) –
      - **SetTuesday** (*string*) –
      - **SetWednesday** (*string*) –
    - \* **RequestMonth** (*dict*) –
      - **EndingMonth** (*string*) –
      - **StartingMonth** (*string*) –
    - \* **RequestTimeZone** (*dict*) –
      - **EndingTimeZone** (*string*) –
      - **StartingTimeZone** (*string*) –
  - **ScalingTrigger** (*list*) – [REQUIRED]
    - (*dict*) –
      - \* **BreachDuration** (*integer*) –
      - \* **Resource** (*string*) – [REQUIRED]
      - \* **UpperThreshold** (*float*) – [REQUIRED]
  - **SecurityGroup** (*list*) –
    - (*string*) –

Return type dict

Returns

#### Response Syntax

```
{
  'RequestId': 'string',
  'Return': True|False
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_create\_dhcp\_config

### nifty\_create\_dhcp\_config

`computing.Client.nifty_create_dhcp_config()`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.nifty_create_dhcp_config()
```

Return type dict

Returns

#### Response Syntax

```
{
    'DhcpConfig': {
        'DhcpConfigId': 'string'
    },
    'RequestId': 'string'
}
```

**Response Structure**

- (*dict*) –
  - **DhcpConfig** (*dict*) –
    - \* **DhcpConfigId** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / `nifty_create_dhcp_ip_address_pool`

**nifty\_create\_dhcp\_ip\_address\_pool**

`computing.Client.nifty_create_dhcp_ip_address_pool` (*\*\*kwargs*)

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.nifty_create_dhcp_ip_address_pool(
    Description='string',
    DhcpConfigId='string',
    StartIpAddress='string',
    StopIpAddress='string'
)
```

**Parameters**

- **Description** (*string*) –
- **DhcpConfigId** (*string*) – [REQUIRED]
- **StartIpAddress** (*string*) – [REQUIRED]
- **StopIpAddress** (*string*) – [REQUIRED]

**Return type** dict

**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / `nifty_create_dhcp_static_mapping`

**nifty\_create\_dhcp\_static\_mapping**

`computing.Client.nifty_create_dhcp_static_mapping` (*\*\*kwargs*)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_create_dhcp_static_mapping(
    Description='string',
    DhcpConfigId='string',
    IpAddress='string',
    MacAddress='string'
)
```

#### Parameters

- **Description** (*string*) –
- **DhcpConfigId** (*string*) – [REQUIRED]
- **IpAddress** (*string*) – [REQUIRED]
- **MacAddress** (*string*) – [REQUIRED]

Return type dict

#### Returns

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_create\_elastic\_load\_balancer

### nifty\_create\_elastic\_load\_balancer

`computing.Client.nifty_create_elastic_load_balancer(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.nifty_create_elastic_load_balancer(
    AccountingType='1'|'2',
    AvailabilityZones=[
        'string',
    ],
    ElasticLoadBalancerName='string',
    Listeners=[
        {
            'BalancingType': 123,
            'Description': 'string',
            'ElasticLoadBalancerPort': 123,
            'InstancePort': 123,
            'ListOfRequestInstances': [
                {
                    'InstanceId': 'string',
                    'InstanceUniqueId': 'string'
                },
            ],
            'Protocol': 'TCP'|'UDP'|'HTTP'|'HTTPS',
```

(continues on next page)

(continued from previous page)

```

        'RequestHealthCheck': {
            'Interval': 123,
            'ListOfRequestExpectation': [
                {
                    'HttpCode': '1xx'|'2xx'|'3xx'|'4xx'|'5xx'
                },
            ],
            'Path': 'string',
            'Target': 'string',
            'UnhealthyThreshold': 123
        },
        'RequestSession': {
            'RequestStickinessPolicy': {
                'Enable': True|False,
                'ExpirationPeriod': 123,
                'Method': '1'|'2'
            }
        },
        'RequestSorryPage': {
            'Enable': True|False,
            'RedirectUrl': 'string'
        },
        'SSLCertificateId': 'string'
    },
],
NetworkInterface=[
    {
        'IpAddress': 'string',
        'IsVipNetwork': True|False,
        'ListOfRequestSystemIpAddresses': [
            {
                'SystemIpAddress': 'string'
            },
        ],
        'NetworkId': 'string',
        'NetworkName': 'string'
    },
],
NetworkVolume=123
)

```

### Parameters

- **AccountingType** (*string*) –
- **AvailabilityZones** (*list*) – [REQUIRED]
  - (*string*) –
- **ElasticLoadBalancerName** (*string*) –
- **Listeners** (*list*) – [REQUIRED]
  - (*dict*) –
    - \* **BalancingType** (*integer*) –
    - \* **Description** (*string*) –
    - \* **ElasticLoadBalancerPort** (*integer*) – [REQUIRED]
    - \* **InstancePort** (*integer*) –
    - \* **ListOfRequestInstances** (*list*) –
      - (*dict*) –
      - **InstanceId** (*string*) –
      - **InstanceUniqueId** (*string*) –



- \* **Protocol** (*string*) – [REQUIRED]
- \* **RequestHealthCheck** (*dict*) –
  - **Interval** (*integer*) –
  - **ListOfRequestExpectation** (*list*) –
  - (*dict*) –
  - **HttpCode** (*string*) –
  - **Path** (*string*) –
  - **Target** (*string*) –
  - **UnhealthyThreshold** (*integer*) –
- \* **RequestSession** (*dict*) –
  - **RequestStickinessPolicy** (*dict*) –
  - **Enable** (*boolean*) –
  - **ExpirationPeriod** (*integer*) –
  - **Method** (*string*) –
- \* **RequestSorryPage** (*dict*) –
  - **Enable** (*boolean*) –
  - **RedirectUrl** (*string*) –
- \* **SSLCertificateId** (*string*) –
- **NetworkInterface** (*list*) –
  - (*dict*) –
    - \* **IpAddress** (*string*) –
    - \* **IsVipNetwork** (*boolean*) –
    - \* **ListOfRequestSystemIpAddresses** (*list*) –
      - (*dict*) –
      - **SystemIpAddress** (*string*) –
    - \* **NetworkId** (*string*) –
    - \* **NetworkName** (*string*) –
- **NetworkVolume** (*integer*) –

Return type dict

Returns

### Response Syntax

```
{
  'NiftyCreateElasticLoadBalancerResult': {
    'DNSName': 'string'
  },
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

### Response Structure

- (*dict*) –
  - **NiftyCreateElasticLoadBalancerResult** (*dict*) –
    - \* **DNSName** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*computing* / Client / nifty\_create\_instance\_snapshot

## nifty\_create\_instance\_snapshot

`computing.Client.nifty_create_instance_snapshot (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_create_instance_snapshot(  
    Description='string',  
    InstanceId='string',  
    SnapshotName='string'  
)
```

#### Parameters

- **Description** (*string*) –
- **InstanceId** (*string*) – [REQUIRED]
- **SnapshotName** (*string*) – [REQUIRED]

**Return type** dict

#### Returns

### Response Syntax

```
{  
    'InstanceSet': [  
        {  
            'InstanceId': 'string',  
            'InstanceState': 'string',  
            'InstanceUniqueId': 'string'  
        },  
    ],  
    'RequestId': 'string',  
    'SnapshotName': 'string'  
}
```

#### Response Structure

- (*dict*) –
  - **InstanceSet** (*list*) –
    - \* (*dict*) –
      - **InstanceId** (*string*) –
      - **InstanceState** (*string*) –
      - **InstanceUniqueId** (*string*) –
  - **RequestId** (*string*) –
  - **SnapshotName** (*string*) –

*computing* / Client / nifty\_create\_nat\_rule

### nifty\_create\_nat\_rule

`computing.Client.nifty_create_nat_rule(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_create_nat_rule(  
    Description='string',  
    Destination={  
        'Port': 123  
    },  
    InboundInterface={  
        'NetworkId': 'string',  
        'NetworkName': 'string'  
    },  
)
```

(continues on next page)

(continued from previous page)

```

NatTableId='string',
NatType='snat'|'dnat',
OutboundInterface={
    'NetworkId': 'string',
    'NetworkName': 'string'
},
Protocol='ALL'|'TCP'|'UDP'|'TCP_UDP'|'ICMP',
RuleNumber='string',
Source={
    'Address': 'string',
    'Port': 123
},
Translation={
    'Address': 'string',
    'Port': 123
}
)

```

**Parameters**

- **Description** (*string*) –
- **Destination** (*dict*) –
  - **Port** (*integer*) –
- **InboundInterface** (*dict*) –
  - **NetworkId** (*string*) –
  - **NetworkName** (*string*) –
- **NatTableId** (*string*) – [REQUIRED]
- **NatType** (*string*) – [REQUIRED]
- **OutboundInterface** (*dict*) –
  - **NetworkId** (*string*) –
  - **NetworkName** (*string*) –
- **Protocol** (*string*) – [REQUIRED]
- **RuleNumber** (*string*) – [REQUIRED]
- **Source** (*dict*) –
  - **Address** (*string*) –
  - **Port** (*integer*) –
- **Translation** (*dict*) –
  - **Address** (*string*) –
  - **Port** (*integer*) –

**Return type** dict**Returns****Response Syntax**

```

{
  'NatRule': {
    'Description': 'string',
    'Destination': {
      'Port': 123
    },
    'InboundInterface': {
      'NetworkId': 'string',
      'NetworkName': 'string'
    },
    'NatType': 'string',
    'OutboundInterface': {

```

(continues on next page)

(continued from previous page)

```
        'NetworkId': 'string',
        'NetworkName': 'string'
    },
    'Protocol': 'string',
    'RuleNumber': 'string',
    'Source': {
        'Address': 'string',
        'Port': 123
    },
    'Translation': {
        'Address': 'string',
        'Port': 123
    }
},
'NatTableId': 'string',
'RequestId': 'string'
}
```

### Response Structure

- *(dict)* –
  - **NatRule** (*dict*) –
    - \* **Description** (*string*) –
    - \* **Destination** (*dict*) –
      - **Port** (*integer*) –
    - \* **InboundInterface** (*dict*) –
      - **NetworkId** (*string*) –
      - **NetworkName** (*string*) –
    - \* **NatType** (*string*) –
    - \* **OutboundInterface** (*dict*) –
      - **NetworkId** (*string*) –
      - **NetworkName** (*string*) –
    - \* **Protocol** (*string*) –
    - \* **RuleNumber** (*string*) –
    - \* **Source** (*dict*) –
      - **Address** (*string*) –
      - **Port** (*integer*) –
    - \* **Translation** (*dict*) –
      - **Address** (*string*) –
      - **Port** (*integer*) –
  - **NatTableId** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / nifty\_create\_nat\_table

### nifty\_create\_nat\_table

`computing.Client.nifty_create_nat_table()`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_create_nat_table()
```

**Return type** dict

## Returns

### Response Syntax

```
{
  'NatTable': {
    'NatTableId': 'string',
    'TagSet': [
      {
        'Key': 'string',
        'Value': 'string'
      },
    ]
  },
  'RequestId': 'string'
}
```

### Response Structure

- (dict) –
  - **NatTable** (dict) –
    - \* **NatTableId** (string) –
    - \* **TagSet** (list) –
      - (dict) –
      - **Key** (string) –
      - **Value** (string) –
  - **RequestId** (string) –

*computing* / Client / nifty\_create\_private\_lan

## nifty\_create\_private\_lan

`computing.Client.nifty_create_private_lan(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_create_private_lan(
    AccountingType='1'|'2',
    AvailabilityZone='string',
    CidrBlock='string',
    Description='string',
    PrivateLanName='string'
)
```

### Parameters

- **AccountingType** (string) –
- **AvailabilityZone** (string) –
- **CidrBlock** (string) – [REQUIRED]
- **Description** (string) –
- **PrivateLanName** (string) –

Return type dict

### Returns

#### Response Syntax

```
{
  'PrivateLan': {
```

(continues on next page)

(continued from previous page)

```

'AccountingType': 'string',
'AvailabilityZone': 'string',
'CidrBlock': 'string',
'CreateTime': datetime(2015, 1, 1),
'Description': 'string',
'ElasticLoadBalancingSet': [
    {
        'ElasticLoadBalancerName': 'string',
        'ElasticLoadBalancerPort': 123,
        'InstancePort': 123,
        'Protocol': 'string'
    },
],
'InstancesSet': [
    {
        'DeviceIndex': 'string',
        'InstanceId': 'string',
        'InstanceUniqueId': 'string',
        'IpAddress': 'string'
    },
],
'NetworkId': 'string',
'NetworkInterfaceSet': [
    {
        'IpAddress': 'string',
        'NetworkInterfaceId': 'string'
    },
],
'NextMonthAccountingType': 'string',
'PrivateLanName': 'string',
'RemoteAccessVpnGatewaySet': [
    {
        'DeviceIndex': 'string',
        'IpAddress': 'string',
        'RemoteAccessVpnGatewayId': 'string',
        'RemoteAccessVpnGatewayName': 'string'
    },
],
'RouterSet': [
    {
        'DeviceIndex': 'string',
        'IpAddress': 'string',
        'RouterId': 'string',
        'RouterName': 'string'
    },
],
'SharingStatus': 'string',
'State': 'string',
'TagSet': [
    {
        'Key': 'string',
        'Value': 'string'
    },
],
'VpnGatewaySet': [
    {
        'DeviceIndex': 'string',

```

(continues on next page)

(continued from previous page)

```

        'IpAddress': 'string',
        'NiftyVpnGatewayName': 'string',
        'VpnGatewayId': 'string'
    },
]
},
'RequestId': 'string'
}

```

## Response Structure

- *(dict)* –
  - **PrivateLan** (*dict*) –
    - \* **AccountingType** (*string*) –
    - \* **AvailabilityZone** (*string*) –
    - \* **CidrBlock** (*string*) –
    - \* **CreatedTime** (*datetime*) –
    - \* **Description** (*string*) –
    - \* **ElasticLoadBalancingSet** (*list*) –
      - (*dict*) –
      - **ElasticLoadBalancerName** (*string*) –
      - **ElasticLoadBalancerPort** (*integer*) –
      - **InstancePort** (*integer*) –
      - **Protocol** (*string*) –
    - \* **InstancesSet** (*list*) –
      - (*dict*) –
      - **DeviceIndex** (*string*) –
      - **InstanceId** (*string*) –
      - **InstanceUniqueId** (*string*) –
      - **IpAddress** (*string*) –
    - \* **NetworkId** (*string*) –
    - \* **NetworkInterfaceSet** (*list*) –
      - (*dict*) –
      - **IpAddress** (*string*) –
      - **NetworkInterfaceId** (*string*) –
    - \* **NextMonthAccountingType** (*string*) –
    - \* **PrivateLanName** (*string*) –
    - \* **RemoteAccessVpnGatewaySet** (*list*) –
      - (*dict*) –
      - **DeviceIndex** (*string*) –
      - **IpAddress** (*string*) –
      - **RemoteAccessVpnGatewayId** (*string*) –
      - **RemoteAccessVpnGatewayName** (*string*) –
    - \* **RouterSet** (*list*) –
      - (*dict*) –
      - **DeviceIndex** (*string*) –
      - **IpAddress** (*string*) –
      - **RouterId** (*string*) –
      - **RouterName** (*string*) –
    - \* **SharingStatus** (*string*) –
    - \* **State** (*string*) –
    - \* **TagSet** (*list*) –
      - (*dict*) –
      - **Key** (*string*) –

- **Value** (*string*) –
- \* **VpnGatewaySet** (*list*) –
  - (*dict*) –
  - **DeviceIndex** (*string*) –
  - **IpAddress** (*string*) –
  - **NiftyVpnGatewayName** (*string*) –
  - **VpnGatewayId** (*string*) –
- **RequestId** (*string*) –

*computing* / Client / nifty\_create\_router

## nifty\_create\_router

`computing.Client.nifty_create_router` (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_create_router(  
    AccountingType='1'|'2',  
    AvailabilityZone='string',  
    Description='string',  
    NetworkInterface=[  
        {  
            'DeviceIndex': 123,  
            'Dhcp': True|False,  
            'DhcpConfigId': 'string',  
            'DhcpOptionsId': 'string',  
            'IpAddress': 'string',  
            'ListOfRequestSecurityGroupId': [  
                'string',  
            ],  
            'NetworkId': 'string',  
            'NetworkName': 'string'  
        },  
    ],  
    RouterName='string',  
    SecurityGroup=[  
        'string',  
    ],  
    Type='small'|'medium'|'large'  
)
```

### Parameters

- **AccountingType** (*string*) –
- **AvailabilityZone** (*string*) –
- **Description** (*string*) –
- **NetworkInterface** (*list*) –
  - (*dict*) –
  - \* **DeviceIndex** (*integer*) –
  - \* **Dhcp** (*boolean*) –
  - \* **DhcpConfigId** (*string*) –
  - \* **DhcpOptionsId** (*string*) –
  - \* **IpAddress** (*string*) –
  - \* **ListOfRequestSecurityGroupId** (*list*) –
    - (*string*) –



- \* **NetworkId** (*string*) –
- \* **NetworkName** (*string*) –
- **RouterName** (*string*) –
- **SecurityGroup** (*list*) –
  - (*string*) –
- **Type** (*string*) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'RequestId': 'string',
  'Router': {
    'AccountingType': 'string',
    'AvailabilityZone': 'string',
    'BackupInformation': {
      'IsBackup': True|False
    },
    'Description': 'string',
    'GroupSet': [
      {
        'GroupId': 'string'
      },
    ],
    'NetworkInterfaceSet': [
      {
        'Dhcp': True|False,
        'DhcpConfigId': 'string',
        'DhcpOptionsId': 'string',
        'IpAddress': 'string',
        'NetworkId': 'string',
        'NetworkName': 'string'
      },
    ],
    'NextMonthAccountingType': 'string',
    'RouterId': 'string',
    'RouterName': 'string',
    'State': 'string',
    'Type': 'string',
    'VersionInformation': {
      'IsLatest': True|False,
      'Version': 'string'
    }
  },
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Router** (*dict*) –
    - \* **AccountingType** (*string*) –
    - \* **AvailabilityZone** (*string*) –
    - \* **BackupInformation** (*dict*) –
      - **IsBackup** (*boolean*) –
    - \* **Description** (*string*) –
    - \* **GroupSet** (*list*) –
      - (*dict*) –

- **GroupId** (*string*) –
- \* **NetworkInterfaceSet** (*list*) –
  - (*dict*) –
  - **Dhcp** (*boolean*) –
  - **DhcpConfigId** (*string*) –
  - **DhcpOptionsId** (*string*) –
  - **IpAddress** (*string*) –
  - **NetworkId** (*string*) –
  - **NetworkName** (*string*) –
- \* **NextMonthAccountingType** (*string*) –
- \* **RouterId** (*string*) –
- \* **RouterName** (*string*) –
- \* **State** (*string*) –
- \* **Type** (*string*) –
- \* **VersionInformation** (*dict*) –
  - **IsLatest** (*boolean*) –
  - **Version** (*string*) –

*computing* / Client / nifty\_create\_separate\_instance\_rule

## nifty\_create\_separate\_instance\_rule

`computing.Client.nifty_create_separate_instance_rule(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_create_separate_instance_rule(
    InstanceId=[
        'string',
    ],
    InstanceUniqueId=[
        'string',
    ],
    Placement={
        'AvailabilityZone': 'string'
    },
    SeparateInstanceRuleDescription='string',
    SeparateInstanceRuleName='string'
)
```

### Parameters

- **InstanceId** (*list*) –
  - (*string*) –
- **InstanceUniqueId** (*list*) –
  - (*string*) –
- **Placement** (*dict*) – [REQUIRED]
  - **AvailabilityZone** (*string*) – [REQUIRED]
- **SeparateInstanceRuleDescription** (*string*) –
- **SeparateInstanceRuleName** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
    'RequestId': 'string',
    'Return': 'string'
}
```

### Response Structure

- (dict) –
  - RequestId (string) –
  - Return (string) –

*computing* / Client / nifty\_create\_web\_proxy

## nifty\_create\_web\_proxy

`computing.Client.nifty_create_web_proxy(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_create_web_proxy(
    Agreement=True|False,
    BypassInterface={
        'NetworkId': 'string',
        'NetworkName': 'string'
    },
    Description='string',
    ListenInterface={
        'NetworkId': 'string',
        'NetworkName': 'string'
    },
    ListenPort='string',
    Option={
        'NameServer': 'string'
    },
    RouterId='string',
    RouterName='string'
)
```

### Parameters

- **Agreement** (*boolean*) –
- **BypassInterface** (*dict*) –
  - NetworkId (*string*) –
  - NetworkName (*string*) –
- **Description** (*string*) –
- **ListenInterface** (*dict*) –
  - NetworkId (*string*) –
  - NetworkName (*string*) –
- **ListenPort** (*string*) – [REQUIRED]
- **Option** (*dict*) –
  - NameServer (*string*) –
- **RouterId** (*string*) –
- **RouterName** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
  'RequestId': 'string',
  'WebProxy': {
    'BypassInterface': {
      'NetworkId': 'string',
      'NetworkName': 'string'
    },
    'Description': 'string',
    'ListenInterface': {
      'NetworkId': 'string',
      'NetworkName': 'string'
    },
    'ListenPort': 'string',
    'Option': {
      'NameServer': 'string'
    },
    'RouterId': 'string',
    'RouterName': 'string'
  }
}
```

#### Response Structure

- (dict) –
  - **RequestId** (string) –
  - **WebProxy** (dict) –
    - \* **BypassInterface** (dict) –
      - **NetworkId** (string) –
      - **NetworkName** (string) –
    - \* **Description** (string) –
    - \* **ListenInterface** (dict) –
      - **NetworkId** (string) –
      - **NetworkName** (string) –
    - \* **ListenPort** (string) –
    - \* **Option** (dict) –
      - **NameServer** (string) –
    - \* **RouterId** (string) –
    - \* **RouterName** (string) –

*computing* / Client / nifty\_delete\_alarm

### nifty\_delete\_alarm

`computing.Client.nifty_delete_alarm(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.nifty_delete_alarm(
    FunctionName='Server'|'LoadBalancer'|'DiskPartition'|'ElasticLoadBalancer',
    RuleName='string'
)
```

#### Parameters

- **FunctionName** (string) – [REQUIRED]
- **RuleName** (string) – [REQUIRED]

**Return type** dict

**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_delete\_auto\_scaling\_group

**nifty\_delete\_auto\_scaling\_group**

`computing.Client.nifty_delete_auto_scaling_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.nifty_delete_auto_scaling_group(
    AutoScalingGroupName='string'
)
```

**Parameters** `AutoScalingGroupName` (*string*) – [REQUIRED]

**Return type** dict

**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_delete\_dhcp\_config

**nifty\_delete\_dhcp\_config**

`computing.Client.nifty_delete_dhcp_config(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.nifty_delete_dhcp_config(
    DhcpConfigId='string'
)
```

**Parameters** `DhcpConfigId` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_delete\_dhcp\_ip\_address\_pool

### nifty\_delete\_dhcp\_ip\_address\_pool

`computing.Client.nifty_delete_dhcp_ip_address_pool (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.nifty_delete_dhcp_ip_address_pool(
    DhcpConfigId='string',
    StartIpAddress='string',
    StopIpAddress='string'
)
```

#### Parameters

- **DhcpConfigId** (*string*) – [REQUIRED]
- **StartIpAddress** (*string*) – [REQUIRED]
- **StopIpAddress** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_delete\_dhcp\_static\_mapping

### nifty\_delete\_dhcp\_static\_mapping

`computing.Client.nifty_delete_dhcp_static_mapping (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.nifty_delete_dhcp_static_mapping(
    DhcpConfigId='string',
    IpAddress='string',
    MacAddress='string'
)
```

**Parameters**

- **DhcpConfigId** (*string*) – [REQUIRED]
- **IpAddress** (*string*) – [REQUIRED]
- **MacAddress** (*string*) – [REQUIRED]

**Return type** dict**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_delete\_elastic\_load\_balancer**nifty\_delete\_elastic\_load\_balancer**`computing.Client.nifty_delete_elastic_load_balancer(**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.nifty_delete_elastic_load_balancer(
    ElasticLoadBalancerId='string',
    ElasticLoadBalancerName='string',
    ElasticLoadBalancerPort=123,
    InstancePort=123,
    Protocol='TCP'|'UDP'|'HTTP'|'HTTPS'
)
```

**Parameters**

- **ElasticLoadBalancerId** (*string*) –
- **ElasticLoadBalancerName** (*string*) –
- **ElasticLoadBalancerPort** (*integer*) – [REQUIRED]
- **InstancePort** (*integer*) – [REQUIRED]
- **Protocol** (*string*) – [REQUIRED]

**Return type** dict**Returns****Response Syntax**

```
{
    'NiftyDeleteElasticLoadBalancerResult': 'string',
    'ResponseMetadata': {
```

(continues on next page)

(continued from previous page)

```
    'RequestId': 'string'
  }
}
```

**Response Structure**

- (*dict*) –
  - **NiftyDeleteElasticLoadBalancerResult** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*computing* / Client / nifty\_delete\_instance\_snapshot

**nifty\_delete\_instance\_snapshot**

`computing.Client.nifty_delete_instance_snapshot (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.nifty_delete_instance_snapshot(
    InstanceSnapshotId='string',
    SnapshotName='string'
)
```

**Parameters**

- **InstanceSnapshotId** (*string*) –
- **SnapshotName** (*string*) –

**Return type** dict

**Returns****Response Syntax**

```
{
  'RequestId': 'string',
  'SnapshotInfoSet': [
    {
      'CreatedTime': 'string',
      'Difference': 'string',
      'ExpiredTime': 'string',
      'InstanceId': 'string',
      'InstanceSnapshotId': 'string',
      'Memo': 'string',
      'PowerStatus': 'string',
      'SnapshotName': 'string',
      'Status': 'string',
      'UpdateTime': 'string'
    },
  ],
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **SnapshotInfoSet** (*list*) –
    - \* (*dict*) –
      - **CreatedTime** (*string*) –



- **Difference** (*string*) –
- **ExpiredTime** (*string*) –
- **InstanceId** (*string*) –
- **InstanceSnapshotId** (*string*) –
- **Memo** (*string*) –
- **PowerStatus** (*string*) –
- **SnapshotName** (*string*) –
- **Status** (*string*) –
- **UpdateTime** (*string*) –

*computing* / Client / nifty\_delete\_nat\_rule

## nifty\_delete\_nat\_rule

`computing.Client.nifty_delete_nat_rule(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_delete_nat_rule(
    NatTableId='string',
    NatType='snat'|'dnat',
    RuleNumber='string'
)
```

### Parameters

- **NatTableId** (*string*) – [REQUIRED]
- **NatType** (*string*) – [REQUIRED]
- **RuleNumber** (*string*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_delete\_nat\_table

## nifty\_delete\_nat\_table

`computing.Client.nifty_delete_nat_table(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_delete_nat_table(
    NatTableId='string'
)
```

**Parameters** `NatTableId` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_delete\_private\_lan

### nifty\_delete\_private\_lan

`computing.Client.nifty_delete_private_lan` (*\*\*kwargs*)

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.nifty_delete_private_lan(
    NetworkId='string',
    PrivateLanName='string'
)
```

#### Parameters

- **NetworkId** (*string*) –
- **PrivateLanName** (*string*) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_delete\_router

### nifty\_delete\_router

`computing.Client.nifty_delete_router` (*\*\*kwargs*)

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.nifty_delete_router(
    RouterId='string',
    RouterName='string'
)
```

**Parameters**

- **RouterId** (*string*) –
- **RouterName** (*string*) –

**Return type** dict**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_delete\_separate\_instance\_rule**nifty\_delete\_separate\_instance\_rule**`computing.Client.nifty_delete_separate_instance_rule(**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.nifty_delete_separate_instance_rule(
    SeparateInstanceRuleName='string'
)
```

**Parameters** **SeparateInstanceRuleName** (*string*) – [REQUIRED]**Return type** dict**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': 'string'
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*string*) –

*computing* / Client / nifty\_delete\_web\_proxy

## nifty\_delete\_web\_proxy

computing.Client.nifty\_delete\_web\_proxy(\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_delete_web_proxy(  
    Agreement=True|False,  
    RouterId='string',  
    RouterName='string'  
)
```

### Parameters

- **Agreement** (*boolean*) –
- **RouterId** (*string*) –
- **RouterName** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{  
    'RequestId': 'string',  
    'Return': True|False  
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_deregister\_instances\_from\_elastic\_load\_balancer

## nifty\_deregister\_instances\_from\_elastic\_load\_balancer

computing.Client.nifty\_deregister\_instances\_from\_elastic\_load\_balancer(\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_deregister_instances_from_elastic_load_balancer(  
    ElasticLoadBalancerId='string',  
    ElasticLoadBalancerName='string',  
    ElasticLoadBalancerPort=123,  
    InstancePort=123,  
    Instances=[  
        {  
            'InstanceId': 'string',  
            'InstanceUniqueId': 'string'  
        },  
    ],  
    Protocol='TCP'|'UDP'|'HTTP'|'HTTPS'  
)
```

### Parameters

- **ElasticLoadBalancerId** (*string*) –
- **ElasticLoadBalancerName** (*string*) –

- **ElasticLoadBalancerPort** (*integer*) – [REQUIRED]
- **InstancePort** (*integer*) – [REQUIRED]
- **Instances** (*list*) –
  - (*dict*) –
    - \* **InstanceId** (*string*) –
    - \* **InstanceUniqueId** (*string*) –
- **Protocol** (*string*) – [REQUIRED]

Return type dict

Returns

#### Response Syntax

```
{
    'NiftyDeregisterInstancesFromElasticLoadBalancerResult': 'string'
    ↪ ',
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

#### Response Structure

- (*dict*) –
  - **NiftyDeregisterInstancesFromElasticLoadBalancerResult** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*computing* / Client / nifty\_deregister\_instances\_from\_separate\_instance\_rule

### nifty\_deregister\_instances\_from\_separate\_instance\_rule

`computing.Client.nifty_deregister_instances_from_separate_instance_rule(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.nifty_deregister_instances_from_separate_instance_rule(
    InstanceId=[
        'string',
    ],
    InstanceUniqueId=[
        'string',
    ],
    SeparateInstanceRuleName='string'
)
```

#### Parameters

- **InstanceId** (*list*) –
  - (*string*) –
- **InstanceUniqueId** (*list*) –
  - (*string*) –
- **SeparateInstanceRuleName** (*string*) – [REQUIRED]

Return type dict

Returns

#### Response Syntax

```
{
    'InstancesSet': [
        {
            'InstanceId': 'string',
            'InstanceUniqueId': 'string'
        },
    ],
    'RequestId': 'string'
}
```

#### Response Structure

- *(dict)* –
  - **InstancesSet** (*list*) –
    - \* *(dict)* –
      - **InstanceId** (*string*) –
      - **InstanceUniqueId** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / nifty\_deregister\_routers\_from\_security\_group

### nifty\_deregister\_routers\_from\_security\_group

`computing.Client.nifty_deregister_routers_from_security_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.nifty_deregister_routers_from_security_group(
    GroupName='string',
    RouterSet=[
        {
            'RouterId': 'string',
            'RouterName': 'string'
        },
    ]
)
```

#### Parameters

- **GroupName** (*string*) – [REQUIRED]
- **RouterSet** (*list*) –
  - *(dict)* –
    - \* **RouterId** (*string*) –
    - \* **RouterName** (*string*) –

**Return type** dict

#### Returns

#### Response Syntax

```
{
    'RequestId': 'string',
    'RouterSet': [
        {
            'RouterId': 'string',
            'RouterName': 'string'
        },
    ],
}
```

(continues on next page)

(continued from previous page)

```
    ]
}
```

**Response Structure**

- (dict) –
  - **RequestId** (string) –
  - **RouterSet** (list) –
    - \* (dict) –
      - **RouterId** (string) –
      - **RouterName** (string) –

*computing* / Client / `nifty_deregister_vpn_gateways_from_security_group`

**nifty\_deregister\_vpn\_gateways\_from\_security\_group**

`computing.Client.nifty_deregister_vpn_gateways_from_security_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.nifty_deregister_vpn_gateways_from_security_group(
    GroupName='string',
    VpnGatewaySet=[
        {
            'NiftyVpnGatewayName': 'string',
            'VpnGatewayId': 'string'
        },
    ]
)
```

**Parameters**

- **GroupName** (string) – [REQUIRED]
- **VpnGatewaySet** (list) –
  - (dict) –
    - \* **NiftyVpnGatewayName** (string) –
    - \* **VpnGatewayId** (string) –

**Return type** dict

**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'VpnGatewaySet': [
        {
            'NiftyVpnGatewayName': 'string',
            'VpnGatewayId': 'string'
        },
    ]
}
```

**Response Structure**

- (dict) –
  - **RequestId** (string) –
  - **VpnGatewaySet** (list) –
    - \* (dict) –

- **NiftyVpnGatewayName** (*string*) –
- **VpnGatewayId** (*string*) –

*computing* / Client / nifty\_describe\_alarm\_history

## nifty\_describe\_alarm\_history

`computing.Client.nifty_describe_alarm_history(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_describe_alarm_history(
    Rule=[
        {
            'FromDate': 'string',
            'FunctionName': 'Server'|'LoadBalancer'|'DiskPartition'|
↪ 'ElasticLoadBalancer',
            'RuleName': 'string',
            'ToDate': 'string'
        },
    ]
)
```

**Parameters** **Rule** (*list*) –

- (*dict*) –
  - **FromDate** (*string*) –
  - **FunctionName** (*string*) –
  - **RuleName** (*string*) –
  - **ToDate** (*string*) –

**Return type** dict

**Returns**

### Response Syntax

```
{
    'RequestId': 'string',
    'ReservationSet': [
        {
            'AlarmDateHistorySet': [
                {
                    'AlarmEventHistorySet': [
                        {
                            'AlarmEvent': 'string',
                            'AlarmEventDatetime': datetime(2015, 1, ↪
↪ 1)
                        },
                    ],
                    'Date': 'string'
                },
            ],
            'FunctionName': 'string',
            'RuleName': 'string'
        },
    ]
}
```

### Response Structure



- (*dict*) –
  - **RequestId** (*string*) –
  - **ReservationSet** (*list*) –
    - \* (*dict*) –
      - **AlarmDateHistorySet** (*list*) –
      - (*dict*) –
      - **AlarmEventHistorySet** (*list*) –
      - (*dict*) –
      - **AlarmEvent** (*string*) –
      - **AlarmEventDatetime** (*datetime*) –
      - **Date** (*string*) –
      - **FunctionName** (*string*) –
      - **RuleName** (*string*) –

*computing* / Client / nifty\_describe\_alarm\_rules\_activities

## nifty\_describe\_alarm\_rules\_activities

`computing.Client.nifty_describe_alarm_rules_activities(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_describe_alarm_rules_activities(
    Rule=[
        {
            'DataType': 'string',
            'FromDate': 'string',
            'FunctionName': 'Server'|'LoadBalancer'|'DiskPartition',
            'RuleName': 'string',
            'ToDate': 'string'
        },
    ]
)
```

**Parameters** **Rule** (*list*) –

- (*dict*) –
  - **DataType** (*string*) –
  - **FromDate** (*string*) –
  - **FunctionName** (*string*) –
  - **RuleName** (*string*) –
  - **ToDate** (*string*) –

**Return type** dict

**Returns**

### Response Syntax

```
{
    'RequestId': 'string',
    'ReservationSet': [
        {
            'AlarmRulesActivitiesSet': [
                {
                    'AlarmRulesActivitiesDateSet': [
                        {
                            'AlarmRulesActivitiesEventSet': [
```

(continues on next page)

(continued from previous page)

```

        {
            'AlarmRulesActivitiesEvent':
↪ 'string',
↪ 'AlarmRulesActivitiesEventDatetime': datetime(2015, 1, 1),
            'ResourceName': 'string',
            'Value': 'string'
        },
    ],
    'Date': 'string'
},
],
'DataType': 'string'
},
],
'FunctionName': 'string',
'RuleName': 'string'
},
]
}

```

**Response Structure**

- (dict) –
  - **RequestId** (string) –
  - **ReservationSet** (list) –
    - \* (dict) –
      - **AlarmRulesActivitiesSet** (list) –
      - (dict) –
      - **AlarmRulesActivitiesDataSet** (list) –
      - (dict) –
      - **AlarmRulesActivitiesEventSet** (list) –
      - (dict) –
      - **AlarmRulesActivitiesEvent** (string) –
      - **AlarmRulesActivitiesEventDatetime** (datetime) –
      - **ResourceName** (string) –
      - **Value** (string) –
      - **Date** (string) –
      - **DataType** (string) –
      - **FunctionName** (string) –
      - **RuleName** (string) –

*computing* / Client / nifty\_describe\_alarms**nifty\_describe\_alarms**`computing.Client.nifty_describe_alarms (**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```

response = client.nifty_describe_alarms(
    Rule=[
        {
            'FunctionName': 'Server'|'LoadBalancer'|'DiskPartition'|
↪ 'ElasticLoadBalancer',

```

(continues on next page)

(continued from previous page)

```

        'RuleName': 'string'
    },
]
)

```

**Parameters** **Rule** (*list*) –

- (*dict*) –
  - **FunctionName** (*string*) –
  - **RuleName** (*string*) –

**Return type** dict

**Returns**

### Response Syntax

```

{
    'RequestId': 'string',
    'ReservationSet': [
        {
            'AlarmCondition': 'string',
            'AlarmState': 'string',
            'AlarmTargetsSet': [
                {
                    'ResourceName': 'string'
                },
            ],
            'CreatedTime': datetime(2015, 1, 1),
            'Description': 'string',
            'EmailAddressSet': [
                {
                    'EmailAddress': 'string'
                },
            ],
            'FunctionName': 'string',
            'RuleName': 'string',
            'RuleSet': [
                {
                    'AddDatetime': datetime(2015, 1, 1),
                    'BreachDuration': 123,
                    'DataType': 'string',
                    'Threshold': 123.0,
                    'UpperLowerCondition': 'string'
                },
            ],
            'Zone': 'string'
        },
    ],
}

```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **ReservationSet** (*list*) –
    - \* (*dict*) –
      - **AlarmCondition** (*string*) –
      - **AlarmState** (*string*) –
      - **AlarmTargetsSet** (*list*) –

- *(dict)* –
- **ResourceName** (*string*) –
- **CreatedTime** (*datetime*) –
- **Description** (*string*) –
- **EmailAddressSet** (*list*) –
- *(dict)* –
- **EmailAddress** (*string*) –
- **FunctionName** (*string*) –
- **RuleName** (*string*) –
- **RuleSet** (*list*) –
- *(dict)* –
- **AddDatetime** (*datetime*) –
- **BreachDuration** (*integer*) –
- **DataType** (*string*) –
- **Threshold** (*float*) –
- **UpperLowerCondition** (*string*) –
- **Zone** (*string*) –

*computing* / Client / nifty\_describe\_alarms\_partitions

## nifty\_describe\_alarms\_partitions

`computing.Client.nifty_describe_alarms_partitions (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_describe_alarms_partitions(
    InstanceId=[
        'string',
    ]
)
```

**Parameters** **InstanceId** (*list*) –

- (*string*) –

**Return type** dict

**Returns**

### Response Syntax

```
{
    'AlarmTargetSet': [
        {
            'InstanceId': 'string',
            'PartitionsSet': [
                {
                    'Partition': 'string'
                },
            ],
        },
    ],
    'RequestId': 'string'
}
```

**Response Structure**

- (*dict*) –
  - **AlarmTargetSet** (*list*) –

- \* (*dict*) –
  - **InstanceId** (*string*) –
  - **PartitionsSet** (*list*) –
  - (*dict*) –
  - **Partition** (*string*) –
- **RequestId** (*string*) –

*computing* / Client / nifty\_describe\_auto\_scaling\_groups

## nifty\_describe\_auto\_scaling\_groups

`computing.Client.nifty_describe_auto_scaling_groups(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_describe_auto_scaling_groups(
    AutoScalingGroupName=[
        'string',
    ]
)
```

**Parameters** `AutoScalingGroupName` (*list*) –

- (*string*) –

**Return type** dict

**Returns**

### Response Syntax

```
{
    'AutoScalingReservationSet': [
        {
            'Alarm': 'string',
            'AutoScalingGroupName': 'string',
            'ChangeInCapacity': 123,
            'CreatedTime': datetime(2015, 1, 1),
            'DefaultCooldown': 123,
            'Description': 'string',
            'GroupSet': [
                {
                    'GroupId': 'string'
                },
            ],
            'ImageId': 'string',
            'InstanceLifecycleLimit': 123,
            'InstanceType': 'string',
            'InstancesSet': [
                {
                    'DnsName': 'string',
                    'ExpireTime': datetime(2015, 1, 1),
                    'InstanceId': 'string',
                    'InstanceState': {
                        'Code': 123,
                        'Name': 'string'
                    },
                    'InstanceType': 'string',
                    'InstanceUniqueId': 'string'
                },
            ],
        },
    ],
}
```

(continues on next page)

(continued from previous page)

```

        },
    ],
    'LoadBalancing': [
        {
            'InstancePort': 123,
            'LoadBalancerName': 'string',
            'LoadBalancerPort': 123
        },
    ],
    'MaxSize': 123,
    'MinSize': 123,
    'Placement': {
        'AvailabilityZone': 'string'
    },
    'Scaleout': 123,
    'ScaleoutCondition': 'string',
    'ScheduleSet': [
        {
            'DDay': {
                'EndingDDay': 'string',
                'StartingDDay': 'string'
            },
            'Day': {
                'SetFriday': 'string',
                'SetMonday': 'string',
                'SetSaturday': 'string',
                'SetSunday': 'string',
                'SetThursday': 'string',
                'SetTuesday': 'string',
                'SetWednesday': 'string'
            },
            'Month': {
                'EndingMonth': 'string',
                'StartingMonth': 'string'
            },
            'TimeZone': {
                'EndingTimeZone': 'string',
                'StartingTimeZone': 'string'
            }
        },
    ],
    'TriggerSet': [
        {
            'BreachDuration': 123,
            'Resource': 'string',
            'UpperThreshold': 123.0
        },
    ],
    ],
    'RequestId': 'string'
}

```

**Response Structure**

- (dict) –
  - **AutoScalingReservationSet** (list) –
    - \* (dict) –

- **Alarm** (*string*) –
- **AutoScalingGroupName** (*string*) –
- **ChangeInCapacity** (*integer*) –
- **CreatedTime** (*datetime*) –
- **DefaultCooldown** (*integer*) –
- **Description** (*string*) –
- **GroupSet** (*list*) –
- (*dict*) –
- **GroupId** (*string*) –
- **ImageId** (*string*) –
- **InstanceLifecycleLimit** (*integer*) –
- **InstanceType** (*string*) –
- **InstancesSet** (*list*) –
- (*dict*) –
- **DnsName** (*string*) –
- **ExpireTime** (*datetime*) –
- **InstanceId** (*string*) –
- **InstanceState** (*dict*) –
- **Code** (*integer*) –
- **Name** (*string*) –
- **InstanceType** (*string*) –
- **InstanceUniqueId** (*string*) –
- **LoadBalancing** (*list*) –
- (*dict*) –
- **InstancePort** (*integer*) –
- **LoadBalancerName** (*string*) –
- **LoadBalancerPort** (*integer*) –
- **MaxSize** (*integer*) –
- **MinSize** (*integer*) –
- **Placement** (*dict*) –
- **AvailabilityZone** (*string*) –
- **Scaleout** (*integer*) –
- **ScaleoutCondition** (*string*) –
- **ScheduleSet** (*list*) –
- (*dict*) –
- **DDay** (*dict*) –
- **EndingDDay** (*string*) –
- **StartingDDay** (*string*) –
- **Day** (*dict*) –
- **SetFriday** (*string*) –
- **SetMonday** (*string*) –
- **SetSaturday** (*string*) –
- **SetSunday** (*string*) –
- **SetThursday** (*string*) –
- **SetTuesday** (*string*) –
- **SetWednesday** (*string*) –
- **Month** (*dict*) –
- **EndingMonth** (*string*) –
- **StartingMonth** (*string*) –
- **TimeZone** (*dict*) –
- **EndingTimeZone** (*string*) –
- **StartingTimeZone** (*string*) –
- **TriggerSet** (*list*) –
- (*dict*) –

- **BreachDuration** (*integer*) –
- **Resource** (*string*) –
- **UpperThreshold** (*float*) –
- **RequestId** (*string*) –

*computing* / Client / `nifty_describe_corporate_info_for_certificate`

## **nifty\_describe\_corporate\_info\_for\_certificate**

`computing.Client.nifty_describe_corporate_info_for_certificate()`

See also: [NIFCLOUD API Documentation](#)

### **Request Syntax**

```
response = client.nifty_describe_corporate_info_for_certificate()
```

**Return type** dict

**Returns**

### **Response Syntax**

```
{
    'AlphabetName1': 'string',
    'AlphabetName2': 'string',
    'City': 'string',
    'CorpGrade': 'string',
    'CorpName': 'string',
    'DivisionName': 'string',
    'EmailAddress': 'string',
    'KanaName1': 'string',
    'KanaName2': 'string',
    'Name1': 'string',
    'Name2': 'string',
    'PhoneNumber': 'string',
    'PostName': 'string',
    'Pref': 'string',
    'PresidentName1': 'string',
    'PresidentName2': 'string',
    'RequestId': 'string',
    'TdbCode': 'string',
    'Zip1': 'string',
    'Zip2': 'string'
}
```

### **Response Structure**

- (*dict*) –
  - **AlphabetName1** (*string*) –
  - **AlphabetName2** (*string*) –
  - **City** (*string*) –
  - **CorpGrade** (*string*) –
  - **CorpName** (*string*) –
  - **DivisionName** (*string*) –
  - **EmailAddress** (*string*) –
  - **KanaName1** (*string*) –
  - **KanaName2** (*string*) –
  - **Name1** (*string*) –
  - **Name2** (*string*) –



- **PhoneNumber** (*string*) –
- **PostName** (*string*) –
- **Pref** (*string*) –
- **PresidentName1** (*string*) –
- **PresidentName2** (*string*) –
- **RequestId** (*string*) –
- **TdbCode** (*string*) –
- **Zip1** (*string*) –
- **Zip2** (*string*) –

*computing* / Client / nifty\_describe\_dhcp\_configs

## nifty\_describe\_dhcp\_configs

`computing.Client.nifty_describe_dhcp_configs (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_describe_dhcp_configs(
    DhcpConfigId=[
        'string',
    ],
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'dhcp-config-id'|'ipaddress-pool-start'|'ipaddress-pool-stop'|
↪ 'ipaddress-pool-description'|'static-mapping-ipaddress'|'static-mapping-
↪ macaddress'|'static-mapping-description'
        },
    ]
)
```

### Parameters

- **DhcpConfigId** (*list*) –
  - (*string*) –
- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
    'DhcpConfigsSet': [
        {
            'DhcpConfigId': 'string',
            'IpAddressPoolsSet': [
                {
                    'Description': 'string',
                    'StartIpAddress': 'string',
```

(continues on next page)

(continued from previous page)

```

        'StopIpAddress': 'string'
    },
],
'StaticMappingsSet': [
    {
        'Description': 'string',
        'IpAddress': 'string',
        'MacAddress': 'string'
    },
]
},
'RequestId': 'string'
}

```

**Response Structure**

- (dict) –
  - **DhcpConfigsSet** (list) –
    - \* (dict) –
      - **DhcpConfigId** (string) –
      - **IpAddressPoolsSet** (list) –
      - (dict) –
      - **Description** (string) –
      - **StartIpAddress** (string) –
      - **StopIpAddress** (string) –
      - **StaticMappingsSet** (list) –
      - (dict) –
      - **Description** (string) –
      - **IpAddress** (string) –
      - **MacAddress** (string) –
  - **RequestId** (string) –

*computing* / Client / nifty\_describe\_dhcp\_status

**nifty\_describe\_dhcp\_status**

`computing.Client.nifty_describe_dhcp_status (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```

response = client.nifty_describe_dhcp_status(
    RouterId='string',
    RouterName='string'
)

```

**Parameters**

- **RouterId** (string) –
- **RouterName** (string) –

**Return type** dict

**Returns****Response Syntax**

```
{
  'DhcpStatusInformationSet': [
    {
      'DhcpIpAddressInformation': {
        'DhcpIpAddressSet': [
          {
            'ClientName': 'string',
            'Description': 'string',
            'IpAddress': 'string',
            'LeaseExpiration': datetime(2015, 1, 1),
            'LeaseType': 'string',
            'MacAddress': 'string'
          },
        ],
        'IpAddressPoolSet': [
          {
            'Description': 'string',
            'StartIpAddress': 'string',
            'StopIpAddress': 'string'
          },
        ],
      },
      'NetworkId': 'string',
      'PrivateLanName': 'string'
    },
  ],
  'RequestId': 'string',
  'RouterId': 'string',
  'RouterName': 'string'
}
```

### Response Structure

- (dict) –
  - **DhcpStatusInformationSet** (list) –
    - \* (dict) –
      - **DhcpIpAddressInformation** (dict) –
      - **DhcpIpAddressSet** (list) –
      - (dict) –
      - **ClientName** (string) –
      - **Description** (string) –
      - **IpAddress** (string) –
      - **LeaseExpiration** (datetime) –
      - **LeaseType** (string) –
      - **MacAddress** (string) –
      - **IpAddressPoolSet** (list) –
      - (dict) –
      - **Description** (string) –
      - **StartIpAddress** (string) –
      - **StopIpAddress** (string) –
      - **NetworkId** (string) –
      - **PrivateLanName** (string) –
    - **RequestId** (string) –
    - **RouterId** (string) –
    - **RouterName** (string) –

*computing* / Client / nifty\_describe\_elastic\_load\_balancers

## nifty\_describe\_elastic\_load\_balancers

computing.Client.**nifty\_describe\_elastic\_load\_balancers** (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_describe_elastic_load_balancers(
    ElasticLoadBalancers={
        'ListOfRequestElasticLoadBalancerId': [
            'string',
        ],
        'ListOfRequestElasticLoadBalancerName': [
            'string',
        ],
        'ListOfRequestElasticLoadBalancerPort': [
            123,
        ],
        'ListOfRequestInstancePort': [
            123,
        ],
        'ListOfRequestProtocol': [
            'string',
        ]
    },
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'availability-zone'|'state'|'elastic-loadbalancer-id'|
↪ 'elastic-loadbalancer-name'|'description'|'accounting-type'|'ip-address'|
↪ 'version'
        },
    ]
)
```

### Parameters

- **ElasticLoadBalancers** (*dict*) –
  - **ListOfRequestElasticLoadBalancerId** (*list*) –
    - \* (*string*) –
  - **ListOfRequestElasticLoadBalancerName** (*list*) –
    - \* (*string*) –
  - **ListOfRequestElasticLoadBalancerPort** (*list*) –
    - \* (*integer*) –
  - **ListOfRequestInstancePort** (*list*) –
    - \* (*integer*) –
  - **ListOfRequestProtocol** (*list*) –
    - \* (*string*) –
- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –

**Return type** dict

**Returns**

### Response Syntax

```

{
  'NiftyDescribeElasticLoadBalancersResult': {
    'ElasticLoadBalancerDescriptions': [
      {
        'AccountingType': 'string',
        'AvailabilityZones': [
          'string',
        ],
        'CreatedTime': datetime(2015, 1, 1),
        'DNSName': 'string',
        'ElasticLoadBalancerId': 'string',
        'ElasticLoadBalancerListenerDescriptions': [
          {
            'Listener': {
              'BalancingType': 123,
              'Description': 'string',
              'ElasticLoadBalancerPort': 123,
              'HealthCheck': {
                'Expectation': [
                  {
                    'HttpCode': 'string'
                  },
                ],
              },
              'InstanceStates': [
                {
                  'Description': 'string',
                  'InstanceId': 'string',
                  'InstanceUniqueId': 'string',
                  'ReasonCode': 'string',
                  'State': 'string'
                },
              ],
              'Interval': 123,
              'Path': 'string',
              'Target': 'string',
              'UnhealthyThreshold': 123
            },
            'InstancePort': 123,
            'Instances': [
              {
                'InstanceId': 'string',
                'InstanceUniqueId': 'string'
              },
            ],
            'Protocol': 'string',
            'SSLCertificateId': 'string',
            'SessionStickinessPolicy': {
              'Enabled': True|False,
              'ExpirationPeriod': 123,
              'Method': 123
            },
            'SorryPage': {
              'Enabled': True|False,
              'RedirectUrl': 'string'
            }
          },
        ],
      },
    ],
  },
},

```

(continues on next page)

(continued from previous page)

```

        },
    ],
    'ElasticLoadBalancerName': 'string',
    'NetworkInterfaces': [
        {
            'Description': 'string',
            'DeviceIndex': 'string',
            'IpAddress': 'string',
            'IsVipNetwork': True|False,
            'NetworkId': 'string',
            'NetworkName': 'string',
            'SystemIpAddresses': [
                {
                    'SystemIpAddress': 'string'
                },
            ],
        },
    ],
    'NetworkVolume': 'string',
    'NextMonthAccountingType': 'string',
    'RouteTableAssociationId': 'string',
    'RouteTableId': 'string',
    'State': 'string',
    'VersionInformation': {
        'IsLatest': True|False,
        'Version': 'string'
    },
},
]
},
'ResponseMetadata': {
    'RequestId': 'string'
}
}

```

**Response Structure**

- (dict) –
  - **NiftyDescribeElasticLoadBalancersResult** (dict) –
    - \* **ElasticLoadBalancerDescriptions** (list) –
      - (dict) –
      - **AccountingType** (string) –
      - **AvailabilityZones** (list) –
      - (string) –
      - **CreatedTime** (datetime) –
      - **DNSName** (string) –
      - **ElasticLoadBalancerId** (string) –
      - **ElasticLoadBalancerListenerDescriptions** (list) –
      - (dict) –
      - **Listener** (dict) –
      - **BalancingType** (integer) –
      - **Description** (string) –
      - **ElasticLoadBalancerPort** (integer) –
      - **HealthCheck** (dict) –
      - **Expectation** (list) –
      - (dict) –

- **HttpCode** (*string*) –
- **InstanceStates** (*list*) –
- (*dict*) –
- **Description** (*string*) –
- **InstanceId** (*string*) –
- **InstanceUniqueId** (*string*) –
- **ReasonCode** (*string*) –
- **State** (*string*) –
- **Interval** (*integer*) –
- **Path** (*string*) –
- **Target** (*string*) –
- **UnhealthyThreshold** (*integer*) –
- **InstancePort** (*integer*) –
- **Instances** (*list*) –
- (*dict*) –
- **InstanceId** (*string*) –
- **InstanceUniqueId** (*string*) –
- **Protocol** (*string*) –
- **SSLCertificateId** (*string*) –
- **SessionStickinessPolicy** (*dict*) –
- **Enabled** (*boolean*) –
- **ExpirationPeriod** (*integer*) –
- **Method** (*integer*) –
- **SorryPage** (*dict*) –
- **Enabled** (*boolean*) –
- **RedirectUrl** (*string*) –
- **ElasticLoadBalancerName** (*string*) –
- **NetworkInterfaces** (*list*) –
- (*dict*) –
- **Description** (*string*) –
- **DeviceIndex** (*string*) –
- **IpAddress** (*string*) –
- **IsVipNetwork** (*boolean*) –
- **NetworkId** (*string*) –
- **NetworkName** (*string*) –
- **SystemIpAddresses** (*list*) –
- (*dict*) –
- **SystemIpAddress** (*string*) –
- **NetworkVolume** (*string*) –
- **NextMonthAccountingType** (*string*) –
- **RouteTableAssociationId** (*string*) –
- **RouteTableId** (*string*) –
- **State** (*string*) –
- **VersionInformation** (*dict*) –
- **IsLatest** (*boolean*) –
- **Version** (*string*) –
- **ResponseMetadata** (*dict*) –
- \* **RequestId** (*string*) –

*computing* / Client / nifty\_describe\_instance\_elastic\_load\_balancer\_health

## nifty\_describe\_instance\_elastic\_load\_balancer\_health

`computing.Client.nifty_describe_instance_elastic_load_balancer_health(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_describe_instance_elastic_load_balancer_health(
    ElasticLoadBalancerId='string',
    ElasticLoadBalancerName='string',
    ElasticLoadBalancerPort=123,
    InstancePort=123,
    Instances=[
        {
            'InstanceId': 'string',
            'InstanceUniqueId': 'string'
        },
    ],
    Protocol='TCP'|'UDP'|'HTTP'|'HTTPS'
)
```

### Parameters

- **ElasticLoadBalancerId** (*string*) –
- **ElasticLoadBalancerName** (*string*) –
- **ElasticLoadBalancerPort** (*integer*) – [REQUIRED]
- **InstancePort** (*integer*) – [REQUIRED]
- **Instances** (*list*) –
  - (*dict*) –
    - \* **InstanceId** (*string*) –
    - \* **InstanceUniqueId** (*string*) –
- **Protocol** (*string*) – [REQUIRED]

**Return type** dict

### Returns

#### Response Syntax

```
{
    'NiftyDescribeInstanceElasticLoadBalancerHealthResult': {
        'InstanceStates': [
            {
                'Description': 'string',
                'InstanceId': 'string',
                'InstanceUniqueId': 'string',
                'ReasonCode': 'string',
                'State': 'string'
            },
        ],
    },
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

### Response Structure

- (*dict*) –
  - **NiftyDescribeInstanceElasticLoadBalancerHealthResult** (*dict*) –
    - \* **InstanceStates** (*list*) –
      - (*dict*) –



- **Description** (*string*) –
- **InstanceId** (*string*) –
- **InstanceUniqueId** (*string*) –
- **ReasonCode** (*string*) –
- **State** (*string*) –
- **ResponseMetadata** (*dict*) –
- \* **RequestId** (*string*) –

*computing* / Client / nifty\_describe\_instance\_snapshots

## nifty\_describe\_instance\_snapshots

`computing.Client.nifty_describe_instance_snapshots (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_describe_instance_snapshots(
    InstanceSnapshotId=[
        'string',
    ],
    SnapshotName=[
        'string',
    ]
)
```

### Parameters

- **InstanceSnapshotId** (*list*) –
- (*string*) –
- **SnapshotName** (*list*) –
- (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
    'RequestId': 'string',
    'SnapshotInfoSet': [
        {
            'CreatedTime': 'string',
            'Difference': 'string',
            'ExpiredTime': 'string',
            'InstanceId': 'string',
            'InstanceSnapshotId': 'string',
            'Memo': 'string',
            'PowerStatus': 'string',
            'SnapshotName': 'string',
            'Status': 'string',
            'UpdateTime': 'string'
        },
    ]
}
```

### Response Structure

- (*dict*) –
- **RequestId** (*string*) –

- **SnapshotInfoSet** (*list*) –
  - \* (*dict*) –
    - **CreatedTime** (*string*) –
    - **Difference** (*string*) –
    - **ExpiredTime** (*string*) –
    - **InstanceId** (*string*) –
    - **InstanceSnapshotId** (*string*) –
    - **Memo** (*string*) –
    - **PowerStatus** (*string*) –
    - **SnapshotName** (*string*) –
    - **Status** (*string*) –
    - **UpdatedTime** (*string*) –

*computing* / Client / nifty\_describe\_load\_balancer\_ssl\_policies

## nifty\_describe\_load\_balancer\_ssl\_policies

`computing.Client.nifty_describe_load_balancer_ssl_policies(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_describe_load_balancer_ssl_policies(
    LoadBalancerName='string'
)
```

**Parameters** `LoadBalancerName` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
  'NiftyDescribeLoadBalancerSSLPoliciesResult': {
    'LoadBalancerName': 'string',
    'SSLPoliciesDescriptions': [
      {
        'SSLPolicyId': 123,
        'SSLPolicyName': 'string',
        'SSLPolicySet': [
          {
            'Cipher': 'string'
          },
        ],
      },
    ],
  },
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

### Response Structure

- (*dict*) –
  - **NiftyDescribeLoadBalancerSSLPoliciesResult** (*dict*) –
    - \* **LoadBalancerName** (*string*) –
    - \* **SSLPoliciesDescriptions** (*list*) –

- (dict) –
- **SSLPolicyId** (integer) –
- **SSLPolicyName** (string) –
- **SSLPolicySet** (list) –
- (dict) –
- **Cipher** (string) –
- **ResponseMetadata** (dict) –
  - \* **RequestId** (string) –

*computing* / Client / nifty\_describe\_nat\_tables

## nifty\_describe\_nat\_tables

`computing.Client.nifty_describe_nat_tables(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_describe_nat_tables(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'association.association-id'|'association.nat-table-id'|
↪ 'association.router-id'|'nat-table-id'|'nat-rule.nat-type'|'nat-rule.rule-number
↪ '| 'nat-rule.description'|'nat-rule.protocol'|'nat-rule.outbound-interface.
↪ network-id'|'nat-rule.outbound-interface.network-name'|'nat-rule.inbound-
↪ interface.network-id'|'nat-rule.inbound-interface.network-name'|'nat-rule.
↪ destination.address'|'nat-rule.destination.port'|'nat-rule.source.address'|'nat-
↪ rule.source.port'|'nat-rule.translation.address'|'nat-rule.translation.port'
        },
    ],
    NatTableId=[
        'string',
    ]
)
```

### Parameters

- **Filter** (list) –
  - (dict) –
    - \* **ListOfRequestValue** (list) –
      - (string) –
    - \* **Name** (string) –
- **NatTableId** (list) –
  - (string) –

**Return type** dict

**Returns**

### Response Syntax

```
{
    'NatTableSet': [
        {
            'AssociationSet': [
                {
```

(continues on next page)

(continued from previous page)

```

        'AssociationId': 'string',
        'NatTableId': 'string',
        'RouterId': 'string',
        'RouterName': 'string'
    },
],
'NatRuleSet': [
    {
        'Description': 'string',
        'Destination': {
            'Port': 123
        },
        'InboundInterface': {
            'NetworkId': 'string',
            'NetworkName': 'string'
        },
        'NatType': 'string',
        'OutboundInterface': {
            'NetworkId': 'string',
            'NetworkName': 'string'
        },
        'Protocol': 'string',
        'RuleNumber': 'string',
        'Source': {
            'Address': 'string',
            'Port': 123
        },
        'Translation': {
            'Address': 'string',
            'Port': 123
        }
    },
],
'NatTableId': 'string',
'TagSet': [
    {
        'Key': 'string',
        'Value': 'string'
    },
],
],
'RequestId': 'string'
}

```

**Response Structure**

- (dict) –
  - **NatTableSet** (list) –
    - \* (dict) –
      - **AssociationSet** (list) –
      - (dict) –
      - **AssociationId** (string) –
      - **NatTableId** (string) –
      - **RouterId** (string) –
      - **RouterName** (string) –
      - **NatRuleSet** (list) –

- (dict) –
- **Description** (string) –
- **Destination** (dict) –
- **Port** (integer) –
- **InboundInterface** (dict) –
- **NetworkId** (string) –
- **NetworkName** (string) –
- **NatType** (string) –
- **OutboundInterface** (dict) –
- **NetworkId** (string) –
- **NetworkName** (string) –
- **Protocol** (string) –
- **RuleNumber** (string) –
- **Source** (dict) –
- **Address** (string) –
- **Port** (integer) –
- **Translation** (dict) –
- **Address** (string) –
- **Port** (integer) –
- **NatTableId** (string) –
- **TagSet** (list) –
- (dict) –
- **Key** (string) –
- **Value** (string) –
- **RequestId** (string) –

*computing* / Client / nifty\_describe\_performance\_chart

## nifty\_describe\_performance\_chart

`computing.Client.nifty_describe_performance_chart (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_describe_performance_chart(
    DataType=[
        'string',
    ],
    FromDate='string',
    FunctionName='Server'|'LoadBalancer'|'LB'|'DiskPartition'|'DP'|
    ↪ 'ElasticLoadBalancer'|'ELB',
    ResourceName=[
        'string',
    ],
    ToDate='string',
    ValueType='1'|'2'
)
```

### Parameters

- **DataType** (list) –
  - (string) –
- **FromDate** (string) –
- **FunctionName** (string) – [REQUIRED]
- **ResourceName** (list) – [REQUIRED]
  - (string) –

- **ToDate**(*string*)-
- **ValueType**(*string*)-

**Return type** dict

## Returns

## Response Syntax

```
{
  'FunctionName': 'string',
  'PerformanceChartSet': [
    {
      'DataSet': [
        {
          'DateTime': 'string',
          'Value': 'string'
        },
        {
          'DateTime': 'string',
          'Value': 'string'
        }
      ],
      'DataSetType': 'string',
      'ResourceName': 'string'
    },
    {
      'DataSet': [
        {
          'DateTime': 'string',
          'Value': 'string'
        },
        {
          'DateTime': 'string',
          'Value': 'string'
        }
      ],
      'DataSetType': 'string',
      'ResourceName': 'string'
    }
  ],
  'RequestId': 'string',
  'ValueType': 'string'
}
```

## Response Structure

- *(dict)* –
  - **FunctionName** (*string*) –
  - **PerformanceChartSet** (*list*) –
    - \* (*dict*) –
      - **DataSet** (*list*) –
      - (*dict*) –
      - **DateTime** (*string*) –
      - **Value** (*string*) –
      - **DataType** (*string*) –
      - **ResourceName** (*string*) –
  - **RequestId** (*string*) –
  - **ValueType** (*string*) –

*computing* / Client / nifty\_describe\_private\_lans

## nifty\_describe\_private\_lans

```
computing.Client.nifty_describe_private_lans (**kwargs)
```

See also: [NIFCLOUD API Documentation](#)

## Request Syntax

```
response = client.nifty_describe_private_lans(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'availabilityZone, availability-zone'|'cidrBlock, cidr, cidr-
↪block'|'state'|'network-id'|'private-lan-name'|'accountingType'|'description'
        },
    ],
)
```

(continues on next page)

(continued from previous page)

```

],
NetworkId=[
    'string',
],
PrivateLanName=[
    'string',
]
)

```

**Parameters**

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **NetworkId** (*list*) –
  - (*string*) –
- **PrivateLanName** (*list*) –
  - (*string*) –

**Return type** dict**Returns****Response Syntax**

```

{
  'PrivateLanSet': [
    {
      'AccountingType': 'string',
      'AvailabilityZone': 'string',
      'CidrBlock': 'string',
      'CreatedTime': datetime(2015, 1, 1),
      'Description': 'string',
      'ElasticLoadBalancingSet': [
        {
          'ElasticLoadBalancerName': 'string',
          'ElasticLoadBalancerPort': 123,
          'InstancePort': 123,
          'Protocol': 'string'
        },
      ],
      'InstancesSet': [
        {
          'DeviceIndex': 'string',
          'InstanceId': 'string',
          'InstanceUniqueId': 'string',
          'IpAddress': 'string'
        },
      ],
      'NetworkId': 'string',
      'NetworkInterfaceSet': [
        {
          'IpAddress': 'string',
          'NetworkInterfaceId': 'string'
        },
      ],
      'NextMonthAccountingType': 'string',
    },
  ],
}

```

(continues on next page)

(continued from previous page)

```

    'PrivateLanName': 'string',
    'RemoteAccessVpnGatewaySet': [
        {
            'DeviceIndex': 'string',
            'IpAddress': 'string',
            'RemoteAccessVpnGatewayId': 'string',
            'RemoteAccessVpnGatewayName': 'string'
        },
    ],
    'RouterSet': [
        {
            'DeviceIndex': 'string',
            'IpAddress': 'string',
            'RouterId': 'string',
            'RouterName': 'string'
        },
    ],
    'SharingStatus': 'string',
    'State': 'string',
    'TagSet': [
        {
            'Key': 'string',
            'Value': 'string'
        },
    ],
    'VpnGatewaySet': [
        {
            'DeviceIndex': 'string',
            'IpAddress': 'string',
            'NiftyVpnGatewayName': 'string',
            'VpnGatewayId': 'string'
        },
    ],
    ],
    'RequestId': 'string'
}

```

**Response Structure**

- (dict) –
  - PrivateLanSet (list) –
    - \* (dict) –
      - AccountingType (string) –
      - AvailabilityZone (string) –
      - CidrBlock (string) –
      - CreatedTime (datetime) –
      - Description (string) –
      - ElasticLoadBalancingSet (list) –
      - (dict) –
      - ElasticLoadBalancerName (string) –
      - ElasticLoadBalancerPort (integer) –
      - InstancePort (integer) –
      - Protocol (string) –
      - InstancesSet (list) –
      - (dict) –
      - DeviceIndex (string) –



- **InstanceId** (*string*) –
- **InstanceUniqueId** (*string*) –
- **IpAddress** (*string*) –
- **NetworkId** (*string*) –
- **NetworkInterfaceSet** (*list*) –
- (*dict*) –
- **IpAddress** (*string*) –
- **NetworkInterfaceId** (*string*) –
- **NextMonthAccountingType** (*string*) –
- **PrivateLanName** (*string*) –
- **RemoteAccessVpnGatewaySet** (*list*) –
- (*dict*) –
- **DeviceIndex** (*string*) –
- **IpAddress** (*string*) –
- **RemoteAccessVpnGatewayId** (*string*) –
- **RemoteAccessVpnGatewayName** (*string*) –
- **RouterSet** (*list*) –
- (*dict*) –
- **DeviceIndex** (*string*) –
- **IpAddress** (*string*) –
- **RouterId** (*string*) –
- **RouterName** (*string*) –
- **SharingStatus** (*string*) –
- **State** (*string*) –
- **TagSet** (*list*) –
- (*dict*) –
- **Key** (*string*) –
- **Value** (*string*) –
- **VpnGatewaySet** (*list*) –
- (*dict*) –
- **DeviceIndex** (*string*) –
- **IpAddress** (*string*) –
- **NiftyVpnGatewayName** (*string*) –
- **VpnGatewayId** (*string*) –
- **RequestId** (*string*) –

*computing* / Client / nifty\_describe\_routers

## nifty\_describe\_routers

`computing.Client.nifty_describe_routers(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_describe_routers(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'availability-zone'|'state'|'router-id'|'router-name'|
↪ 'description'|'accountingType'|'type'|'ip-address'|'version'|'latest-version-
↪ information'
```

(continues on next page)

(continued from previous page)

```

    },
],
RouterId=[
    'string',
],
RouterName=[
    'string',
]
)

```

**Parameters**

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **RouterId** (*list*) –
  - (*string*) –
- **RouterName** (*list*) –
  - (*string*) –

**Return type** dict**Returns****Response Syntax**

```

{
  'RequestId': 'string',
  'RouterSet': [
    {
      'AccountingType': 'string',
      'AvailabilityZone': 'string',
      'BackupInformation': {
        'ExpirationDate': datetime(2015, 1, 1),
        'IsBackup': True|False
      },
      'CreatedTime': datetime(2015, 1, 1),
      'Description': 'string',
      'GroupSet': [
        {
          'GroupId': 'string'
        },
      ],
      'NatTableAssociationId': 'string',
      'NatTableId': 'string',
      'NetworkInterfaceSet': [
        {
          'CidrBlock': 'string',
          'Description': 'string',
          'DeviceIndex': 'string',
          'Dhcp': True|False,
          'DhcpConfigId': 'string',
          'DhcpOptionsId': 'string',
          'IpAddress': 'string',
          'NetworkId': 'string',
          'NetworkName': 'string'
        },
      ],
    },
  ],
}

```

(continues on next page)

(continued from previous page)

```

    ],
    'NextMonthAccountingType': 'string',
    'RouteTableAssociationId': 'string',
    'RouteTableId': 'string',
    'RouterId': 'string',
    'RouterName': 'string',
    'State': 'string',
    'TagSet': [
        {
            'Key': 'string',
            'Value': 'string'
        },
    ],
    ],
    'Type': 'string',
    'VersionInformation': {
        'IsLatest': True|False,
        'Version': 'string'
    }
  },
]
}

```

**Response Structure**

- (dict) –
  - **RequestId** (string) –
  - **RouterSet** (list) –
    - \* (dict) –
      - **AccountingType** (string) –
      - **AvailabilityZone** (string) –
      - **BackupInformation** (dict) –
      - **ExpirationDate** (datetime) –
      - **IsBackup** (boolean) –
      - **CreatedTime** (datetime) –
      - **Description** (string) –
      - **GroupSet** (list) –
      - (dict) –
        - **GroupId** (string) –
        - **NatTableAssociationId** (string) –
        - **NatTableId** (string) –
        - **NetworkInterfaceSet** (list) –
        - (dict) –
          - **CidrBlock** (string) –
          - **Description** (string) –
          - **DeviceIndex** (string) –
          - **Dhcp** (boolean) –
          - **DhcpConfigId** (string) –
          - **DhcpOptionsId** (string) –
          - **IpAddress** (string) –
          - **NetworkId** (string) –
          - **NetworkName** (string) –
          - **NextMonthAccountingType** (string) –
          - **RouteTableAssociationId** (string) –
          - **RouteTableId** (string) –
          - **RouterId** (string) –

- **RouterName** (*string*) –
- **State** (*string*) –
- **TagSet** (*list*) –
- (*dict*) –
- **Key** (*string*) –
- **Value** (*string*) –
- **Type** (*string*) –
- **VersionInformation** (*dict*) –
- **IsLatest** (*boolean*) –
- **Version** (*string*) –

*computing* / Client / nifty\_describe\_scaling\_activities

## nifty\_describe\_scaling\_activities

`computing.Client.nifty_describe_scaling_activities(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_describe_scaling_activities(
    ActivityDateFrom='string',
    ActivityDateTo='string',
    AutoScalingGroupName='string',
    Range={
        'All': True|False,
        'EndNumber': 123,
        'StartNumber': 123
    }
)
```

### Parameters

- **ActivityDateFrom** (*string*) –
- **ActivityDateTo** (*string*) –
- **AutoScalingGroupName** (*string*) – [REQUIRED]
- **Range** (*dict*) –
  - **All** (*boolean*) –
  - **EndNumber** (*integer*) –
  - **StartNumber** (*integer*) –

**Return type** dict

### Returns

### Response Syntax

```
{
    'AutoScalingGroupName': 'string',
    'LogSet': [
        {
            'Details': {
                'ChangeInCapacity': 123,
                'CurrentServersCount': 123,
                'Resource': 'string',
                'ResourceValue': 123.0,
                'UpperThreshold': 123.0
            },
            'Process': 'string',
```

(continues on next page)

(continued from previous page)

```

        'Time': datetime(2015, 1, 1)
    },
],
'RequestId': 'string'
}

```

**Response Structure**

- *(dict)* –
  - **AutoScalingGroupName** (*string*) –
  - **LogSet** (*list*) –
    - \* *(dict)* –
      - **Details** (*dict*) –
      - **ChangeInCapacity** (*integer*) –
      - **CurrentServersCount** (*integer*) –
      - **Resource** (*string*) –
      - **ResourceValue** (*float*) –
      - **UpperThreshold** (*float*) –
      - **Process** (*string*) –
      - **Time** (*datetime*) –
  - **RequestId** (*string*) –

*computing* / Client / nifty\_describe\_separate\_instance\_rules

**nifty\_describe\_separate\_instance\_rules**

`computing.Client.nifty_describe_separate_instance_rules(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```

response = client.nifty_describe_separate_instance_rules(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'description'|'instanceId'|'instanceUniqueId'
        },
    ],
    SeparateInstanceRuleName=[
        'string',
    ]
)

```

**Parameters**

- **Filter** (*list*) –
  - *(dict)* –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
  - **SeparateInstanceRuleName** (*list*) –
    - (*string*) –

**Return type** dict

**Returns**

### Response Syntax

```
{
  'RequestId': 'string',
  'SeparateInstanceRulesInfo': [
    {
      'AvailabilityZone': 'string',
      'InstancesSet': [
        {
          'InstanceId': 'string',
          'InstanceUniqueId': 'string'
        },
      ],
      'SeparateInstanceRuleDescription': 'string',
      'SeparateInstanceRuleName': 'string',
      'SeparateInstanceRuleStatus': 'string'
    },
  ]
}
```

### Response Structure

- (dict) –
  - **RequestId** (string) –
  - **SeparateInstanceRulesInfo** (list) –
    - \* (dict) –
      - **AvailabilityZone** (string) –
      - **InstancesSet** (list) –
      - (dict) –
      - **InstanceId** (string) –
      - **InstanceUniqueId** (string) –
      - **SeparateInstanceRuleDescription** (string) –
      - **SeparateInstanceRuleName** (string) –
      - **SeparateInstanceRuleStatus** (string) –

*computing* / Client / `nifty_describe_vpn_gateway_activities`

## nifty\_describe\_vpn\_gateway\_activities

`computing.Client.nifty_describe_vpn_gateway_activities(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_describe_vpn_gateway_activities(
    NiftyVpnGatewayName='string',
    VpnGatewayId='string'
)
```

### Parameters

- **NiftyVpnGatewayName** (string) –
- **VpnGatewayId** (string) –

**Return type** dict

**Returns**

### Response Syntax

```
{
  'AnalyzeResultSet': [
    {
      'AnalyzeCode': 'string',
      'Line': 'string'
    },
  ],
  'Log': 'string',
  'NiftyVpnGatewayName': 'string',
  'RequestId': 'string',
  'VpnGatewayId': 'string'
}
```

### Response Structure

- *(dict)* –
  - **AnalyzeResultSet** (*list*) –
    - \* *(dict)* –
      - **AnalyzeCode** (*string*) –
      - **Line** (*string*) –
  - **Log** (*string*) –
  - **NiftyVpnGatewayName** (*string*) –
  - **RequestId** (*string*) –
  - **VpnGatewayId** (*string*) –

*computing* / Client / nifty\_describe\_web\_proxies

### nifty\_describe\_web\_proxies

`computing.Client.nifty_describe_web_proxies(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_describe_web_proxies(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'router-id'|'router-name'|'listen-network-id'|'listen-network-
→name'|'listen-port'|'proxy-bypass-network-id'|'proxy-bypass-network-name'|
→'option-name-server'
        },
    ],
    RouterId=[
        'string',
    ],
    RouterName=[
        'string',
    ]
)
```

### Parameters

- **Filter** (*list*) –
  - *(dict)* –
    - \* **ListOfRequestValue** (*list*) –

- (string) –
- \* **Name** (string) –
- **RouterId** (list) –
  - (string) –
- **RouterName** (list) –
  - (string) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'RequestId': 'string',
  'WebProxy': [
    {
      'BypassInterface': {
        'NetworkId': 'string',
        'NetworkName': 'string'
      },
      'Description': 'string',
      'ListenInterface': {
        'NetworkId': 'string',
        'NetworkName': 'string'
      },
      'ListenPort': 'string',
      'Option': {
        'NameServer': 'string'
      },
      'RouterId': 'string',
      'RouterName': 'string'
    },
  ]
}
```

#### Response Structure

- (dict) –
  - **RequestId** (string) –
  - **WebProxy** (list) –
    - \* (dict) –
      - **BypassInterface** (dict) –
      - **NetworkId** (string) –
      - **NetworkName** (string) –
      - **Description** (string) –
      - **ListenInterface** (dict) –
      - **NetworkId** (string) –
      - **NetworkName** (string) –
      - **ListenPort** (string) –
      - **Option** (dict) –
      - **NameServer** (string) –
      - **RouterId** (string) –
      - **RouterName** (string) –

*computing* / Client / nifty\_disable\_dhcp



## nifty\_disable\_dhcp

`computing.Client.nifty_disable_dhcp(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_disable_dhcp(
    Agreement=True|False,
    NetworkId='string',
    NetworkName='string',
    RouterId='string',
    RouterName='string'
)
```

### Parameters

- **Agreement** (*boolean*) –
- **NetworkId** (*string*) –
- **NetworkName** (*string*) –
- **RouterId** (*string*) –
- **RouterName** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_disassociate\_nat\_table

## nifty\_disassociate\_nat\_table

`computing.Client.nifty_disassociate_nat_table(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_disassociate_nat_table(
    Agreement=True|False,
    AssociationId='string'
)
```

### Parameters

- **Agreement** (*boolean*) –
- **AssociationId** (*string*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (dict) –
  - **RequestId** (string) –
  - **Return** (boolean) –

*computing* / Client / `nifty_disassociate_route_table_from_elastic_load_balancer`

**nifty\_disassociate\_route\_table\_from\_elastic\_load\_balancer**

`computing.Client.nifty_disassociate_route_table_from_elastic_load_balancer(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.nifty_disassociate_route_table_from_elastic_load_balancer(
    AssociationId='string'
)
```

**Parameters** **AssociationId** (string) – [REQUIRED]

**Return type** dict

**Returns**

**Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (dict) –
  - **RequestId** (string) –
  - **Return** (boolean) –

*computing* / Client / `nifty_disassociate_route_table_from_vpn_gateway`

**nifty\_disassociate\_route\_table\_from\_vpn\_gateway**

`computing.Client.nifty_disassociate_route_table_from_vpn_gateway(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.nifty_disassociate_route_table_from_vpn_gateway(
    Agreement=True|False,
    AssociationId='string'
)
```

**Parameters**

- **Agreement** (boolean) –
- **AssociationId** (string) – [REQUIRED]

**Return type** dict

**Returns**

**Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_enable\_dhcp

**nifty\_enable\_dhcp**

`computing.Client.nifty_enable_dhcp(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.nifty_enable_dhcp(
    Agreement=True|False,
    DhcpConfigId='string',
    DhcpOptionsId='string',
    NetworkId='string',
    NetworkName='string',
    RouterId='string',
    RouterName='string'
)
```

**Parameters**

- **Agreement** (*boolean*) –
- **DhcpConfigId** (*string*) –
- **DhcpOptionsId** (*string*) –
- **NetworkId** (*string*) –
- **NetworkName** (*string*) –
- **RouterId** (*string*) –
- **RouterName** (*string*) –

**Return type** dict

**Returns**

**Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_modify\_address\_attribute

## nifty\_modify\_address\_attribute

`computing.Client.nifty_modify_address_attribute(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_modify_address_attribute(
    Attribute='description',
    PrivateIpAddress='string',
    PublicIp='string',
    Value='string'
)
```

### Parameters

- **Attribute** (*string*) – [REQUIRED]
- **PrivateIpAddress** (*string*) –
- **PublicIp** (*string*) –
- **Value** (*string*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / `nifty_modify_customer_gateway_attribute`

## nifty\_modify\_customer\_gateway\_attribute

`computing.Client.nifty_modify_customer_gateway_attribute(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_modify_customer_gateway_attribute(
    Attribute='niftyCustomerGatewayName'|'niftyCustomerGatewayDescription',
    CustomerGatewayId='string',
    NiftyCustomerGatewayName='string',
    Value='string'
)
```

### Parameters

- **Attribute** (*string*) – [REQUIRED]
- **CustomerGatewayId** (*string*) –
- **NiftyCustomerGatewayName** (*string*) –
- **Value** (*string*) – [REQUIRED]

**Return type** dict

## Returns

### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_modify\_elastic\_load\_balancer\_attributes

## nifty\_modify\_elastic\_load\_balancer\_attributes

`computing.Client.nifty_modify_elastic_load_balancer_attributes(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_modify_elastic_load_balancer_attributes(
    ElasticLoadBalancerId='string',
    ElasticLoadBalancerName='string',
    ElasticLoadBalancerPort=123,
    InstancePort=123,
    LoadBalancerAttributes={
        'ListOfRequestAdditionalAttributes': [
            {
                'Key': 'protocol'|'elasticLoadBalancerPort'|'instancePort'|
→ 'description'|'balancingType'|'sslCertificateId',
                'Value': 'string'
            },
        ],
        'RequestSession': {
            'RequestStickinessPolicy': {
                'Enable': True|False,
                'ExpirationPeriod': 123,
                'Method': '1'|'2'
            },
        },
        'RequestSorryPage': {
            'Enable': True|False,
            'RedirectUrl': 'string'
        },
    },
    Protocol='TCP'|'UDP'|'HTTP'|'HTTPS'
)
```

### Parameters

- **ElasticLoadBalancerId** (*string*) –
- **ElasticLoadBalancerName** (*string*) –
- **ElasticLoadBalancerPort** (*integer*) – [REQUIRED]
- **InstancePort** (*integer*) – [REQUIRED]
- **LoadBalancerAttributes** (*dict*) –
  - **ListOfRequestAdditionalAttributes** (*list*) –

- \* (*dict*) –
  - **Key** (*string*) –
  - **Value** (*string*) –
- **RequestSession** (*dict*) –
  - \* **RequestStickinessPolicy** (*dict*) –
    - **Enable** (*boolean*) –
    - **ExpirationPeriod** (*integer*) –
    - **Method** (*string*) –
- **RequestSorryPage** (*dict*) –
  - \* **Enable** (*boolean*) –
  - \* **RedirectUrl** (*string*) –
- **Protocol** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

#### Response Structure

- (*dict*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*computing* / Client / nifty\_modify\_instance\_snapshot\_attribute

### nifty\_modify\_instance\_snapshot\_attribute

`computing.Client.nifty_modify_instance_snapshot_attribute(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.nifty_modify_instance_snapshot_attribute(
    Attribute='description',
    InstanceSnapshotId='string',
    SnapshotName='string',
    Value='string'
)
```

#### Parameters

- **Attribute** (*string*) – [REQUIRED]
- **InstanceSnapshotId** (*string*) –
- **SnapshotName** (*string*) –
- **Value** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'RequestId': 'string',
```

(continues on next page)

(continued from previous page)

```

    'Return': True|False
}

```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_modify\_key\_pair\_attribute

**nifty\_modify\_key\_pair\_attribute**

`computing.Client.nifty_modify_key_pair_attribute(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```

response = client.nifty_modify_key_pair_attribute(
    Attribute='description',
    KeyName='string',
    Value='string'
)

```

**Parameters**

- **Attribute** (*string*) – [REQUIRED]
- **KeyName** (*string*) – [REQUIRED]
- **Value** (*string*) – [REQUIRED]

**Return type** dict

**Returns****Response Syntax**

```

{
    'Attribute': 'string',
    'RequestId': 'string',
    'Return': True|False,
    'Value': 'string'
}

```

**Response Structure**

- (*dict*) –
  - **Attribute** (*string*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –
  - **Value** (*string*) –

*computing* / Client / nifty\_modify\_private\_lan\_attribute

**nifty\_modify\_private\_lan\_attribute**

`computing.Client.nifty_modify_private_lan_attribute(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.nifty_modify_private_lan_attribute(  
    Attribute='privateLanName'|'cidrBlock'|'accountingType'|'description',  
    NetworkId='string',  
    PrivateLanName='string',  
    Value='string'  
)
```

**Parameters**

- **Attribute** (*string*) – [REQUIRED]
- **NetworkId** (*string*) –
- **PrivateLanName** (*string*) –
- **Value** (*string*) – [REQUIRED]

**Return type** dict

**Returns****Response Syntax**

```
{  
    'RequestId': 'string',  
    'Return': True|False  
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_modify\_router\_attribute

**nifty\_modify\_router\_attribute**

`computing.Client.nifty_modify_router_attribute(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.nifty_modify_router_attribute(  
    Agreement=True|False,  
    Attribute='routerName'|'description'|'accountingType'|'type'|'groupId',  
    RouterId='string',  
    RouterName='string',  
    Value='string'  
)
```

**Parameters**

- **Agreement** (*boolean*) –
- **Attribute** (*string*) – [REQUIRED]
- **RouterId** (*string*) –
- **RouterName** (*string*) –
- **Value** (*string*) – [REQUIRED]

**Return type** dict

**Returns****Response Syntax**



```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_modify\_vpn\_gateway\_attribute

**nifty\_modify\_vpn\_gateway\_attribute**

`computing.Client.nifty_modify_vpn_gateway_attribute(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.nifty_modify_vpn_gateway_attribute(
    Agreement=True|False,
    Attribute='niftyVpnGatewayName'|'niftyVpnGatewayType'|
    →'niftyVpnGatewayDescription'|'niftyVpnGatewayAccountingType'|'groupId',
    NiftyVpnGatewayName='string',
    Value='string',
    VpnGatewayId='string'
)
```

**Parameters**

- **Agreement** (*boolean*) –
- **Attribute** (*string*) – [REQUIRED]
- **NiftyVpnGatewayName** (*string*) –
- **Value** (*string*) – [REQUIRED]
- **VpnGatewayId** (*string*) –

**Return type** dict

**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_modify\_web\_proxy\_attribute

**nifty\_modify\_web\_proxy\_attribute**

`computing.Client.nifty_modify_web_proxy_attribute(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_modify_web_proxy_attribute(  
    Agreement=True|False,  
    Attribute='listenInterface.NetworkId'|'listenInterface.NetworkName'|  
    ↪ 'listenPort'|'bypassInterface.NetworkId'|'bypassInterface.NetworkName'|'option-  
    ↪ nameServer'|'description',  
    RouterId='string',  
    RouterName='string',  
    Value='string'  
)
```

#### Parameters

- **Agreement** (*boolean*) –
- **Attribute** (*string*) –
- **RouterId** (*string*) –
- **RouterName** (*string*) –
- **Value** (*string*) – [REQUIRED]

**Return type** dict

#### Returns

### Response Syntax

```
{  
    'RequestId': 'string',  
    'Return': True|False  
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_reboot\_routers

## nifty\_reboot\_routers

`computing.Client.nifty_reboot_routers (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_reboot_routers(  
    Router=[  
        {  
            'NiftyReboot': 'force'|'true',  
            'RouterId': 'string',  
            'RouterName': 'string'  
        },  
    ],  
)
```

#### Parameters **Router** (*list*) –

- (*dict*) –
  - **NiftyReboot** (*string*) –
  - **RouterId** (*string*) –
  - **RouterName** (*string*) –

**Return type** dict

**Returns**

**Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (dict) –
  - **RequestId** (string) –
  - **Return** (boolean) –

*computing* / Client / nifty\_reboot\_vpn\_gateways

**nifty\_reboot\_vpn\_gateways**

`computing.Client.nifty_reboot_vpn_gateways (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.nifty_reboot_vpn_gateways(
    VpnGateway=[
        {
            'NiftyReboot': 'force'|'true',
            'NiftyVpnGatewayName': 'string',
            'VpnGatewayId': 'string'
        },
    ]
)
```

**Parameters** **VpnGateway** (*list*) –

- (dict) –
  - **NiftyReboot** (string) –
  - **NiftyVpnGatewayName** (string) –
  - **VpnGatewayId** (string) –

**Return type** dict

**Returns**

**Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (dict) –
  - **RequestId** (string) –
  - **Return** (boolean) –

*computing* / Client / nifty\_register\_instances\_with\_elastic\_load\_balancer

## nifty\_register\_instances\_with\_elastic\_load\_balancer

`computing.Client.nifty_register_instances_with_elastic_load_balancer(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_register_instances_with_elastic_load_balancer(  
    ElasticLoadBalancerId='string',  
    ElasticLoadBalancerName='string',  
    ElasticLoadBalancerPort=123,  
    InstancePort=123,  
    Instances=[  
        {  
            'InstanceId': 'string',  
            'InstanceUniqueId': 'string'  
        },  
    ],  
    Protocol='TCP'|'UDP'|'HTTP'|'HTTPS'  
)
```

### Parameters

- **ElasticLoadBalancerId** (*string*) –
- **ElasticLoadBalancerName** (*string*) –
- **ElasticLoadBalancerPort** (*integer*) – [REQUIRED]
- **InstancePort** (*integer*) – [REQUIRED]
- **Instances** (*list*) –
  - (*dict*) –
    - \* **InstanceId** (*string*) –
    - \* **InstanceUniqueId** (*string*) –
- **Protocol** (*string*) – [REQUIRED]

**Return type** dict

### Returns

#### Response Syntax

```
{  
    'NiftyRegisterInstancesWithElasticLoadBalancerResult': 'string',  
    'ResponseMetadata': {  
        'RequestId': 'string'  
    }  
}
```

### Response Structure

- (*dict*) –
  - **NiftyRegisterInstancesWithElasticLoadBalancerResult** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*computing* / Client / `nifty_register_instances_with_separate_instance_rule`

## nifty\_register\_instances\_with\_separate\_instance\_rule

`computing.Client.nifty_register_instances_with_separate_instance_rule(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_register_instances_with_separate_instance_rule(
    InstanceId=[
        'string',
    ],
    InstanceUniqueId=[
        'string',
    ],
    SeparateInstanceRuleName='string'
)
```

**Parameters**

- **InstanceId** (*list*) –
  - (*string*) –
- **InstanceUniqueId** (*list*) –
  - (*string*) –
- **SeparateInstanceRuleName** (*string*) – [REQUIRED]

**Return type** dict**Returns****Response Syntax**

```
{
    'InstancesSet': [
        {
            'InstanceId': 'string',
            'InstanceUniqueId': 'string'
        },
    ],
    'RequestId': 'string'
}
```

**Response Structure**

- (*dict*) –
  - **InstancesSet** (*list*) –
    - \* (*dict*) –
      - **InstanceId** (*string*) –
      - **InstanceUniqueId** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / nifty\_register\_port\_with\_elastic\_load\_balancer**nifty\_register\_port\_with\_elastic\_load\_balancer**computing.Client.**nifty\_register\_port\_with\_elastic\_load\_balancer** (\*\*kwargs)See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.nifty_register_port_with_elastic_load_balancer(
    ElasticLoadBalancerId='string',
    ElasticLoadBalancerName='string',
    Listeners=[
        {
            'BalancingType': 123,
            'Description': 'string',
            'ElasticLoadBalancerPort': 123,
```

(continues on next page)

(continued from previous page)

```

        'InstancePort': 123,
        'Protocol': 'TCP' | 'UDP' | 'HTTP' | 'HTTPS',
        'SSLCertificateId': 'string'
    },
]
)

```

**Parameters**

- **ElasticLoadBalancerId** (*string*) –
- **ElasticLoadBalancerName** (*string*) –
- **Listeners** (*list*) – [REQUIRED]
  - (*dict*) –
    - \* **BalancingType** (*integer*) –
    - \* **Description** (*string*) –
    - \* **ElasticLoadBalancerPort** (*integer*) – [REQUIRED]
    - \* **InstancePort** (*integer*) – [REQUIRED]
    - \* **Protocol** (*string*) – [REQUIRED]
    - \* **SSLCertificateId** (*string*) –

**Return type** dict**Returns****Response Syntax**

```

{
  'NiftyRegisterPortWithElasticLoadBalancerResult': {
    'Listeners': [
      {
        'BalancingType': 123,
        'Description': 'string',
        'ElasticLoadBalancerPort': 123,
        'InstancePort': 123,
        'Protocol': 'string',
        'SSLCertificateId': 'string'
      },
    ],
  },
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}

```

**Response Structure**

- (*dict*) –
  - **NiftyRegisterPortWithElasticLoadBalancerResult** (*dict*) –
    - \* **Listeners** (*list*) –
      - (*dict*) –
        - **BalancingType** (*integer*) –
        - **Description** (*string*) –
        - **ElasticLoadBalancerPort** (*integer*) –
        - **InstancePort** (*integer*) –
        - **Protocol** (*string*) –
        - **SSLCertificateId** (*string*) –
    - **ResponseMetadata** (*dict*) –
      - \* **RequestId** (*string*) –

*computing* / Client / nifty\_register\_routers\_with\_security\_group

## nifty\_register\_routers\_with\_security\_group

`computing.Client.nifty_register_routers_with_security_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_register_routers_with_security_group(
    GroupName='string',
    RouterSet=[
        {
            'RouterId': 'string',
            'RouterName': 'string'
        },
    ]
)
```

### Parameters

- **GroupName** (*string*) – [REQUIRED]
- **RouterSet** (*list*) –
  - (*dict*) –
    - \* **RouterId** (*string*) –
    - \* **RouterName** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
    'RequestId': 'string',
    'RouterSet': [
        {
            'RouterId': 'string',
            'RouterName': 'string'
        },
    ]
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **RouterSet** (*list*) –
    - \* (*dict*) –
      - **RouterId** (*string*) –
      - **RouterName** (*string*) –

*computing* / Client / `nifty_register_vpn_gateways_with_security_group`

## nifty\_register\_vpn\_gateways\_with\_security\_group

`computing.Client.nifty_register_vpn_gateways_with_security_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_register_vpn_gateways_with_security_group(
    GroupName='string',
```

(continues on next page)

(continued from previous page)

```

VpnGatewaySet=[
    {
        'NiftyVpnGatewayName': 'string',
        'VpnGatewayId': 'string'
    },
]
)

```

**Parameters**

- **GroupName** (*string*) – [REQUIRED]
- **VpnGatewaySet** (*list*) –
  - (*dict*) –
    - \* **NiftyVpnGatewayName** (*string*) –
    - \* **VpnGatewayId** (*string*) –

**Return type** dict**Returns****Response Syntax**

```

{
    'RequestId': 'string',
    'VpnGatewaySet': [
        {
            'NiftyVpnGatewayName': 'string',
            'VpnGatewayId': 'string'
        },
    ]
}

```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **VpnGatewaySet** (*list*) –
    - \* (*dict*) –
      - **NiftyVpnGatewayName** (*string*) –
      - **VpnGatewayId** (*string*) –

*computing* / Client / nifty\_release\_router\_backup\_state**nifty\_release\_router\_backup\_state**computing.Client.**nifty\_release\_router\_backup\_state** (\*\*kwargs)See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```

response = client.nifty_release_router_backup_state(
    RouterId='string',
    RouterName='string'
)

```

**Parameters**

- **RouterId** (*string*) –
- **RouterName** (*string*) –

**Return type** dict



**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_release\_vpn\_gateway\_backup\_state

**nifty\_release\_vpn\_gateway\_backup\_state**

`computing.Client.nifty_release_vpn_gateway_backup_state(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.nifty_release_vpn_gateway_backup_state(
    NiftyVpnGatewayName='string',
    VpnGatewayId='string'
)
```

**Parameters**

- **NiftyVpnGatewayName** (*string*) –
- **VpnGatewayId** (*string*) –

**Return type** dict

**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_replace\_dhcp\_config

**nifty\_replace\_dhcp\_config**

`computing.Client.nifty_replace_dhcp_config(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.nifty_replace_dhcp_config(
    Agreement=True|False,
    DhcpConfigId='string',
    NetworkId='string',
    NetworkName='string',
    RouterId='string',
    RouterName='string'
)
```

**Parameters**

- **Agreement** (*boolean*) –
- **DhcpConfigId** (*string*) – [REQUIRED]
- **NetworkId** (*string*) –
- **NetworkName** (*string*) –
- **RouterId** (*string*) –
- **RouterName** (*string*) –

**Return type** dict

**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_replace\_dhcp\_option

**nifty\_replace\_dhcp\_option**

`computing.Client.nifty_replace_dhcp_option(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.nifty_replace_dhcp_option(
    Agreement=True|False,
    DhcpOptionsId='string',
    NetworkId='string',
    NetworkName='string',
    RouterId='string',
    RouterName='string'
)
```

**Parameters**

- **Agreement** (*boolean*) –
- **DhcpOptionsId** (*string*) – [REQUIRED]
- **NetworkId** (*string*) –
- **NetworkName** (*string*) –
- **RouterId** (*string*) –
- **RouterName** (*string*) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (dict) –
  - **RequestId** (string) –
  - **Return** (boolean) –

*computing* / Client / nifty\_replace\_elastic\_load\_balancer\_latest\_version

### nifty\_replace\_elastic\_load\_balancer\_latest\_version

`computing.Client.nifty_replace_elastic_load_balancer_latest_version(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.nifty_replace_elastic_load_balancer_latest_version(
    ElasticLoadBalancerId='string',
    ElasticLoadBalancerName='string',
    NetworkInterface=[
        {
            'ListOfWorkRequestSystemIpAddresses': [
                {
                    'SystemIpAddress': 'string'
                },
            ],
            'NetworkId': 'string'
        },
    ]
)
```

#### Parameters

- **ElasticLoadBalancerId** (string) –
- **ElasticLoadBalancerName** (string) –
- **NetworkInterface** (list) –
  - (dict) –
    - \* **ListOfWorkRequestSystemIpAddresses** (list) –
      - (dict) –
      - **SystemIpAddress** (string) –
    - \* **NetworkId** (string) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'NiftyReplaceElasticLoadBalancerLatestVersionResult': 'string',
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

(continues on next page)

(continued from previous page)

```
}  
}
```

**Response Structure**

- (*dict*) –
  - **NiftyReplaceElasticLoadBalancerLatestVersionResult** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*computing* / Client / nifty\_replace\_elastic\_load\_balancer\_listener\_ssl\_certificate**nifty\_replace\_elastic\_load\_balancer\_listener\_ssl\_certificate**`computing.Client.nifty_replace_elastic_load_balancer_listener_ssl_certificate(**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.nifty_replace_elastic_load_balancer_listener_ssl_certificate(  
    ElasticLoadBalancerId='string',  
    ElasticLoadBalancerName='string',  
    ElasticLoadBalancerPort=123,  
    InstancePort=123,  
    Protocol='HTTPS',  
    SSLCertificateId='string'  
)
```

**Parameters**

- **ElasticLoadBalancerId** (*string*) –
- **ElasticLoadBalancerName** (*string*) –
- **ElasticLoadBalancerPort** (*integer*) – [REQUIRED]
- **InstancePort** (*integer*) – [REQUIRED]
- **Protocol** (*string*) – [REQUIRED]
- **SSLCertificateId** (*string*) – [REQUIRED]

**Return type** dict**Returns****Response Syntax**

```
{  
    'NiftyReplaceElasticLoadBalancerListenerSSLCertificateResult':  
    ↪ 'string',  
    'ResponseMetadata': {  
        'RequestId': 'string'  
    }  
}
```

**Response Structure**

- (*dict*) –
  - **NiftyReplaceElasticLoadBalancerListenerSSLCertificateResult** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*computing* / Client / nifty\_replace\_nat\_rule

## nifty\_replace\_nat\_rule

computing.Client.**nifty\_replace\_nat\_rule** (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```

response = client.nifty_replace_nat_rule(
    Description='string',
    Destination={
        'Port': 123
    },
    InboundInterface={
        'NetworkId': 'string',
        'NetworkName': 'string'
    },
    NatTableId='string',
    NatType='snat'|'dnat',
    OutboundInterface={
        'NetworkId': 'string',
        'NetworkName': 'string'
    },
    Protocol='ALL'|'TCP'|'UDP'|'TCP_UDP'|'ICMP',
    RuleNumber='string',
    Source={
        'Address': 'string',
        'Port': 123
    },
    Translation={
        'Address': 'string',
        'Port': 123
    }
)

```

### Parameters

- **Description** (*string*) –
- **Destination** (*dict*) –
  - **Port** (*integer*) –
- **InboundInterface** (*dict*) –
  - **NetworkId** (*string*) –
  - **NetworkName** (*string*) –
- **NatTableId** (*string*) – [REQUIRED]
- **NatType** (*string*) – [REQUIRED]
- **OutboundInterface** (*dict*) –
  - **NetworkId** (*string*) –
  - **NetworkName** (*string*) –
- **Protocol** (*string*) – [REQUIRED]
- **RuleNumber** (*string*) – [REQUIRED]
- **Source** (*dict*) –
  - **Address** (*string*) –
  - **Port** (*integer*) –
- **Translation** (*dict*) –
  - **Address** (*string*) –
  - **Port** (*integer*) –

**Return type** dict

**Returns**

### Response Syntax

```
{
  'NatRule': {
    'Description': 'string',
    'Destination': {
      'Port': 123
    },
    'InboundInterface': {
      'NetworkId': 'string',
      'NetworkName': 'string'
    },
    'NatType': 'string',
    'OutboundInterface': {
      'NetworkId': 'string',
      'NetworkName': 'string'
    },
    'Protocol': 'string',
    'RuleNumber': 'string',
    'Source': {
      'Address': 'string',
      'Port': 123
    },
    'Translation': {
      'Address': 'string',
      'Port': 123
    }
  },
  'NatTableId': 'string',
  'RequestId': 'string'
}
```

### Response Structure

- (dict) –
  - **NatRule** (dict) –
    - \* **Description** (string) –
    - \* **Destination** (dict) –
      - **Port** (integer) –
    - \* **InboundInterface** (dict) –
      - **NetworkId** (string) –
      - **NetworkName** (string) –
    - \* **NatType** (string) –
    - \* **OutboundInterface** (dict) –
      - **NetworkId** (string) –
      - **NetworkName** (string) –
    - \* **Protocol** (string) –
    - \* **RuleNumber** (string) –
    - \* **Source** (dict) –
      - **Address** (string) –
      - **Port** (integer) –
    - \* **Translation** (dict) –
      - **Address** (string) –
      - **Port** (integer) –
  - **NatTableId** (string) –
  - **RequestId** (string) –

*computing* / Client / nifty\_replace\_nat\_table\_association

## nifty\_replace\_nat\_table\_association

`computing.Client.nifty_replace_nat_table_association(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_replace_nat_table_association(
    Agreement=True|False,
    AssociationId='string',
    NatTableId='string'
)
```

### Parameters

- **Agreement** (*boolean*) –
- **AssociationId** (*string*) – [REQUIRED]
- **NatTableId** (*string*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```
{
    'NewAssociationId': 'string',
    'RequestId': 'string'
}
```

### Response Structure

- (*dict*) –
  - **NewAssociationId** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / `nifty_replace_route_table_association_with_elastic_load_balancer`

## nifty\_replace\_route\_table\_association\_with\_elastic\_load\_balancer

`computing.Client.nifty_replace_route_table_association_with_elastic_load_balancer(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_replace_route_table_association_with_elastic_load_
↪balancer(
    AssociationId='string',
    RouteTableId='string'
)
```

### Parameters

- **AssociationId** (*string*) – [REQUIRED]
- **RouteTableId** (*string*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```
{
    'RequestId': 'string',
```

(continues on next page)

(continued from previous page)

```
'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / `nifty_replace_route_table_association_with_vpn_gateway`

**nifty\_replace\_route\_table\_association\_with\_vpn\_gateway**

`computing.Client.nifty_replace_route_table_association_with_vpn_gateway(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.nifty_replace_route_table_association_with_vpn_gateway(
    Agreement=True|False,
    AssociationId='string',
    RouteTableId='string'
)
```

**Parameters**

- **Agreement** (*boolean*) –
- **AssociationId** (*string*) – [REQUIRED]
- **RouteTableId** (*string*) – [REQUIRED]

**Return type** `dict`

**Returns****Response Syntax**

```
{
    'NewAssociationId': 'string',
    'RequestId': 'string'
}
```

**Response Structure**

- (*dict*) –
  - **NewAssociationId** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / `nifty_replace_router_latest_version`

**nifty\_replace\_router\_latest\_version**

`computing.Client.nifty_replace_router_latest_version(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.nifty_replace_router_latest_version(
    Agreement=True|False,
    RouterId='string',
```

(continues on next page)



(continued from previous page)

```
RouterName='string'
)
```

**Parameters**

- **Agreement** (*boolean*) –
- **RouterId** (*string*) –
- **RouterName** (*string*) –

**Return type** dict**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_replace\_vpn\_gateway\_latest\_version**nifty\_replace\_vpn\_gateway\_latest\_version***computing*.Client.**nifty\_replace\_vpn\_gateway\_latest\_version** (\*\*kwargs)See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.nifty_replace_vpn_gateway_latest_version(
    Agreement=True|False,
    NiftyVpnGatewayName='string',
    VpnGatewayId='string'
)
```

**Parameters**

- **Agreement** (*boolean*) –
- **NiftyVpnGatewayName** (*string*) –
- **VpnGatewayId** (*string*) –

**Return type** dict**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_restore\_instance\_snapshot

## nifty\_restore\_instance\_snapshot

`computing.Client.nifty_restore_instance_snapshot (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_restore_instance_snapshot(  
    InstanceSnapshotId='string',  
    SnapshotName='string'  
)
```

### Parameters

- **InstanceSnapshotId** (*string*) –
- **SnapshotName** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{  
    'RequestId': 'string',  
    'Return': True|False  
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_restore\_router\_previous\_version

## nifty\_restore\_router\_previous\_version

`computing.Client.nifty_restore_router_previous_version (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_restore_router_previous_version(  
    RouterId='string',  
    RouterName='string'  
)
```

### Parameters

- **RouterId** (*string*) –
- **RouterName** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{  
    'RequestId': 'string',  
    'Return': True|False  
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_restore\_vpn\_gateway\_previous\_version

## nifty\_restore\_vpn\_gateway\_previous\_version

`computing.Client.nifty_restore_vpn_gateway_previous_version(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_restore_vpn_gateway_previous_version(
    NiftyVpnGatewayName='string',
    VpnGatewayId='string'
)
```

### Parameters

- **NiftyVpnGatewayName** (*string*) –
- **VpnGatewayId** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_retry\_import\_instance

## nifty\_retry\_import\_instance

`computing.Client.nifty_retry_import_instance(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_retry_import_instance(
    InstanceId='string',
    InstanceUniqueId='string'
)
```

### Parameters

- **InstanceId** (*string*) –
- **InstanceUniqueId** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
    'InstanceId': 'string',
    'InstanceState': 'string',
    'InstanceUniqueId': 'string',
    'RequestId': 'string'
}
```

#### Response Structure

- (*dict*) –
  - **InstanceId** (*string*) –
  - **InstanceState** (*string*) –
  - **InstanceUniqueId** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / `nifty_set_load_balancer_ssl_policies_of_listener`

### `nifty_set_load_balancer_ssl_policies_of_listener`

`computing.Client.nifty_set_load_balancer_ssl_policies_of_listener(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.nifty_set_load_balancer_ssl_policies_of_listener(
    InstancePort=123,
    LoadBalancerName='string',
    LoadBalancerPort=123,
    SSLPolicyId='string',
    SSLPolicyName='string'
)
```

#### Parameters

- **InstancePort** (*integer*) – [REQUIRED]
- **LoadBalancerName** (*string*) – [REQUIRED]
- **LoadBalancerPort** (*integer*) – [REQUIRED]
- **SSLPolicyId** (*string*) –
- **SSLPolicyName** (*string*) –

**Return type** `dict`

#### Returns

#### Response Syntax

```
{
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

#### Response Structure

- (*dict*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*computing* / Client / `nifty_unset_load_balancer_ssl_policies_of_listener`

## nifty\_unset\_load\_balancer\_ssl\_policies\_of\_listener

computing.Client.**nifty\_unset\_load\_balancer\_ssl\_policies\_of\_listener**(\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_unset_load_balancer_ssl_policies_of_listener(
    InstancePort=123,
    LoadBalancerName='string',
    LoadBalancerPort=123
)
```

### Parameters

- **InstancePort** (*integer*) – [REQUIRED]
- **LoadBalancerName** (*string*) – [REQUIRED]
- **LoadBalancerPort** (*integer*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```
{
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

### Response Structure

- (*dict*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*computing* / Client / nifty\_update\_alarm

## nifty\_update\_alarm

computing.Client.**nifty\_update\_alarm**(\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.nifty_update_alarm(
    AlarmCondition='and'|'or',
    Description='string',
    ElasticLoadBalancerName=[
        'string',
    ],
    ElasticLoadBalancerPort=[
        123,
    ],
    ElasticLoadBalancerProtocol=[
        'string',
    ],
    EmailAddress=[
        'string',
    ],
```

(continues on next page)

(continued from previous page)

```

FunctionName='Server'|'LoadBalancer'|'DiskPartition'|'ElasticLoadBalancer',
InstanceId=[
    'string',
],
LoadBalancerName=[
    'string',
],
LoadBalancerPort=[
    123,
],
Partition=[
    'string',
],
Rule=[
    {
        'BreachDuration': 123,
        'DataType': 'string',
        'Threshold': 123.0,
        'UpperLowerCondition': 'upper'|'lower'
    },
],
RuleName='string',
RuleNameUpdate='string'
)

```

**Parameters**

- **AlarmCondition** (*string*) –
- **Description** (*string*) –
- **ElasticLoadBalancerName** (*list*) –  
– (*string*) –
- **ElasticLoadBalancerPort** (*list*) –  
– (*integer*) –
- **ElasticLoadBalancerProtocol** (*list*) –  
– (*string*) –
- **EmailAddress** (*list*) –  
– (*string*) –
- **FunctionName** (*string*) – [REQUIRED]
- **InstanceId** (*list*) –  
– (*string*) –
- **LoadBalancerName** (*list*) –  
– (*string*) –
- **LoadBalancerPort** (*list*) –  
– (*integer*) –
- **Partition** (*list*) –  
– (*string*) –
- **Rule** (*list*) – [REQUIRED]  
– (*dict*) –
  - \* **BreachDuration** (*integer*) – [REQUIRED]
  - \* **DataType** (*string*) –
  - \* **Threshold** (*float*) –
  - \* **UpperLowerCondition** (*string*) –
- **RuleName** (*string*) – [REQUIRED]
- **RuleNameUpdate** (*string*) –

**Return type** dict**Returns**

## Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

## Response Structure

- (dict) –
  - **RequestId** (string) –
  - **Return** (boolean) –

*computing* / Client / nifty\_update\_auto\_scaling\_group

## nifty\_update\_auto\_scaling\_group

`computing.Client.nifty_update_auto_scaling_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

## Request Syntax

```
response = client.nifty_update_auto_scaling_group(
    AutoScalingGroupName='string',
    AutoScalingGroupNameUpdate='string',
    ChangeInCapacity=123,
    DefaultCooldown=123,
    Description='string',
    ImageId='string',
    InstanceLifecycleLimit=123,
    InstanceType='e-mini'|'h2-mini'|'mini'|'c-small'|'e-small'|'h2-small'|'small'|
    ↪ 'c-small2'|'e-small2'|'h2-small2'|'small2'|'c-small4'|'e-small4'|'h2-small4'|
    ↪ 'small4'|'e-small8'|'h2-small8'|'small8'|'e-small16'|'h2-small16'|'small16'|'c-
    ↪ medium'|'e-medium'|'h2-medium'|'medium'|'c-medium4'|'e-medium4'|'h2-medium4'|
    ↪ 'medium4'|'c-medium8'|'e-medium8'|'h2-medium8'|'medium8'|'e-medium16'|'h2-
    ↪ medium16'|'medium16'|'e-medium24'|'h2-medium24'|'medium24'|'c-large'|'e-large'|
    ↪ 'h2-large'|'large'|'c-large8'|'e-large8'|'h2-large8'|'large8'|'e-large16'|'h2-
    ↪ large16'|'large16'|'e-large24'|'h2-large24'|'large24'|'e-large32'|'h2-large32'|
    ↪ 'large32'|'e-extra-large8'|'h2-extra-large8'|'extra-large8'|'e-extra-large16'|
    ↪ 'h2-extra-large16'|'extra-large16'|'e-extra-large24'|'h2-extra-large24'|'extra-
    ↪ large24'|'e-extra-large32'|'h2-extra-large32'|'extra-large32'|'e-extra-large48'|
    ↪ 'h2-extra-large48'|'extra-large48'|'e-double-large16'|'h2-double-large16'|
    ↪ 'double-large16'|'e-double-large24'|'h2-double-large24'|'double-large24'|'e-
    ↪ double-large32'|'h2-double-large32'|'double-large32'|'e-double-large48'|'h2-
    ↪ double-large48'|'double-large48'|'e-double-large64'|'h2-double-large64'|'double-
    ↪ large64'|'e-double-large96'|'h2-double-large96'|'double-large96'|'h2-triple-
    ↪ large32'|'triple-large32'|'h2-triple-large48'|'triple-large48'|'h2-triple-
    ↪ large64'|'triple-large64'|'h2-triple-large96'|'triple-large96'|'h2-triple-
    ↪ large128'|'triple-large128'|'h2-quad-large64'|'quad-large64'|'h2-quad-large96'|
    ↪ 'quad-large96'|'h2-quad-large128'|'quad-large128'|'h2-septa-large128'|'septa-
    ↪ large128',
    LoadBalancers=[
        {
            'InstancePort': 123,
            'LoadBalancerPort': 123,
            'Name': 'string'
        },
    ],
```

(continues on next page)

(continued from previous page)

```

MaxSize=123,
MinSize=123,
Scaleout=123,
ScaleoutCondition='or'|'and',
ScalingSchedule=[
    {
        'RequestDDay': {
            'EndingDDay': 'string',
            'StartingDDay': 'string'
        },
        'RequestDay': {
            'SetFriday': '0'|'1',
            'SetMonday': '0'|'1',
            'SetSaturday': '0'|'1',
            'SetSunday': '0'|'1',
            'SetThursday': '0'|'1',
            'SetTuesday': '0'|'1',
            'SetWednesday': '0'|'1'
        },
        'RequestMonth': {
            'EndingMonth': 'string',
            'StartingMonth': 'string'
        },
        'RequestTimeZone': {
            'EndingTimeZone': 'string',
            'StartingTimeZone': 'string'
        }
    },
],
ScalingTrigger=[
    {
        'BreachDuration': 123,
        'Resource': 'Server-cpu'|'Server-memory'|'Server-network'|
→ 'LoadBalancer-network',
        'UpperThreshold': 123.0
    },
],
SecurityGroup=[
    'string',
]
)

```

### Parameters

- **AutoScalingGroupName** (*string*) – [REQUIRED]
- **AutoScalingGroupNameUpdate** (*string*) –
- **ChangeInCapacity** (*integer*) – [REQUIRED]
- **DefaultCooldown** (*integer*) –
- **Description** (*string*) –
- **ImageId** (*string*) –
- **InstanceLifecycleLimit** (*integer*) –
- **InstanceType** (*string*) –
- **LoadBalancers** (*list*) –
  - (*dict*) –
  - \* **InstancePort** (*integer*) –
  - \* **LoadBalancerPort** (*integer*) –
  - \* **Name** (*string*) –



- **MaxSize** (*integer*) – [REQUIRED]
- **MinSize** (*integer*) – [REQUIRED]
- **Scaleout** (*integer*) –
- **ScaleoutCondition** (*string*) – [REQUIRED]
- **ScalingSchedule** (*list*) –
  - (*dict*) –
    - \* **RequestDDay** (*dict*) –
      - **EndingDDay** (*string*) –
      - **StartingDDay** (*string*) –
    - \* **RequestDay** (*dict*) –
      - **SetFriday** (*string*) –
      - **SetMonday** (*string*) –
      - **SetSaturday** (*string*) –
      - **SetSunday** (*string*) –
      - **SetThursday** (*string*) –
      - **SetTuesday** (*string*) –
      - **SetWednesday** (*string*) –
    - \* **RequestMonth** (*dict*) –
      - **EndingMonth** (*string*) –
      - **StartingMonth** (*string*) –
    - \* **RequestTimeZone** (*dict*) –
      - **EndingTimeZone** (*string*) –
      - **StartingTimeZone** (*string*) –
- **ScalingTrigger** (*list*) – [REQUIRED]
  - (*dict*) –
    - \* **BreachDuration** (*integer*) –
    - \* **Resource** (*string*) – [REQUIRED]
    - \* **UpperThreshold** (*float*) – [REQUIRED]
- **SecurityGroup** (*list*) –
  - (*string*) –

Return type dict

Returns

#### Response Syntax

```
{
  'RequestId': 'string',
  'Return': True|False
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_update\_elastic\_load\_balancer

#### nifty\_update\_elastic\_load\_balancer

`computing.Client.nifty_update_elastic_load_balancer(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.nifty_update_elastic_load_balancer(  
    AccountingTypeUpdate=123,  
    ElasticLoadBalancerId='string',  
    ElasticLoadBalancerName='string',  
    ElasticLoadBalancerNameUpdate='string',  
    NetworkVolumeUpdate=123  
)
```

**Parameters**

- **AccountingTypeUpdate** (*integer*) –
- **ElasticLoadBalancerId** (*string*) –
- **ElasticLoadBalancerName** (*string*) –
- **ElasticLoadBalancerNameUpdate** (*string*) –
- **NetworkVolumeUpdate** (*integer*) –

**Return type** dict

**Returns****Response Syntax**

```
{  
    'ResponseMetadata': {  
        'RequestId': 'string'  
    }  
}
```

**Response Structure**

- (*dict*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*computing* / Client / nifty\_update\_instance\_network\_interfaces

**nifty\_update\_instance\_network\_interfaces**

`computing.Client.nifty_update_instance_network_interfaces` (\*\**kwargs*)

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.nifty_update_instance_network_interfaces(  
    InstanceId='string',  
    NetworkInterface=[  
        {  
            'DeviceIndex': 123,  
            'IpAddress': 'string',  
            'ListOfRequestSecurityGroupId': [  
                'string',  
            ],  
            'NetworkId': 'string',  
            'NetworkName': 'string'  
        },  
    ],  
    NiftyReboot='force'|'true'|'false'  
)
```

**Parameters**

- **InstanceId** (*string*) – [REQUIRED]

- **NetworkInterface** (*list*) –
  - (*dict*) –
    - \* **DeviceIndex** (*integer*) –
    - \* **IpAddress** (*string*) –
    - \* **ListOfRequestSecurityGroupId** (*list*) –
      - (*string*) –
    - \* **NetworkId** (*string*) –
    - \* **NetworkName** (*string*) –
- **NiftyReboot** (*string*) –

Return type dict

Returns

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_update\_router\_network\_interfaces

### nifty\_update\_router\_network\_interfaces

`computing.Client.nifty_update_router_network_interfaces (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.nifty_update_router_network_interfaces(
    Agreement=True|False,
    NetworkInterface=[
        {
            'DeviceIndex': 123,
            'Dhcp': True|False,
            'DhcpConfigId': 'string',
            'DhcpOptionsId': 'string',
            'IpAddress': 'string',
            'ListOfRequestSecurityGroupId': [
                'string',
            ],
            'NetworkId': 'string',
            'NetworkName': 'string'
        },
    ],
    NiftyReboot='force'|'true',
    RouterId='string',
    RouterName='string'
)
```

#### Parameters

- **Agreement** (*boolean*) –
- **NetworkInterface** (*list*) –

- (dict) –
  - \* **DeviceIndex** (*integer*) –
  - \* **Dhcp** (*boolean*) –
  - \* **DhcpConfigId** (*string*) –
  - \* **DhcpOptionsId** (*string*) –
  - \* **IpAddress** (*string*) –
  - \* **ListOfRequestSecurityGroupId** (*list*) –
    - (*string*) –
  - \* **NetworkId** (*string*) –
  - \* **NetworkName** (*string*) –
- **NiftyReboot** (*string*) –
- **RouterId** (*string*) –
- **RouterName** (*string*) –

Return type dict

Returns

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (dict) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / nifty\_update\_separate\_instance\_rule

### nifty\_update\_separate\_instance\_rule

`computing.Client.nifty_update_separate_instance_rule(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.nifty_update_separate_instance_rule(
    SeparateInstanceRuleDescriptionUpdate='string',
    SeparateInstanceRuleName='string',
    SeparateInstanceRuleNameUpdate='string'
)
```

#### Parameters

- **SeparateInstanceRuleDescriptionUpdate** (*string*) –
- **SeparateInstanceRuleName** (*string*) – [REQUIRED]
- **SeparateInstanceRuleNameUpdate** (*string*) –

Return type dict

Returns

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': 'string'
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*string*) –

*computing* / Client / nifty\_update\_vpn\_gateway\_network\_interfaces

**nifty\_update\_vpn\_gateway\_network\_interfaces**

`computing.Client.nifty_update_vpn_gateway_network_interfaces(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.nifty_update_vpn_gateway_network_interfaces(
    Agreement=True|False,
    NetworkInterface={
        'IpAddress': 'string'
    },
    NiftyReboot='force'|'true',
    NiftyVpnGatewayName='string',
    VpnGatewayId='string'
)
```

**Parameters**

- **Agreement** (*boolean*) –
- **NetworkInterface** (*dict*) –
  - **IpAddress** (*string*) –
- **NiftyReboot** (*string*) –
- **NiftyVpnGatewayName** (*string*) –
- **VpnGatewayId** (*string*) –

**Return type** `dict`

**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / reboot\_instances

**reboot\_instances**

`computing.Client.reboot_instances(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.reboot_instances(  
    Force=True|False,  
    InstanceId=[  
        'string',  
    ],  
    NiftyIsBios=True|False,  
    Tenancy=[  
        'string',  
    ],  
    UserData={  
        'Content': 'string',  
        'Encoding': 'string'  
    }  
)
```

**Parameters**

- **Force** (*boolean*) –
- **InstanceId** (*list*) – [REQUIRED]
  - (*string*) –
- **NiftyIsBios** (*boolean*) –
- **Tenancy** (*list*) –
  - (*string*) –
- **UserData** (*dict*) –
  - **Content** (*string*) –
  - **Encoding** (*string*) –

**Return type** dict

**Returns****Response Syntax**

```
{  
    'RequestId': 'string',  
    'Return': True|False  
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / reboot\_remote\_access\_vpn\_gateway

**reboot\_remote\_access\_vpn\_gateway**

`computing.Client.reboot_remote_access_vpn_gateway(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.reboot_remote_access_vpn_gateway(  
    NiftyReboot='force'|'true',  
    RemoteAccessVpnGatewayId='string'  
)
```

**Parameters**

- **NiftyReboot** (*string*) –

- **RemoteAccessVpnGatewayId** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / refresh\_instance\_backup\_rule

### refresh\_instance\_backup\_rule

`computing.Client.refresh_instance_backup_rule(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.refresh_instance_backup_rule(
    InstanceBackupRuleId='string'
)
```

**Parameters** **InstanceBackupRuleId** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / register\_corporate\_info\_for\_certificate

### register\_corporate\_info\_for\_certificate

`computing.Client.register_corporate_info_for_certificate(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.register_corporate_info_for_certificate(
    Agreement=True|False,
    AlphabetName1='string',
```

(continues on next page)

(continued from previous page)

```

AlphabetName2='string',
City='string',
CorpGrade='string',
CorpName='string',
DivisionName='string',
EmailAddress='string',
KanaName1='string',
KanaName2='string',
Name1='string',
Name2='string',
PhoneNumber='string',
PostName='string',
Pref='string',
PresidentName1='string',
PresidentName2='string',
TdbCode='string',
Zip1='string',
Zip2='string'
)

```

### Parameters

- **Agreement** (*boolean*) – [REQUIRED]
- **AlphabetName1** (*string*) – [REQUIRED]
- **AlphabetName2** (*string*) – [REQUIRED]
- **City** (*string*) – [REQUIRED]
- **CorpGrade** (*string*) – [REQUIRED]
- **CorpName** (*string*) – [REQUIRED]
- **DivisionName** (*string*) – [REQUIRED]
- **EmailAddress** (*string*) – [REQUIRED]
- **KanaName1** (*string*) – [REQUIRED]
- **KanaName2** (*string*) – [REQUIRED]
- **Name1** (*string*) – [REQUIRED]
- **Name2** (*string*) – [REQUIRED]
- **PhoneNumber** (*string*) – [REQUIRED]
- **PostName** (*string*) – [REQUIRED]
- **Pref** (*string*) – [REQUIRED]
- **PresidentName1** (*string*) – [REQUIRED]
- **PresidentName2** (*string*) – [REQUIRED]
- **TdbCode** (*string*) –
- **Zip1** (*string*) – [REQUIRED]
- **Zip2** (*string*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```

{
  'AlphabetName1': 'string',
  'AlphabetName2': 'string',
  'City': 'string',
  'CorpGrade': 'string',
  'CorpName': 'string',
  'DivisionName': 'string',
  'EmailAddress': 'string',
  'KanaName1': 'string',

```

(continues on next page)



(continued from previous page)

```

'KanaName2': 'string',
'Name1': 'string',
'Name2': 'string',
'PhoneNumber': 'string',
'PostName': 'string',
'Pref': 'string',
'PresidentName1': 'string',
'PresidentName2': 'string',
'RequestId': 'string',
'TdbCode': 'string',
'Zip1': 'string',
'Zip2': 'string'
}

```

**Response Structure**

- (dict) –
  - **AlphabetName1** (string) –
  - **AlphabetName2** (string) –
  - **City** (string) –
  - **CorpGrade** (string) –
  - **CorpName** (string) –
  - **DivisionName** (string) –
  - **EmailAddress** (string) –
  - **KanaName1** (string) –
  - **KanaName2** (string) –
  - **Name1** (string) –
  - **Name2** (string) –
  - **PhoneNumber** (string) –
  - **PostName** (string) –
  - **Pref** (string) –
  - **PresidentName1** (string) –
  - **PresidentName2** (string) –
  - **RequestId** (string) –
  - **TdbCode** (string) –
  - **Zip1** (string) –
  - **Zip2** (string) –

*computing* / Client / register\_instances\_with\_load\_balancer

**register\_instances\_with\_load\_balancer**

`computing.Client.register_instances_with_load_balancer(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```

response = client.register_instances_with_load_balancer(
    InstancePort=123,
    Instances=[
        {
            'InstanceId': 'string'
        },
    ],
    LoadBalancerName='string',

```

(continues on next page)

(continued from previous page)

```
LoadBalancerPort=123
)
```

**Parameters**

- **InstancePort** (*integer*) – [REQUIRED]
- **Instances** (*list*) – [REQUIRED]
  - (*dict*) –
    - \* **InstanceId** (*string*) – [REQUIRED]
- **LoadBalancerName** (*string*) – [REQUIRED]
- **LoadBalancerPort** (*integer*) – [REQUIRED]

**Return type** dict**Returns****Response Syntax**

```
{
  'RegisterInstancesWithLoadBalancerResult': {
    'Instances': [
      {
        'InstanceId': 'string',
        'InstanceUniqueId': 'string'
      },
    ],
  },
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

**Response Structure**

- (*dict*) –
  - **RegisterInstancesWithLoadBalancerResult** (*dict*) –
    - \* **Instances** (*list*) –
      - (*dict*) –
        - **InstanceId** (*string*) –
        - **InstanceUniqueId** (*string*) –
    - **ResponseMetadata** (*dict*) –
      - \* **RequestId** (*string*) –

*computing* / Client / register\_instances\_with\_security\_group**register\_instances\_with\_security\_group**computing.Client.**register\_instances\_with\_security\_group** (\*\*kwargs)See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.register_instances_with_security_group(
    GroupName='string',
    InstanceId=[
        'string',
    ]
)
```

**Parameters**

- **GroupName** (*string*) – [REQUIRED]
- **InstanceId** (*list*) – [REQUIRED]
  - (*string*) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'InstancesSet': [
        {
            'InstanceId': 'string'
        },
    ],
    'RequestId': 'string'
}
```

#### Response Structure

- (*dict*) –
  - **InstancesSet** (*list*) –
    - \* (*dict*) –
      - **InstanceId** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / register\_port\_with\_load\_balancer

### register\_port\_with\_load\_balancer

`computing.Client.register_port_with_load_balancer(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.register_port_with_load_balancer(
    Listeners=[
        {
            'BalancingType': 123,
            'InstancePort': 123,
            'LoadBalancerPort': 123,
            'Protocol': 'HTTP'|'HTTPS'|'FTP'
        },
    ],
    LoadBalancerName='string'
)
```

#### Parameters

- **Listeners** (*list*) –
  - (*dict*) –
    - \* **BalancingType** (*integer*) –
    - \* **InstancePort** (*integer*) –
    - \* **LoadBalancerPort** (*integer*) –
    - \* **Protocol** (*string*) –
- **LoadBalancerName** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'RegisterPortWithLoadBalancerResult': {
    'Listeners': [
      {
        'BalancingType': 123,
        'InstancePort': 123,
        'LoadBalancerPort': 123,
        'Protocol': 'string'
      },
    ],
  },
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

#### Response Structure

- *(dict)* –
  - **RegisterPortWithLoadBalancerResult** (*dict*) –
    - \* **Listeners** (*list*) –
      - *(dict)* –
      - **BalancingType** (*integer*) –
      - **InstancePort** (*integer*) –
      - **LoadBalancerPort** (*integer*) –
      - **Protocol** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*computing* / Client / `release_address`

### `release_address`

`computing.Client.release_address` (*\*\*kwargs*)

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.release_address(
    AllocationId='string',
    PrivateIpAddress='string',
    PublicIp='string'
)
```

#### Parameters

- **AllocationId** (*string*) –
- **PrivateIpAddress** (*string*) –
- **PublicIp** (*string*) –

**Return type** dict

#### Returns

#### Response Syntax

```
{
  'RequestId': 'string',
  'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / `release_multi_ip_addresses`

**release\_multi\_ip\_addresses**

`computing.Client.release_multi_ip_addresses(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.release_multi_ip_addresses(
    IpAddress=[
        'string',
    ],
    MultiIpAddressGroupId='string'
)
```

**Parameters**

- **IpAddress** (*list*) – [REQUIRED]
  - (*string*) –
- **MultiIpAddressGroupId** (*string*) – [REQUIRED]

**Return type** dict

**Returns****Response Syntax**

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / `replace_remote_access_vpn_gateway_latest_version`

**replace\_remote\_access\_vpn\_gateway\_latest\_version**

`computing.Client.replace_remote_access_vpn_gateway_latest_version(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.replace_remote_access_vpn_gateway_latest_version(
    RemoteAccessVpnGatewayId='string'
)
```

**Parameters** **RemoteAccessVpnGatewayId** (*string*) – [REQUIRED]

**Return type** dict

## Returns

### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / `replace_route`

## `replace_route`

`computing.Client.replace_route(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.replace_route(
    DestinationCidrBlock='string',
    GatewayId='string',
    InstanceId='string',
    IpAddress='string',
    NetworkId='string',
    NetworkInterfaceId='string',
    NetworkName='string',
    RouteTableId='string',
    VpcPeeringConnectionId='string'
)
```

### Parameters

- **DestinationCidrBlock** (*string*) – [REQUIRED]
- **GatewayId** (*string*) –
- **InstanceId** (*string*) –
- **IpAddress** (*string*) –
- **NetworkId** (*string*) –
- **NetworkInterfaceId** (*string*) –
- **NetworkName** (*string*) –
- **RouteTableId** (*string*) – [REQUIRED]
- **VpcPeeringConnectionId** (*string*) –

Return type `dict`

## Returns

### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

### Response Structure

- (*dict*) –

- **RequestId** (*string*) –
- **Return** (*boolean*) –

*computing* / Client / `replace_route_table_association`

## replace\_route\_table\_association

`computing.Client.replace_route_table_association(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.replace_route_table_association(
    Agreement=True|False,
    AssociationId='string',
    RouteTableId='string'
)
```

### Parameters

- **Agreement** (*boolean*) –
- **AssociationId** (*string*) – [REQUIRED]
- **RouteTableId** (*string*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```
{
    'NewAssociationId': 'string',
    'RequestId': 'string'
}
```

### Response Structure

- (*dict*) –
  - **NewAssociationId** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / `revoke_security_group_ingress`

## revoke\_security\_group\_ingress

`computing.Client.revoke_security_group_ingress(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.revoke_security_group_ingress(
    GroupName='string',
    IpPermissions=[
        {
            'FromPort': 123,
            'InOut': 'IN'|'OUT',
            'IpProtocol': 'ANY'|'TCP'|'UDP'|'ICMP'|'SSH'|'HTTP'|'HTTPS'|'RDP'|'GRE
→ '| 'ESP'|'AH'|'VRRP'|'L2TP'|'ICMPv6-all',
            'ListOfRequestGroups': [
                {
```

(continues on next page)

(continued from previous page)

```

        'GroupName': 'string',
        'UserId': 'string'
    },
],
'ListOfRequestIpRanges': [
    {
        'CidrIp': 'string'
    },
],
'ToPort': 123
},
],
UserId='string'
)

```

**Parameters**

- **GroupName** (*string*) – [REQUIRED]
- **IpPermissions** (*list*) –
  - (*dict*) –
    - \* **FromPort** (*integer*) –
    - \* **InOut** (*string*) –
    - \* **IpProtocol** (*string*) –
    - \* **ListOfRequestGroups** (*list*) –
      - (*dict*) –
      - **GroupName** (*string*) –
      - **UserId** (*string*) –
    - \* **ListOfRequestIpRanges** (*list*) –
      - (*dict*) –
      - **CidrIp** (*string*) –
    - \* **ToPort** (*integer*) –
- **UserId** (*string*) –

**Return type** dict**Returns****Response Syntax**

```

{
    'RequestId': 'string',
    'Return': True|False
}

```

**Response Structure**

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / run\_instances**run\_instances***computing*.Client.**run\_instances** (*\*\*kwargs*)See also: [NIFCLOUD API Documentation](#)**Request Syntax**



```

response = client.run_instances(
    AccountingType='1'|'2',
    AddressingType='string',
    Admin='string',
    Agreement=True|False,
    BlockDeviceMapping=[
        {
            'DeviceName': 'string',
            'RequestEbs': {
                'DeleteOnTermination': True|False,
                'NoDevice': True|False,
                'SnapshotId': 'string',
                'VolumeSize': 123
            },
            'VirtualName': 'string'
        },
    ],
    Description='string',
    DisableApiTermination=True|False,
    ImageId='string',
    InstanceId='string',
    InstanceInitiatedShutdownBehavior='string',
    InstanceType='e-mini'|'h2-mini'|'mini'|'c-small'|'e-small'|'h2-small'|'small'|
↪ 'c-small2'|'e-small2'|'h2-small2'|'small2'|'c-small4'|'e-small4'|'h2-small4'|
↪ 'small4'|'e-small8'|'h2-small8'|'small8'|'e-small16'|'h2-small16'|'small16'|'c-
↪ medium'|'e-medium'|'h2-medium'|'medium'|'c-medium4'|'e-medium4'|'h2-medium4'|
↪ 'medium4'|'c-medium8'|'e-medium8'|'h2-medium8'|'medium8'|'e-medium16'|'h2-
↪ medium16'|'medium16'|'e-medium24'|'h2-medium24'|'medium24'|'c-large'|'e-large'|
↪ 'h2-large'|'large'|'c-large8'|'e-large8'|'h2-large8'|'large8'|'e-large16'|'h2-
↪ large16'|'large16'|'e-large24'|'h2-large24'|'large24'|'e-large32'|'h2-large32'|
↪ 'large32'|'e-extra-large8'|'h2-extra-large8'|'extra-large8'|'e-extra-large16'|
↪ 'h2-extra-large16'|'extra-large16'|'e-extra-large24'|'h2-extra-large24'|'extra-
↪ large24'|'e-extra-large32'|'h2-extra-large32'|'extra-large32'|'e-extra-large48'|
↪ 'h2-extra-large48'|'extra-large48'|'e-double-large16'|'h2-double-large16'|
↪ 'double-large16'|'e-double-large24'|'h2-double-large24'|'double-large24'|'e-
↪ double-large32'|'h2-double-large32'|'double-large32'|'e-double-large48'|'h2-
↪ double-large48'|'double-large48'|'e-double-large64'|'h2-double-large64'|'double-
↪ large64'|'e-double-large96'|'h2-double-large96'|'double-large96'|'h2-triple-
↪ large32'|'triple-large32'|'h2-triple-large48'|'triple-large48'|'h2-triple-
↪ large64'|'triple-large64'|'h2-triple-large96'|'triple-large96'|'h2-triple-
↪ large128'|'triple-large128'|'h2-quad-large64'|'quad-large64'|'h2-quad-large96'|
↪ 'quad-large96'|'h2-quad-large128'|'quad-large128'|'h2-septa-large128'|'septa-
↪ large128',
    IpType='static'|'elastic'|'none',
    KernelId='string',
    KeyName='string',
    License=[
        {
            'LicenseName': 'RDS'|'Office(Std)|'Office(Pro Plus)',
            'LicenseNum': 'string'
        },
    ],
    MaxCount=123,
    MinCount=123,
    Monitoring={
        'Enabled': True|False
    },

```

(continues on next page)

(continued from previous page)

```

NetworkInterface=[
    {
        'DeviceIndex': 123,
        'IpAddress': 'string',
        'ListOfRequestSecurityGroupId': [
            'string',
        ],
        'NetworkId': 'string',
        'NetworkName': 'string'
    },
],
Password='string',
Placement={
    'AvailabilityZone': 'string',
    'GroupName': 'string'
},
PublicIp='string',
RamdiskId='string',
SecurityGroup=[
    'string',
],
SubnetId='string',
UserData={
    'Content': 'string',
    'Encoding': 'string'
}
)

```

### Parameters

- **AccountingType** (*string*) –
- **AddressingType** (*string*) –
- **Admin** (*string*) –
- **Agreement** (*boolean*) –
- **BlockDeviceMapping** (*list*) –
  - (*dict*) –
    - \* **DeviceName** (*string*) –
    - \* **RequestEbs** (*dict*) –
      - **DeleteOnTermination** (*boolean*) –
      - **NoDevice** (*boolean*) –
      - **SnapshotId** (*string*) –
      - **VolumeSize** (*integer*) –
    - \* **VirtualName** (*string*) –
- **Description** (*string*) –
- **DisableApiTermination** (*boolean*) –
- **ImageId** (*string*) – [REQUIRED]
- **InstanceId** (*string*) –
- **InstanceInitiatedShutdownBehavior** (*string*) –
- **InstanceType** (*string*) –
- **IpType** (*string*) –
- **KernelId** (*string*) –
- **KeyName** (*string*) –
- **License** (*list*) –
  - (*dict*) –
    - \* **LicenseName** (*string*) –
    - \* **LicenseNum** (*string*) –

- **MaxCount** (*integer*) –
- **MinCount** (*integer*) –
- **Monitoring** (*dict*) –
  - **Enabled** (*boolean*) –
- **NetworkInterface** (*list*) –
  - (*dict*) –
    - \* **DeviceIndex** (*integer*) –
    - \* **IpAddress** (*string*) –
    - \* **ListOfRequestSecurityGroupId** (*list*) –
      - (*string*) –
    - \* **NetworkId** (*string*) –
    - \* **NetworkName** (*string*) –
- **Password** (*string*) –
- **Placement** (*dict*) –
  - **AvailabilityZone** (*string*) –
  - **GroupName** (*string*) –
- **PublicIp** (*string*) –
- **RamdiskId** (*string*) –
- **SecurityGroup** (*list*) –
  - (*string*) –
- **SubnetId** (*string*) –
- **UserData** (*dict*) –
  - **Content** (*string*) –
  - **Encoding** (*string*) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'GroupSet': [
    {
      'GroupId': 'string'
    },
  ],
  'InstancesSet': [
    {
      'AccountingType': 'string',
      'Admin': 'string',
      'Architecture': 'string',
      'BlockDeviceMapping': [
        {
          'DeviceName': 'string',
          'Ebs': {
            'DeleteOnTermination': 'string',
            'Status': 'string',
            'VolumeId': 'string',
            'VolumeUniqueId': 'string'
          }
        },
      ],
      'Description': 'string',
      'DnsName': 'string',
      'ImageId': 'string',
      'InstanceId': 'string',
      'InstanceState': {
```

(continues on next page)

(continued from previous page)

```

        'Code': 123,
        'Name': 'string'
    },
    'InstanceType': 'string',
    'InstanceUniqueId': 'string',
    'IpAddress': 'string',
    'IpAddressV6': 'string',
    'IpType': 'string',
    'IsoImage': [
        {
            'IsoImageId': 'string',
            'IsoImageName': 'string'
        },
    ],
    'KeyName': 'string',
    'LaunchTime': datetime(2015, 1, 1),
    'Monitoring': {
        'State': 'string'
    },
    'NetworkInterfaceSet': [
        {
            'Association': {
                'IpOwnerId': 'string',
                'PublicDnsName': 'string',
                'PublicIp': 'string'
            },
            'Attachment': {
                'AttachTime': 'string',
                'AttachmentID': 'string',
                'DeleteOnTermination': 'string',
                'DeviceIndex': 'string',
                'Status': 'string'
            },
            'Description': 'string',
            'GroupSet': [
                {
                    'GroupId': 'string'
                },
            ],
            'NetworkInterfaceId': 'string',
            'NiftyNetworkId': 'string',
            'NiftyNetworkName': 'string',
            'OwnerId': 'string',
            'PrivateDnsName': 'string',
            'PrivateIpAddressesSet': [
                {
                    'Association': {
                        'IpOwnerId': 'string',
                        'PublicDnsName': 'string',
                        'PublicIp': 'string'
                    },
                    'Primary': True|False,
                    'PrivateDnsName': 'string',
                    'PrivateIpAddress': 'string'
                },
            ],
            'SourceDestCheck': 'string',

```

(continues on next page)

(continued from previous page)

```

        'Status': 'string',
        'SubnetId': 'string',
        'VpcId': 'string'
    },
    ],
    'NiftyPrivateIpType': 'string',
    'Placement': {
        'AvailabilityZone': 'string'
    },
    'Platform': 'string',
    'PrivateDnsName': 'string',
    'PrivateIpAddress': 'string',
    'PrivateIpAddressV6': 'string',
    'Reason': 'string',
    'RootDeviceType': 'string'
},
],
'OwnerId': 'string',
'RequestId': 'string',
'ReservationId': 'string'
}

```

### Response Structure

- (dict) –
  - **GroupSet** (list) –
    - \* (dict) –
      - **GroupId** (string) –
  - **InstancesSet** (list) –
    - \* (dict) –
      - **AccountingType** (string) –
      - **Admin** (string) –
      - **Architecture** (string) –
      - **BlockDeviceMapping** (list) –
      - (dict) –
      - **DeviceName** (string) –
      - **Ebs** (dict) –
      - **DeleteOnTermination** (string) –
      - **Status** (string) –
      - **VolumeId** (string) –
      - **VolumeUniqueId** (string) –
      - **Description** (string) –
      - **DnsName** (string) –
      - **ImageId** (string) –
      - **InstanceId** (string) –
      - **InstanceState** (dict) –
      - **Code** (integer) –
      - **Name** (string) –
      - **InstanceType** (string) –
      - **InstanceUniqueId** (string) –
      - **IpAddress** (string) –
      - **IpAddressV6** (string) –
      - **IpType** (string) –
      - **IsoImage** (list) –
      - (dict) –

- **IsoImageId** (*string*) –
- **IsoImageName** (*string*) –
- **KeyName** (*string*) –
- **LaunchTime** (*datetime*) –
- **Monitoring** (*dict*) –
- **State** (*string*) –
- **NetworkInterfaceSet** (*list*) –
- (*dict*) –
- **Association** (*dict*) –
- **IpOwnerId** (*string*) –
- **PublicDnsName** (*string*) –
- **PublicIp** (*string*) –
- **Attachment** (*dict*) –
- **AttachTime** (*string*) –
- **AttachmentID** (*string*) –
- **DeleteOnTermination** (*string*) –
- **DeviceIndex** (*string*) –
- **Status** (*string*) –
- **Description** (*string*) –
- **GroupSet** (*list*) –
- (*dict*) –
- **GroupId** (*string*) –
- **NetworkInterfaceId** (*string*) –
- **NiftyNetworkId** (*string*) –
- **NiftyNetworkName** (*string*) –
- **OwnerId** (*string*) –
- **PrivateDnsName** (*string*) –
- **PrivateIpAddressesSet** (*list*) –
- (*dict*) –
- **Association** (*dict*) –
- **IpOwnerId** (*string*) –
- **PublicDnsName** (*string*) –
- **PublicIp** (*string*) –
- **Primary** (*boolean*) –
- **PrivateDnsName** (*string*) –
- **PrivateIpAddress** (*string*) –
- **SourceDestCheck** (*string*) –
- **Status** (*string*) –
- **SubnetId** (*string*) –
- **VpcId** (*string*) –
- **NiftyPrivateIpType** (*string*) –
- **Placement** (*dict*) –
- **AvailabilityZone** (*string*) –
- **Platform** (*string*) –
- **PrivateDnsName** (*string*) –
- **PrivateIpAddress** (*string*) –
- **PrivateIpAddressV6** (*string*) –
- **Reason** (*string*) –
- **RootDeviceType** (*string*) –
- **OwnerId** (*string*) –
- **RequestId** (*string*) –
- **ReservationId** (*string*) –

*computing* / Client / set\_filter\_for\_load\_balancer

## set\_filter\_for\_load\_balancer

computing.Client.set\_filter\_for\_load\_balancer(\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```

response = client.set_filter_for_load_balancer(
    FilterType='1'|'2',
    IPAddresses=[
        {
            'AddOnFilter': True|False,
            'IPAddress': 'string'
        },
    ],
    InstancePort=123,
    LoadBalancerName='string',
    LoadBalancerPort=123
)

```

### Parameters

- **FilterType** (*string*) –
- **IPAddresses** (*list*) –
  - (*dict*) –
    - \* **AddOnFilter** (*boolean*) –
    - \* **IPAddress** (*string*) –
- **InstancePort** (*integer*) – [REQUIRED]
- **LoadBalancerName** (*string*) – [REQUIRED]
- **LoadBalancerPort** (*integer*) – [REQUIRED]

Return type dict

### Returns

### Response Syntax

```

{
    'ResponseMetadata': {
        'RequestId': 'string'
    },
    'SetFilterForLoadBalancerResult': {
        'Filter': {
            'FilterType': 'string',
            'IPAddresses': [
                {
                    'IPAddress': 'string'
                },
            ]
        }
    }
}

```

### Response Structure

- (*dict*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –
  - **SetFilterForLoadBalancerResult** (*dict*) –
    - \* **Filter** (*dict*) –
      - **FilterType** (*string*) –
      - **IPAddresses** (*list*) –

- *(dict)* –
- **IPAddress** (*string*) –

*computing* / Client / `set_load_balancer_listener_ssl_certificate`

## `set_load_balancer_listener_ssl_certificate`

`computing.Client.set_load_balancer_listener_ssl_certificate(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.set_load_balancer_listener_ssl_certificate(  
    InstancePort=123,  
    LoadBalancerName='string',  
    LoadBalancerPort=123,  
    SSLCertificateId='string'  
)
```

### Parameters

- **InstancePort** (*integer*) – [REQUIRED]
- **LoadBalancerName** (*string*) – [REQUIRED]
- **LoadBalancerPort** (*integer*) – [REQUIRED]
- **SSLCertificateId** (*string*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```
{  
    'ResponseMetadata': {  
        'RequestId': 'string'  
    },  
    'SetLoadBalancerListenerSSLCertificateResult': 'string'  
}
```

### Response Structure

- *(dict)* –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –
  - **SetLoadBalancerListenerSSLCertificateResult** (*string*) –

*computing* / Client / `set_remote_access_vpn_gateway_ca_certificate`

## `set_remote_access_vpn_gateway_ca_certificate`

`computing.Client.set_remote_access_vpn_gateway_ca_certificate(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.set_remote_access_vpn_gateway_ca_certificate(  
    CACertificateId='string',  
    RemoteAccessVpnGatewayId='string'  
)
```

### Parameters



- **CACertificateId** (*string*) – [REQUIRED]
- **RemoteAccessVpnGatewayId** (*string*) – [REQUIRED]

Return type dict

Returns

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / `set_remote_access_vpn_gateway_ssl_certificate`

### set\_remote\_access\_vpn\_gateway\_ssl\_certificate

`computing.Client.set_remote_access_vpn_gateway_ssl_certificate(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.set_remote_access_vpn_gateway_ssl_certificate(
    RemoteAccessVpnGatewayId='string',
    SSLCertificateId='string'
)
```

#### Parameters

- **RemoteAccessVpnGatewayId** (*string*) – [REQUIRED]
- **SSLCertificateId** (*string*) – [REQUIRED]

Return type dict

Returns

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / `start_instances`

### start\_instances

`computing.Client.start_instances(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.start_instances(  
    AccountingType=[  
        'string',  
    ],  
    InstanceId=[  
        'string',  
    ],  
    InstanceType=[  
        'string',  
    ],  
    NiftyIsBios=True|False,  
    Tenancy=[  
        'string',  
    ],  
    UserData={  
        'Content': 'string',  
        'Encoding': 'string'  
    }  
)
```

#### Parameters

- **AccountingType** (*list*) –
  - (*string*) –
- **InstanceId** (*list*) – [REQUIRED]
  - (*string*) –
- **InstanceType** (*list*) –
  - (*string*) –
- **NiftyIsBios** (*boolean*) –
- **Tenancy** (*list*) –
  - (*string*) –
- **UserData** (*dict*) –
  - **Content** (*string*) –
  - **Encoding** (*string*) –

**Return type** dict

#### Returns

##### Response Syntax

```
{  
    'InstancesSet': [  
        {  
            'CurrentState': {  
                'Code': 123,  
                'Name': 'string'  
            },  
            'InstanceId': 'string',  
            'InstanceUniqueId': 'string',  
            'PreviousState': {  
                'Code': 123,  
                'Name': 'string'  
            },  
            'Tenancy': 'string'  
        },  
    ],  
    'RequestId': 'string'  
}
```

**Response Structure**

- *(dict)* –
  - **InstancesSet** (*list*) –
    - \* *(dict)* –
      - **CurrentState** (*dict*) –
      - **Code** (*integer*) –
      - **Name** (*string*) –
      - **InstanceId** (*string*) –
      - **InstanceUniqueId** (*string*) –
      - **PreviousState** (*dict*) –
      - **Code** (*integer*) –
      - **Name** (*string*) –
      - **Tenancy** (*string*) –
  - **RequestId** (*string*) –

*computing* / Client / stop\_instances

**stop\_instances**

`computing.Client.stop_instances` (*\*\*kwargs*)

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.stop_instances(
    Force=True|False,
    InstanceId=[
        'string',
    ],
    Tenancy=[
        'string',
    ]
)
```

**Parameters**

- **Force** (*boolean*) –
- **InstanceId** (*list*) – **[REQUIRED]**
  - (*string*) –
- **Tenancy** (*list*) –
  - (*string*) –

**Return type** dict

**Returns****Response Syntax**

```
{
  'InstancesSet': [
    {
      'CurrentState': {
        'Code': 123,
        'Name': 'string'
      },
      'InstanceId': 'string',
      'InstanceUniqueId': 'string',
      'PreviousState': {
        'Code': 123,
```

(continues on next page)

(continued from previous page)

```

        'Name': 'string'
    },
    'Tenancy': 'string'
},
],
'RequestId': 'string'
}

```

**Response Structure**

- (dict) –
  - **InstancesSet** (list) –
    - \* (dict) –
      - **CurrentState** (dict) –
      - **Code** (integer) –
      - **Name** (string) –
      - **InstanceId** (string) –
      - **InstanceUniqueId** (string) –
      - **PreviousState** (dict) –
      - **Code** (integer) –
      - **Name** (string) –
      - **Tenancy** (string) –
  - **RequestId** (string) –

*computing* / Client / `terminate_instances`**terminate\_instances**`computing.Client.terminate_instances(**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```

response = client.terminate_instances(
    InstanceId=[
        'string',
    ]
)

```

**Parameters** **InstanceId** (list) – [REQUIRED]

- (string) –

**Return type** dict**Returns****Response Syntax**

```

{
    'InstancesSet': [
        {
            'CurrentState': {
                'Code': 123,
                'Name': 'string'
            },
            'InstanceId': 'string',
            'InstanceUniqueId': 'string',
            'PreviousState': {

```

(continues on next page)

(continued from previous page)

```

        'Code': 123,
        'Name': 'string'
    },
    ],
    'RequestId': 'string'
}

```

**Response Structure**

- (dict) –
  - **InstancesSet** (list) –
    - \* (dict) –
      - **CurrentState** (dict) –
      - **Code** (integer) –
      - **Name** (string) –
      - **InstanceId** (string) –
      - **InstanceUniqueId** (string) –
      - **PreviousState** (dict) –
      - **Code** (integer) –
      - **Name** (string) –
  - **RequestId** (string) –

*computing* / Client / unset\_load\_balancer\_listener\_ssl\_certificate

**unset\_load\_balancer\_listener\_ssl\_certificate**

`computing.Client.unset_load_balancer_listener_ssl_certificate(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```

response = client.unset_load_balancer_listener_ssl_certificate(
    InstancePort=123,
    LoadBalancerName='string',
    LoadBalancerPort=123
)

```

**Parameters**

- **InstancePort** (integer) – [REQUIRED]
- **LoadBalancerName** (string) – [REQUIRED]
- **LoadBalancerPort** (integer) – [REQUIRED]

**Return type** dict

**Returns****Response Syntax**

```

{
    'ResponseMetadata': {
        'RequestId': 'string'
    },
    'UnsetLoadBalancerListenerSSLCertificateResult': 'string'
}

```

**Response Structure**

- (dict) –

- **ResponseMetadata** (*dict*) –
  - \* **RequestId** (*string*) –
- **UnsetLoadBalancerListenerSSLCertificateResult** (*string*) –

*computing* / Client / unset\_remote\_access\_vpn\_gateway\_ca\_certificate

## unset\_remote\_access\_vpn\_gateway\_ca\_certificate

`computing.Client.unset_remote_access_vpn_gateway_ca_certificate(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.unset_remote_access_vpn_gateway_ca_certificate(  
    RemoteAccessVpnGatewayId='string'  
)
```

**Parameters** `RemoteAccessVpnGatewayId` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'RequestId': 'string',  
    'Return': True|False  
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / unset\_remote\_access\_vpn\_gateway\_ssl\_certificate

## unset\_remote\_access\_vpn\_gateway\_ssl\_certificate

`computing.Client.unset_remote_access_vpn_gateway_ssl_certificate(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.unset_remote_access_vpn_gateway_ssl_certificate(  
    RemoteAccessVpnGatewayId='string'  
)
```

**Parameters** `RemoteAccessVpnGatewayId` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'RequestId': 'string',  
    'Return': True|False  
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / update\_load\_balancer

## update\_load\_balancer

`computing.Client.update_load_balancer(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.update_load_balancer(
    AccountingTypeUpdate=123,
    ListenerUpdate={
        'InstancePort': 123,
        'LoadBalancerPort': 123,
        'RequestListener': {
            'BalancingType': 123,
            'InstancePort': 123,
            'LoadBalancerPort': 123,
            'Protocol': 'HTTP' | 'HTTPS' | 'FTP' | 'custom'
        }
    },
    LoadBalancerName='string',
    LoadBalancerNameUpdate='string',
    NetworkVolumeUpdate=123
)
```

### Parameters

- **AccountingTypeUpdate** (*integer*) –
- **ListenerUpdate** (*dict*) –
  - **InstancePort** (*integer*) –
  - **LoadBalancerPort** (*integer*) –
  - **RequestListener** (*dict*) –
    - \* **BalancingType** (*integer*) –
    - \* **InstancePort** (*integer*) –
    - \* **LoadBalancerPort** (*integer*) –
    - \* **Protocol** (*string*) –
- **LoadBalancerName** (*string*) – [REQUIRED]
- **LoadBalancerNameUpdate** (*string*) –
- **NetworkVolumeUpdate** (*integer*) –

**Return type** `dict`

### Returns

### Response Syntax

```
{
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

### Response Structure

- (*dict*) –
  - **ResponseMetadata** (*dict*) –

\* **RequestId** (*string*) –

*computing* / Client / update\_load\_balancer\_option

## update\_load\_balancer\_option

`computing.Client.update_load_balancer_option(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.update_load_balancer_option(
    InstancePort=123,
    LoadBalancerName='string',
    LoadBalancerPort=123,
    SessionStickinessPolicyUpdate={
        'Enable': True|False,
        'ExpirationPeriod': 123
    },
    SorryPageUpdate={
        'Enable': True|False,
        'StatusCode': 123
    }
)
```

### Parameters

- **InstancePort** (*integer*) – [REQUIRED]
- **LoadBalancerName** (*string*) – [REQUIRED]
- **LoadBalancerPort** (*integer*) – [REQUIRED]
- **SessionStickinessPolicyUpdate** (*dict*) –
  - **Enable** (*boolean*) –
  - **ExpirationPeriod** (*integer*) –
- **SorryPageUpdate** (*dict*) –
  - **Enable** (*boolean*) –
  - **StatusCode** (*integer*) –

**Return type** dict

### Returns

#### Response Syntax

```
{
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

### Response Structure

- (*dict*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*computing* / Client / update\_security\_group

## update\_security\_group

`computing.Client.update_security_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)



### Request Syntax

```
response = client.update_security_group(
    GroupDescriptionUpdate='string',
    GroupLogFilterBroadcast=True|False,
    GroupLogFilterNetBios=True|False,
    GroupLogLimitUpdate=123,
    GroupName='string',
    GroupNameUpdate='string',
    GroupRuleLimitUpdate=123
)
```

#### Parameters

- **GroupDescriptionUpdate** (*string*) –
- **GroupLogFilterBroadcast** (*boolean*) –
- **GroupLogFilterNetBios** (*boolean*) –
- **GroupLogLimitUpdate** (*integer*) –
- **GroupName** (*string*) – [REQUIRED]
- **GroupNameUpdate** (*string*) –
- **GroupRuleLimitUpdate** (*integer*) –

Return type dict

#### Returns

#### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

#### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / upload\_iso\_image

### upload\_iso\_image

`computing.Client.upload_iso_image(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.upload_iso_image(
    AvailabilityZone='string',
    Description='string',
    IsoImageName='string',
    IsoUrl='string'
)
```

#### Parameters

- **AvailabilityZone** (*string*) –
- **Description** (*string*) –
- **IsoImageName** (*string*) –
- **IsoUrl** (*string*) – [REQUIRED]

Return type dict

## Returns

### Response Syntax

```
{
    'RequestId': 'string',
    'Return': True|False
}
```

### Response Structure

- (*dict*) –
  - **RequestId** (*string*) –
  - **Return** (*boolean*) –

*computing* / Client / upload\_ssl\_certificate

## upload\_ssl\_certificate

`computing.Client.upload_ssl_certificate(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.upload_ssl_certificate(
    Certificate='string',
    CertificateAuthority='string',
    Key='string'
)
```

### Parameters

- **Certificate** (*string*) – [REQUIRED]
- **CertificateAuthority** (*string*) –
- **Key** (*string*) – [REQUIRED]

**Return type** dict

### Returns

#### Response Syntax

```
{
    'Fqdn': 'string',
    'FqdnId': 'string',
    'KeyFingerprint': 'string',
    'RequestId': 'string'
}
```

### Response Structure

- (*dict*) –
  - **Fqdn** (*string*) –
  - **FqdnId** (*string*) –
  - **KeyFingerprint** (*string*) –
  - **RequestId** (*string*) –

## 1.1.2 Client Exceptions

Client exceptions are available on a client instance via the `exceptions` property. For more detailed instructions and examples on the exact usage of client exceptions, see the [error handling user guide](#).

This client has no modeled exception classes.

### 1.1.3 Waiters

Waiters are available on a client instance via the `get_waiter` method. For more detailed instructions and examples on the usage or waiters, see the waiters [user guide](#).

The available waiters are:

*computing* / Waiter / CustomerGatewayAvailable

#### CustomerGatewayAvailable

**class** `computing.Waiter.CustomerGatewayAvailable`

```
waiter = client.get_waiter('customer_gateway_available')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_customer_gateways()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

#### Request Syntax

```
waiter.wait(
    CustomerGatewayId=[
        'string',
    ],
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'customer-gateway-id'|'nifty-customer-gateway-name'|'ip-
→address'|'state'|'nifty-customer-gateway-description'
        },
    ],
    NiftyCustomerGatewayName=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

#### Parameters

- **CustomerGatewayId** (*list*) –
  - (*string*) –
- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **NiftyCustomerGatewayName** (*list*) –

- (string) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / CustomerGatewayDeleted

## CustomerGatewayDeleted

**class** `computing.Waiter.CustomerGatewayDeleted`

```
waiter = client.get_waiter('customer_gateway_deleted')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_customer_gateways()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    CustomerGatewayId=[
        'string',
    ],
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'customer-gateway-id'|'nifty-customer-gateway-name'|'ip-
→address'|'state'|'nifty-customer-gateway-description'
        },
    ],
    NiftyCustomerGatewayName=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **CustomerGatewayId** (*list*) –
  - (string) –
- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (string) –
    - \* **Name** (*string*) –

- **NiftyCustomerGatewayName** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
    - The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
    - The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / CustomerGatewayExists

## CustomerGatewayExists

**class** `computing.Waiter.CustomerGatewayExists`

```
waiter = client.get_waiter('customer_gateway_exists')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_customer_gateways()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    CustomerGatewayId=[
        'string',
    ],
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'customer-gateway-id'|'nifty-customer-gateway-name'|'ip-
→address'|'state'|'nifty-customer-gateway-description'
        },
    ],
    NiftyCustomerGatewayName=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **CustomerGatewayId** (*list*) –
  - (*string*) –
- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –

- \* **Name** (*string*) –
- **NiftyCustomerGatewayName** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / CustomerGatewayStopped

### CustomerGatewayStopped

**class** `computing.Waiter.CustomerGatewayStopped`

```
waiter = client.get_waiter('customer_gateway_stopped')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_customer_gateways()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

#### Request Syntax

```
waiter.wait(
    CustomerGatewayId=[
        'string',
    ],
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'customer-gateway-id'|'nifty-customer-gateway-name'|'ip-
→address'|'state'|'nifty-customer-gateway-description'
        },
    ],
    NiftyCustomerGatewayName=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

#### Parameters

- **CustomerGatewayId** (*list*) –
  - (*string*) –
- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –

- (string) –
- \* **Name** (string) –
- **NiftyCustomerGatewayName** (list) –
  - (string) –
- **WaiterConfig** (dict) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (integer) –
    - The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (integer) –
    - The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / CustomerGatewayWarning

## CustomerGatewayWarning

**class** `computing.Waiter.CustomerGatewayWarning`

```
waiter = client.get_waiter('customer_gateway_warning')
```

**wait** (\*\*kwargs)

Polls `computing.Client.describe_customer_gateways()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    CustomerGatewayId=[
        'string',
    ],
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'customer-gateway-id'|'nifty-customer-gateway-name'|'ip-
→address'|'state'|'nifty-customer-gateway-description'
        },
    ],
    NiftyCustomerGatewayName=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **CustomerGatewayId** (list) –
  - (string) –
- **Filter** (list) –
  - (dict) –

- \* **ListOfRequestValue** (*list*) –
  - (*string*) –
- \* **Name** (*string*) –
- **NiftyCustomerGatewayName** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / ElasticLoadBalancerAvailable

## ElasticLoadBalancerAvailable

**class** `computing.Waiter.ElasticLoadBalancerAvailable`

```
waiter = client.get_waiter('elastic_load_balancer_available')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.nifty_describe_elastic_load_balancers()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    ElasticLoadBalancers={
        'ListOfRequestElasticLoadBalancerId': [
            'string',
        ],
        'ListOfRequestElasticLoadBalancerName': [
            'string',
        ],
        'ListOfRequestElasticLoadBalancerPort': [
            123,
        ],
        'ListOfRequestInstancePort': [
            123,
        ],
        'ListOfRequestProtocol': [
            'string',
        ],
    },
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'availability-zone'|'state'|'elastic-loadbalancer-id'|
→ 'elastic-loadbalancer-name'|'description'|'accounting-type'|'ip-address'|
→ 'version'
```

(continues on next page)



(continued from previous page)

```

    },
],
WaiterConfig={
    'Delay': 123,
    'MaxAttempts': 123
}
)

```

**Parameters**

- **ElasticLoadBalancers** (*dict*) –
  - **ListOfRequestElasticLoadBalancerId** (*list*) –
    - \* (*string*) –
  - **ListOfRequestElasticLoadBalancerName** (*list*) –
    - \* (*string*) –
  - **ListOfRequestElasticLoadBalancerPort** (*list*) –
    - \* (*integer*) –
  - **ListOfRequestInstancePort** (*list*) –
    - \* (*integer*) –
  - **ListOfRequestProtocol** (*list*) –
    - \* (*string*) –
- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None*computing* / Waiter / ElasticLoadBalancerDeleted**ElasticLoadBalancerDeleted****class** *computing*.Waiter.**ElasticLoadBalancerDeleted**

```
waiter = client.get_waiter('elastic_load_balancer_deleted')
```

**wait** (*\*\*kwargs*)

Polls *computing.Client.nifty\_describe\_elastic\_load\_balancers()* every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```
waiter.wait(
    ElasticLoadBalancers={
        'ListOfRequestElasticLoadBalancerId': [

```

(continues on next page)

(continued from previous page)

```

        'string',
    ],
    'ListOfRequestElasticLoadBalancerName': [
        'string',
    ],
    'ListOfRequestElasticLoadBalancerPort': [
        123,
    ],
    'ListOfRequestInstancePort': [
        123,
    ],
    'ListOfRequestProtocol': [
        'string',
    ]
},
Filter=[
    {
        'ListOfRequestValue': [
            'string',
        ],
        'Name': 'availability-zone'|'state'|'elastic-loadbalancer-id'|
→'elastic-loadbalancer-name'|'description'|'accounting-type'|'ip-address'|
→'version'
    },
],
WaiterConfig={
    'Delay': 123,
    'MaxAttempts': 123
}
)

```

### Parameters

- **ElasticLoadBalancers** (*dict*) –
  - **ListOfRequestElasticLoadBalancerId** (*list*) –
    - \* (*string*) –
  - **ListOfRequestElasticLoadBalancerName** (*list*) –
    - \* (*string*) –
  - **ListOfRequestElasticLoadBalancerPort** (*list*) –
    - \* (*integer*) –
  - **ListOfRequestInstancePort** (*list*) –
    - \* (*integer*) –
  - **ListOfRequestProtocol** (*list*) –
    - \* (*string*) –
- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
    - The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
    - The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / ElasticLoadBalancerExists

## ElasticLoadBalancerExists

**class** *computing*.Waiter.**ElasticLoadBalancerExists**

```
waiter = client.get_waiter('elastic_load_balancer_exists')
```

**wait** (*\*\*kwargs*)

Polls *computing.Client.nifty\_describe\_elastic\_load\_balancers()* every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    ElasticLoadBalancers={
        'ListOfRequestElasticLoadBalancerId': [
            'string',
        ],
        'ListOfRequestElasticLoadBalancerName': [
            'string',
        ],
        'ListOfRequestElasticLoadBalancerPort': [
            123,
        ],
        'ListOfRequestInstancePort': [
            123,
        ],
        'ListOfRequestProtocol': [
            'string',
        ],
    },
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'availability-zone'|'state'|'elastic-loadbalancer-id'|
↪ 'elastic-loadbalancer-name'|'description'|'accounting-type'|'ip-address'|
↪ 'version'
        },
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **ElasticLoadBalancers** (*dict*) –
  - **ListOfRequestElasticLoadBalancerId** (*list*) –
    - \* (*string*) –
  - **ListOfRequestElasticLoadBalancerName** (*list*) –

- \* (*string*) –
  - **ListOfRequestElasticLoadBalancerPort** (*list*) –
  - \* (*integer*) –
  - **ListOfRequestInstancePort** (*list*) –
  - \* (*integer*) –
  - **ListOfRequestProtocol** (*list*) –
  - \* (*string*) –
- **Filter** (*list*) –
- (*dict*) –
  - \* **ListOfRequestValue** (*list*) –
    - (*string*) –
    - \* **Name** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
- **Delay** (*integer*) –
  - The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
  - The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / InstanceDeleted

## InstanceDeleted

**class** `computing.Waiter.InstanceDeleted`

```
waiter = client.get_waiter('instance_deleted')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_instances()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(  
    InstanceId=[  
        'string',  
    ],  
    Tenancy=[  
        'string',  
    ],  
    WaiterConfig={  
        'Delay': 123,  
        'MaxAttempts': 123  
    }  
)
```

### Parameters

- **InstanceId** (*list*) –
- (*string*) –
- **Tenancy** (*list*) –
- (*string*) –

- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / InstanceExists

## InstanceExists

**class** `computing.Waiter.InstanceExists`

```
waiter = client.get_waiter('instance_exists')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_instances()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    InstanceId=[
        'string',
    ],
    Tenancy=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **InstanceId** (*list*) –
  - (*string*) –
- **Tenancy** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / InstanceImportError

## InstanceImportError

**class** `computing.Waiter.InstanceImportError`

```
waiter = client.get_waiter('instance_import_error')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_instances()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    InstanceId=[
        'string',
    ],
    Tenancy=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **InstanceId** (*list*) –
  - (*string*) –
- **Tenancy** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None

`computing` / Waiter / InstanceRunning

## InstanceRunning

**class** `computing.Waiter.InstanceRunning`

```
waiter = client.get_waiter('instance_running')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_instances()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    InstanceId=[
        'string',
    ],
    Tenancy=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

#### Parameters

- **InstanceId** (*list*) –
  - (*string*) –
- **Tenancy** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

#### Returns

None

*computing* / Waiter / InstanceStopped

### InstanceStopped

**class** `computing.Waiter.InstanceStopped`

```
waiter = client.get_waiter('instance_stopped')
```

#### **wait** (*\*\*kwargs*)

Polls `computing.Client.describe_instances()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

#### Request Syntax

```
waiter.wait(
    InstanceId=[
        'string',
    ],
    Tenancy=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

**Parameters**

- **InstanceId** (*list*) –
  - (*string*) –
- **Tenancy** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None*computing* / Waiter / InstanceSuspending**InstanceSuspending****class** `computing.Waiter.InstanceSuspending`

```
waiter = client.get_waiter('instance_suspending')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_instances()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```
waiter.wait(  
    InstanceId=[  
        'string',  
    ],  
    Tenancy=[  
        'string',  
    ],  
    WaiterConfig={  
        'Delay': 123,  
        'MaxAttempts': 123  
    }  
)
```

**Parameters**

- **InstanceId** (*list*) –
  - (*string*) –
- **Tenancy** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40



**Returns** None

*computing* / Waiter / InstanceWarning

## InstanceWarning

**class** `computing.Waiter.InstanceWarning`

```
waiter = client.get_waiter('instance_warning')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_instances()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    InstanceId=[
        'string',
    ],
    Tenancy=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **InstanceId** (*list*) –
  - (*string*) –
- **Tenancy** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / LoadBalancerDeleted

## LoadBalancerDeleted

**class** `computing.Waiter.LoadBalancerDeleted`

```
waiter = client.get_waiter('load_balancer_deleted')
```

**wait** (\*\*kwargs)

Polls `computing.Client.describe_load_balancers()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

#### Request Syntax

```
waiter.wait(  
    LoadBalancerNames=[  
        {  
            'InstancePort': 123,  
            'LoadBalancerName': 'string',  
            'LoadBalancerPort': 123  
        },  
    ],  
    Owner='self'|'other'|'all',  
    WaiterConfig={  
        'Delay': 123,  
        'MaxAttempts': 123  
    }  
)
```

#### Parameters

- **LoadBalancerNames** (*list*) –
  - (*dict*) –
    - \* **InstancePort** (*integer*) –
    - \* **LoadBalancerName** (*string*) –
    - \* **LoadBalancerPort** (*integer*) –
- **Owner** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –

The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / LoadBalancerExists

### LoadBalancerExists

**class** `computing.Waiter.LoadBalancerExists`

```
waiter = client.get_waiter('load_balancer_exists')
```

**wait** (\*\*kwargs)

Polls `computing.Client.describe_load_balancers()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

#### Request Syntax

```

waiter.wait(
    LoadBalancerNames=[
        {
            'InstancePort': 123,
            'LoadBalancerName': 'string',
            'LoadBalancerPort': 123
        },
    ],
    Owner='self'|'other'|'all',
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)

```

### Parameters

- **LoadBalancerNames** (*list*) –
  - (*dict*) –
    - \* **InstancePort** (*integer*) –
    - \* **LoadBalancerName** (*string*) –
    - \* **LoadBalancerPort** (*integer*) –
- **Owner** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / LoadBalancerInService

## LoadBalancerInService

**class** `computing.Waiter.LoadBalancerInService`

```
waiter = client.get_waiter('load_balancer_in_service')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_load_balancers()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```

waiter.wait(
    LoadBalancerNames=[
        {
            'InstancePort': 123,
            'LoadBalancerName': 'string',
            'LoadBalancerPort': 123
        },
    ],

```

(continues on next page)

(continued from previous page)

```

Owner='self' | 'other' | 'all',
WaiterConfig={
    'Delay': 123,
    'MaxAttempts': 123
}
)

```

**Parameters**

- **LoadBalancerNames** (*list*) –
  - (*dict*) –
    - \* **InstancePort** (*integer*) –
    - \* **LoadBalancerName** (*string*) –
    - \* **LoadBalancerPort** (*integer*) –
- **Owner** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None*computing* / Waiter / PrivateLanAvailable**PrivateLanAvailable****class** *computing.Waiter.PrivateLanAvailable*

```
waiter = client.get_waiter('private_lan_available')
```

**wait** (*\*\*kwargs*)

Polls *computing.Client.nifty\_describe\_private\_lans()* every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```

waiter.wait(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'availabilityZone, availability-zone' | 'cidrBlock, cidr,
↪ cidr-block' | 'state' | 'network-id' | 'private-lan-name' | 'accountingType' |
↪ 'description'
        },
    ],
    NetworkId=[
        'string',
    ],
    PrivateLanName=[

```

(continues on next page)

(continued from previous page)

```

        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)

```

**Parameters**

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **NetworkId** (*list*) –
  - (*string*) –
- **PrivateLanName** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None*computing* / Waiter / PrivateLanDeleted**PrivateLanDeleted****class** *computing*.Waiter.**PrivateLanDeleted**

```
waiter = client.get_waiter('private_lan_deleted')
```

**wait** (*\*\*kwargs*)

Polls *computing.Client.nifty\_describe\_private\_lans()* every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```

waiter.wait(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'availabilityZone, availability-zone'|'cidrBlock, cidr,
↪cidr-block'|'state'|'network-id'|'private-lan-name'|'accountingType'|
↪'description'
        },
    ],
)

```

(continues on next page)

(continued from previous page)

```

NetworkId=[
    'string',
],
PrivateLanName=[
    'string',
],
WaiterConfig={
    'Delay': 123,
    'MaxAttempts': 123
}
)

```

**Parameters**

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **NetworkId** (*list*) –
  - (*string*) –
- **PrivateLanName** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None*computing* / Waiter / PrivateLanExists**PrivateLanExists****class** `computing.Waiter.PrivateLanExists`

```
waiter = client.get_waiter('private_lan_exists')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.nifty_describe_private_lans()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```

waiter.wait(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'availabilityZone, availability-zone'|'cidrBlock, cidr,
↪cidr-block'|'state'|'network-id'|'private-lan-name'|'account'|'continues on next page)
↪'description'

```

(continued from previous page)

```

    },
],
NetworkId=[
    'string',
],
PrivateLanName=[
    'string',
],
WaiterConfig={
    'Delay': 123,
    'MaxAttempts': 123
}
)

```

**Parameters**

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **NetworkId** (*list*) –
  - (*string*) –
- **PrivateLanName** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
    - The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
    - The maximum number of attempts to be made. Default: 40

**Returns** None*computing* / Waiter / RemoteAccessVpnGatewayAvailable**RemoteAccessVpnGatewayAvailable****class** `computing.Waiter.RemoteAccessVpnGatewayAvailable`

```
waiter = client.get_waiter('remote_access_vpn_gateway_available')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_remote_access_vpn_gateways()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```

waiter.wait(
    RemoteAccessVpnGatewayId=[
        'string',
    ],
    WaiterConfig={

```

(continues on next page)

(continued from previous page)

```
        'Delay': 123,  
        'MaxAttempts': 123  
    }  
)
```

**Parameters**

- **RemoteAccessVpnGatewayId** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None*computing* / Waiter / RemoteAccessVpnGatewayDeleted**RemoteAccessVpnGatewayDeleted****class** *computing*.Waiter.RemoteAccessVpnGatewayDeleted

```
waiter = client.get_waiter('remote_access_vpn_gateway_deleted')
```

**wait** (*\*\*kwargs*)

Polls *computing.Client.describe\_remote\_access\_vpn\_gateways()* every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```
waiter.wait(  
    RemoteAccessVpnGatewayId=[  
        'string',  
    ],  
    WaiterConfig={  
        'Delay': 123,  
        'MaxAttempts': 123  
    }  
)
```

**Parameters**

- **RemoteAccessVpnGatewayId** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None



*computing* / Waiter / RemoteAccessVpnGatewayExists

## RemoteAccessVpnGatewayExists

**class** `computing.Waiter.RemoteAccessVpnGatewayExists`

```
waiter = client.get_waiter('remote_access_vpn_gateway_exists')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_remote_access_vpn_gateways()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    RemoteAccessVpnGatewayId=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **RemoteAccessVpnGatewayId** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / RouterAvailable

## RouterAvailable

**class** `computing.Waiter.RouterAvailable`

```
waiter = client.get_waiter('router_available')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.nifty_describe_routers()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'availability-zone'|'state'|'router-id'|'router-name'|
→ 'description'|'accountingType'|'type'|'ip-address'|'version'|'latest-
→ version-information'
        },
    ],
    RouterId=[
        'string',
    ],
    RouterName=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

#### Parameters

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **RouterId** (*list*) –
  - (*string*) –
- **RouterName** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –

The maximum number of attempts to be made. Default: 40

#### Returns None

*computing* / Waiter / RouterDeleted

### RouterDeleted

**class** `computing.Waiter.RouterDeleted`

```
waiter = client.get_waiter('router_deleted')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.nifty_describe_routers()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

## Request Syntax

```
waiter.wait(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'availability-zone'|'state'|'router-id'|'router-name'|
→ 'description'|'accountingType'|'type'|'ip-address'|'version'|'latest-
→ version-information'
        },
    ],
    RouterId=[
        'string',
    ],
    RouterName=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **RouterId** (*list*) –
  - (*string*) –
- **RouterName** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / RouterExists

## RouterExists

**class** `computing.Waiter.RouterExists`

```
waiter = client.get_waiter('router_exists')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.nifty_describe_routers()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'availability-zone'|'state'|'router-id'|'router-name'|
→ 'description'|'accountingType'|'type'|'ip-address'|'version'|'latest-
→ version-information'
        },
    ],
    RouterId=[
        'string',
    ],
    RouterName=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

#### Parameters

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **RouterId** (*list*) –
  - (*string*) –
- **RouterName** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / RouterStopped

### RouterStopped

**class** `computing.Waiter.RouterStopped`

```
waiter = client.get_waiter('router_stopped')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.nifty_describe_routers()` every 20 seconds until a successful

state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'availability-zone'|'state'|'router-id'|'router-name'|
→ 'description'|'accountingType'|'type'|'ip-address'|'version'|'latest-
→ version-information'
        },
    ],
    RouterId=[
        'string',
    ],
    RouterName=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **RouterId** (*list*) –
  - (*string*) –
- **RouterName** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / RouterWarning

### RouterWarning

**class** `computing.Waiter.RouterWarning`

```
waiter = client.get_waiter('router_warning')
```

**wait** (\*\*kwargs)

Polls `computing.Client.nifty_describe_routers()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(  
    Filter=[  
        {  
            'ListOfRequestValue': [  
                'string',  
            ],  
            'Name': 'availability-zone'|'state'|'router-id'|'router-name'|  
→ 'description'|'accountingType'|'type'|'ip-address'|'version'|'latest-  
→ version-information'  
        },  
    ],  
    RouterId=[  
        'string',  
    ],  
    RouterName=[  
        'string',  
    ],  
    WaiterConfig={  
        'Delay': 123,  
        'MaxAttempts': 123  
    }  
)
```

### Parameters

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **RouterId** (*list*) –
  - (*string*) –
- **RouterName** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / SecurityGroupApplied

### SecurityGroupApplied

**class** `computing.Waiter.SecurityGroupApplied`

```
waiter = client.get_waiter('security_group_applied')
```

**wait** (\*\*kwargs)

Polls `computing.Client.describe_security_groups()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    Filter=[
        {
            'ListOfTypeValue': [
                'string',
            ],
            'Name': 'description'|'group-name'
        },
    ],
    GroupName=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfTypeValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **GroupName** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None

`computing` / `Waiter` / `SecurityGroupDeleted`

## SecurityGroupDeleted

**class** `computing.Waiter.SecurityGroupDeleted`

```
waiter = client.get_waiter('security_group_deleted')
```

**wait** (\*\*kwargs)

Polls `computing.Client.describe_security_groups()` every 20 seconds until a successful

state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(  
    Filter=[  
        {  
            'ListOfRequestValue': [  
                'string',  
            ],  
            'Name': 'description'|'group-name'  
        },  
    ],  
    GroupName=[  
        'string',  
    ],  
    WaiterConfig={  
        'Delay': 123,  
        'MaxAttempts': 123  
    }  
)
```

### Parameters

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **GroupName** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / SecurityGroupExists

## SecurityGroupExists

**class** `computing.Waiter.SecurityGroupExists`

```
waiter = client.get_waiter('security_group_exists')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_security_groups()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax



```

waiter.wait(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'description'|'group-name'
        },
    ],
    GroupName=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)

```

#### Parameters

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **GroupName** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
    - The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
    - The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / SnapshotDeleted

### SnapshotDeleted

**class** `computing.Waiter.SnapshotDeleted`

```
waiter = client.get_waiter('snapshot_deleted')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.nifty_describe_instance_snapshots()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

#### Request Syntax

```

waiter.wait(
    InstanceSnapshotId=[
        'string',
    ],

```

(continues on next page)

(continued from previous page)

```
SnapshotName=[
    'string',
],
WaiterConfig={
    'Delay': 123,
    'MaxAttempts': 123
}
```

**Parameters**

- **InstanceSnapshotId** (*list*) –
  - (*string*) –
- **SnapshotName** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None*computing* / Waiter / SnapshotExists**SnapshotExists****class** `computing.Waiter.SnapshotExists`

```
waiter = client.get_waiter('snapshot_exists')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.nifty_describe_instance_snapshots()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```
waiter.wait(
    InstanceSnapshotId=[
        'string',
    ],
    SnapshotName=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

**Parameters**

- **InstanceSnapshotId** (*list*) –
  - (*string*) –

- **SnapshotName** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / SnapshotNormal

## SnapshotNormal

**class** `computing.Waiter.SnapshotNormal`

```
waiter = client.get_waiter('snapshot_normal')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.nifty_describe_instance_snapshots()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    InstanceSnapshotId=[
        'string',
    ],
    SnapshotName=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **InstanceSnapshotId** (*list*) –
  - (*string*) –
- **SnapshotName** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / VolumeAttached

## VolumeAttached

**class** `computing.Waiter.VolumeAttached`

```
waiter = client.get_waiter('volume_attached')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_volumes()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    VolumeId=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **VolumeId** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / VolumeAvailable

## VolumeAvailable

**class** `computing.Waiter.VolumeAvailable`

```
waiter = client.get_waiter('volume_available')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_volumes()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    VolumeId=[
        'string',
```

(continues on next page)

(continued from previous page)

```

    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)

```

**Parameters**

- **VolumeId** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None*computing* / Waiter / VolumeDeleted**VolumeDeleted****class** *computing*.Waiter.**VolumeDeleted**

```
waiter = client.get_waiter('volume_deleted')
```

**wait** (*\*\*kwargs*)

Polls *computing.Client.describe\_volumes()* every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```

waiter.wait(
    VolumeId=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)

```

**Parameters**

- **VolumeId** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –

The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / VolumeExists

## VolumeExists

**class** `computing.Waiter.VolumeExists`

```
waiter = client.get_waiter('volume_exists')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_volumes()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    VolumeId=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **VolumeId** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –

The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / VolumeInUse

## VolumeInUse

**class** `computing.Waiter.VolumeInUse`

```
waiter = client.get_waiter('volume_in_use')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_volumes()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    VolumeId=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

#### Parameters

- **VolumeId** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
    - The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
    - The maximum number of attempts to be made. Default: 40

**Returns** None

*computing* / Waiter / VpnConnectionAvailable

### VpnConnectionAvailable

**class** `computing.Waiter.VpnConnectionAvailable`

```
waiter = client.get_waiter('vpn_connection_available')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_vpn_connections()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'customer-gateway-configuration'|'customer-gateway-id'|
↪ 'nifty-customer-gateway-name'|'state'|'option.static-routes-only'|'route.
↪ destination-cidr-block'|'type'|'vpn-connection-id'|'vpn-gateway-id'|'nifty-
↪ vpn-gateway-name'|'nifty-vpn-connection-description'|'nifty-internet-key-
↪ exchange'
        },
    ],
    VpnConnectionId=[
        'string',
    ],
)
```

(continues on next page)

(continued from previous page)

```

WaiterConfig={
    'Delay': 123,
    'MaxAttempts': 123
}
)

```

**Parameters**

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **VpnConnectionId** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None*computing* / Waiter / VpnConnectionDeleted**VpnConnectionDeleted****class** `computing.Waiter.VpnConnectionDeleted`

```
waiter = client.get_waiter('vpn_connection_deleted')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_vpn_connections()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```

waiter.wait(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'customer-gateway-configuration'|'customer-gateway-id'|
→ 'nifty-customer-gateway-name'|'state'|'option.static-routes-only'|'route.
→ destination-cidr-block'|'type'|'vpn-connection-id'|'vpn-gateway-id'|'nifty-
→ vpn-gateway-name'|'nifty-vpn-connection-description'|'nifty-internet-key-
→ exchange'
        },
    ],
    VpnConnectionId=[
        'string',
    ],
)

```

(continues on next page)



(continued from previous page)

```

    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)

```

**Parameters**

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **VpnConnectionId** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None*computing* / Waiter / VpnConnectionExists**VpnConnectionExists****class** `computing.Waiter.VpnConnectionExists`

```
waiter = client.get_waiter('vpn_connection_exists')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_vpn_connections()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```

waiter.wait(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'customer-gateway-configuration'|'customer-gateway-id'|
↪ 'nifty-customer-gateway-name'|'state'|'option.static-routes-only'|'route.
↪ destination-cidr-block'|'type'|'vpn-connection-id'|'vpn-gateway-id'|'nifty-
↪ vpn-gateway-name'|'nifty-vpn-connection-description'|'nifty-internet-key-
↪ exchange'
        },
    ],
    VpnConnectionId=[

```

(continues on next page)

(continued from previous page)

```

        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)

```

**Parameters**

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **VpnConnectionId** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None*computing* / Waiter / VpnGatewayAvailable**VpnGatewayAvailable****class** `computing.Waiter.VpnGatewayAvailable`

```
waiter = client.get_waiter('vpn_gateway_available')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_vpn_gateways()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```

waiter.wait(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'attachment.state'|'attachment.vpc-id'|'availability-zone
→'|'state'|'type'|'vpn-gateway-id'|'nifty-vpn-gateway-name'|'nifty-vpn-
→gateway-type'|'nifty-vpn-gateway-description'|'nifty-vpn-gateway-
→accountingType'|'ip-address'|'latest-version-information'|'version'
        },
    ],
    NiftyVpnGatewayName=[

```

(continues on next page)

(continued from previous page)

```

        'string',
    ],
    VpnGatewayId=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)

```

**Parameters**

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **NiftyVpnGatewayName** (*list*) –
  - (*string*) –
- **VpnGatewayId** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None*computing* / *Waiter* / *VpnGatewayDeleted***VpnGatewayDeleted****class** *computing.Waiter.VpnGatewayDeleted*

```
waiter = client.get_waiter('vpn_gateway_deleted')
```

**wait** (*\*\*kwargs*)

Polls *computing.Client.describe\_vpn\_gateways()* every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```

waiter.wait(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'attachment.state'|'attachment.vpc-id'|'availability-zone
→'|'state'|'type'|'vpn-gateway-id'|'nifty-vpn-gateway-name'|'nifty-vpn-
→gateway-type'|'nifty-vpn-gateway-description'|'nifty-vpn-gate(continues on next page)
→accountingType'|'ip-address'|'latest-version-information'|'version'

```

(continued from previous page)

```

    },
    ],
    NiftyVpnGatewayName=[
        'string',
    ],
    VpnGatewayId=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)

```

**Parameters**

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **NiftyVpnGatewayName** (*list*) –
  - (*string*) –
- **VpnGatewayId** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
    - The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
    - The maximum number of attempts to be made. Default: 40

**Returns** None*computing* / Waiter / VpnGatewayExists**VpnGatewayExists****class** `computing.Waiter.VpnGatewayExists`

```
waiter = client.get_waiter('vpn_gateway_exists')
```

**wait** (*\*\*kwargs*)

Polls `computing.Client.describe_vpn_gateways()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```

waiter.wait(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ]
        }
    ]
)

```

(continues on next page)

(continued from previous page)

```

        ],
        'Name': 'attachment.state'|'attachment.vpc-id'|'availability-zone'
→ '| 'state'|'type'|'vpn-gateway-id'|'nifty-vpn-gateway-name'|'nifty-vpn-
→ gateway-type'|'nifty-vpn-gateway-description'|'nifty-vpn-gateway-
→ accountingType'|'ip-address'|'latest-version-information'|'version'
    },
    ],
    NiftyVpnGatewayName=[
        'string',
    ],
    VpnGatewayId=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)

```

**Parameters**

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **NiftyVpnGatewayName** (*list*) –
  - (*string*) –
- **VpnGatewayId** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None*computing* / Waiter / VpnGatewayStopped**VpnGatewayStopped****class** *computing*.Waiter.VpnGatewayStopped

```
waiter = client.get_waiter('vpn_gateway_stopped')
```

**wait** (*\*\*kwargs*)

Polls *computing.Client.describe\_vpn\_gateways()* every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```
waiter.wait(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'attachment.state'|'attachment.vpc-id'|'availability-zone
→'|'state'|'type'|'vpn-gateway-id'|'nifty-vpn-gateway-name'|'nifty-vpn-
→gateway-type'|'nifty-vpn-gateway-description'|'nifty-vpn-gateway-
→accountingType'|'ip-address'|'latest-version-information'|'version'
        },
    ],
    NiftyVpnGatewayName=[
        'string',
    ],
    VpnGatewayId=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

#### Parameters

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **NiftyVpnGatewayName** (*list*) –
  - (*string*) –
- **VpnGatewayId** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –

The maximum number of attempts to be made. Default: 40

#### Returns None

*computing* / Waiter / VpnGatewayWarning

### VpnGatewayWarning

**class** *computing*.Waiter.VpnGatewayWarning

```
waiter = client.get_waiter('vpn_gateway_warning')
```

**wait** (*\*\*kwargs*)

Polls *computing.Client.describe\_vpn\_gateways()* every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
            ],
            'Name': 'attachment.state'|'attachment.vpc-id'|'availability-zone
→'|'state'|'type'|'vpn-gateway-id'|'nifty-vpn-gateway-name'|'nifty-vpn-
→gateway-type'|'nifty-vpn-gateway-description'|'nifty-vpn-gateway-
→accountingType'|'ip-address'|'latest-version-information'|'version'
        },
    ],
    NiftyVpnGatewayName=[
        'string',
    ],
    VpnGatewayId=[
        'string',
    ],
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **Filter** (*list*) –
  - (*dict*) –
    - \* **ListOfRequestValue** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
- **NiftyVpnGatewayName** (*list*) –
  - (*string*) –
- **VpnGatewayId** (*list*) –
  - (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
    - The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
    - The maximum number of attempts to be made. Default: 40

**Returns** None

## 1.2 dns

### 1.2.1 Client

**class** `dns.Client`

A low-level client representing NIFCLOUD DNS

```
client = session.create_client('dns')
```

These are the available methods:

*dns* / Client / can\_paginate

### can\_paginate

`dns.Client.can_paginate(operation_name)`

Check if an operation can be paginated.

**Parameters** `operation_name` (*string*) – The operation name. This is the same name as the method name on the client. For example, if the method name is `create_foo`, and you'd normally invoke the operation as `client.create_foo(**kwargs)`, if the `create_foo` operation can be paginated, you can use the call `client.get_paginator("create_foo")`.

**Returns** True if the operation can be paginated, False otherwise.

*dns* / Client / change\_resource\_record\_sets

### change\_resource\_record\_sets

`dns.Client.change_resource_record_sets(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.change_resource_record_sets(
    Comment='string',
    RequestChangeBatch={
        'ListOfRequestChanges': [
            {
                'RequestChange': {
                    'Action': 'CREATE'|'DELETE',
                    'RequestResourceRecordSet': {
                        'Failover': 'PRIMARY'|'SECONDARY',
                        'ListOfRequestResourceRecords': [
                            {
                                'RequestResourceRecord': {
                                    'Value': 'string'
                                }
                            }
                        ],
                    },
                },
            ],
        'Name': 'string',
        'Region': 'string',
        'RequestXniftyHealthCheckConfig': {
            'FullyQualifiedDomainName': 'string',
            'IPAddress': 'string',
            'Port': 123,
            'Protocol': 'HTTP'|'HTTPS'|'TCP',
            'ResourcePath': 'string'
        },
        'SetIdentifier': 'string',
        'TTL': 123,
        'Type': 'NS'|'A'|'AAAA'|'CNAME'|'MX'|'TXT'|'PTR',
        'Weight': 123,
```

(continues on next page)



(continued from previous page)

```

        'XniftyComment': 'string'
    }
    },
    ],
    ZoneID='string'
)

```

**Parameters**

- **Comment** (*string*) –
- **RequestChangeBatch** (*dict*) – [REQUIRED]
  - **ListOfRequestChanges** (*list*) – [REQUIRED]
    - \* (*dict*) –
      - **RequestChange** (*dict*) – [REQUIRED]
      - **Action** (*string*) – [REQUIRED]
      - **RequestResourceRecordSet** (*dict*) –
      - **Failover** (*string*) –
      - **ListOfRequestResourceRecords** (*list*) –
      - (*dict*) –
      - **RequestResourceRecord** (*dict*) –
      - **Value** (*string*) –
      - **Name** (*string*) –
      - **Region** (*string*) –
      - **RequestXniftyHealthCheckConfig** (*dict*) –
      - **FullyQualifiedDomainName** (*string*) –
      - **IPAddress** (*string*) –
      - **Port** (*integer*) –
      - **Protocol** (*string*) –
      - **ResourcePath** (*string*) –
      - **SetIdentifier** (*string*) –
      - **TTL** (*integer*) –
      - **Type** (*string*) –
      - **Weight** (*integer*) –
      - **XniftyComment** (*string*) –
- **ZoneID** (*string*) – [REQUIRED]

**Return type** dict**Returns****Response Syntax**

```

{
    'ChangeInfo': {
        'Id': 'string',
        'Status': 'string',
        'SubmittedAt': 'string'
    }
}

```

**Response Structure**

- (*dict*) –
  - **ChangeInfo** (*dict*) –
    - \* **Id** (*string*) –
    - \* **Status** (*string*) –
    - \* **SubmittedAt** (*string*) –

*dns* / Client / close

## close

`dns.Client.close()`  
Closes underlying endpoint connections.

*dns* / Client / create\_hosted\_zone

## create\_hosted\_zone

`dns.Client.create_hosted_zone(**kwargs)`  
See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.create_hosted_zone(
    CallerReference='string',
    Name='string',
    RequestHostedZoneConfig={
        'Comment': 'string'
    }
)
```

### Parameters

- **CallerReference** (*string*) –
- **Name** (*string*) – **[REQUIRED]**
- **RequestHostedZoneConfig** (*dict*) –
  - **Comment** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
  'ChangeInfo': {
    'Id': 'string',
    'Status': 'string',
    'SubmittedAt': 'string'
  },
  'DelegationSet': {
    'NameServers': [
      'string',
    ]
  },
  'HostedZone': {
    'CallerReference': 'string',
    'Config': {
      'Comment': 'string'
    },
    'Id': 'string',
    'Name': 'string',
    'ResourceRecordSetCount': 123
  }
}
```

### Response Structure

- (*dict*) –
  - **ChangeInfo** (*dict*) –
    - \* **Id** (*string*) –
    - \* **Status** (*string*) –
    - \* **SubmittedAt** (*string*) –
  - **DelegationSet** (*dict*) –
    - \* **NameServers** (*list*) –
      - (*string*) –
  - **HostedZone** (*dict*) –
    - \* **CallerReference** (*string*) –
    - \* **Config** (*dict*) –
      - **Comment** (*string*) –
    - \* **Id** (*string*) –
    - \* **Name** (*string*) –
    - \* **ResourceRecordSetCount** (*integer*) –

*dns* / Client / delete\_hosted\_zone

## delete\_hosted\_zone

`dns.Client.delete_hosted_zone(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_hosted_zone(
    ZoneID='string'
)
```

**Parameters** `ZoneID` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
  'ChangeInfo': {
    'Id': 'string',
    'Status': 'string',
    'SubmittedAt': 'string'
  }
}
```

### Response Structure

- (*dict*) –
  - **ChangeInfo** (*dict*) –
    - \* **Id** (*string*) –
    - \* **Status** (*string*) –
    - \* **SubmittedAt** (*string*) –

*dns* / Client / get\_change

## get\_change

`dns.Client.get_change(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.get_change(  
    ChangeID='string'  
)
```

**Parameters** **ChangeID** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'ChangeInfo': {  
        'Id': 'string',  
        'Status': 'string',  
        'SubmittedAt': 'string'  
    }  
}
```

### Response Structure

- (*dict*) –
  - **ChangeInfo** (*dict*) –
    - \* **Id** (*string*) –
    - \* **Status** (*string*) –
    - \* **SubmittedAt** (*string*) –

*dns* / Client / get\_hosted\_zone

## get\_hosted\_zone

`dns.Client.get_hosted_zone(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.get_hosted_zone(  
    ZoneID='string'  
)
```

**Parameters** **ZoneID** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'DelegationSet': {  
        'NameServers': [  
            'string',  
        ]  
    },  
    'HostedZone': {  
        'CallerReference': 'string',  
        'Config': {  
            'Comment': 'string'  
        },  
    },  
}
```

(continues on next page)

(continued from previous page)

```

        'Id': 'string',
        'Name': 'string',
        'ResourceRecordSetCount': 123
    }
}

```

**Response Structure**

- (dict) –
  - **DelegationSet** (dict) –
    - \* **NameServers** (list) –
      - (string) –
  - **HostedZone** (dict) –
    - \* **CallerReference** (string) –
    - \* **Config** (dict) –
      - **Comment** (string) –
    - \* **Id** (string) –
    - \* **Name** (string) –
    - \* **ResourceRecordSetCount** (integer) –

*dns* / Client / get\_paginator**get\_paginator**`dns.Client.get_paginator(operation_name)`

Create a paginator for an operation.

**Parameters** `operation_name` (string) – The operation name. This is the same name as the method name on the client. For example, if the method name is `create_foo`, and you'd normally invoke the operation as `client.create_foo(**kwargs)`, if the `create_foo` operation can be paginated, you can use the call `client.get_paginator("create_foo").`

**Raises** **OperationNotPageableError** – Raised if the operation is not pageable. You can use the `client.can_paginate` method to check if an operation is pageable.

**Return type** L{botocore.paginate.Paginator}

**Returns** A paginator object.

*dns* / Client / get\_waiter**get\_waiter**`dns.Client.get_waiter(waiter_name)`

Returns an object that can wait for some condition.

**Parameters** `waiter_name` (str) – The name of the waiter to get. See the waiters section of the service docs for a list of available waiters.

**Returns** The specified waiter object.

**Return type** botocore.waiter.Waiter

*dns* / Client / list\_hosted\_zones**list\_hosted\_zones**`dns.Client.list_hosted_zones(**kwargs)`See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.list_hosted_zones(  
    Marker='string',  
    MaxItems=123  
)
```

#### Parameters

- **Marker** (*string*) –
- **MaxItems** (*integer*) –

**Return type** dict

#### Returns

### Response Syntax

```
{  
    'HostedZones': [  
        {  
            'CallerReference': 'string',  
            'Config': {  
                'Comment': 'string'  
            },  
            'Id': 'string',  
            'Name': 'string',  
            'ResourceRecordSetCount': 123  
        },  
    ],  
    'IsTruncated': True|False,  
    'Marker': 'string',  
    'MaxItems': 123,  
    'NextMarker': 'string'  
}
```

### Response Structure

- (*dict*) –
  - **HostedZones** (*list*) –
    - \* (*dict*) –
      - **CallerReference** (*string*) –
      - **Config** (*dict*) –
      - **Comment** (*string*) –
      - **Id** (*string*) –
      - **Name** (*string*) –
      - **ResourceRecordSetCount** (*integer*) –
    - **IsTruncated** (*boolean*) –
    - **Marker** (*string*) –
    - **MaxItems** (*integer*) –
    - **NextMarker** (*string*) –

*dns* / Client / list\_resource\_record\_sets

### list\_resource\_record\_sets

`dns.Client.list_resource_record_sets(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.list_resource_record_sets(
    Identifier='string',
    Maxitems=123,
    Name='string',
    Type='NS'|'A'|'AAAA'|'CNAME'|'MX'|'TXT'|'PTR',
    ZoneID='string'
)
```

#### Parameters

- **Identifier** (*string*) –
- **Maxitems** (*integer*) –
- **Name** (*string*) –
- **Type** (*string*) –
- **ZoneID** (*string*) – [REQUIRED]

**Return type** dict

#### Returns

#### Response Syntax

```
{
  'IsTruncated': True|False,
  'MaxItems': 123,
  'NextRecordIdentifier': 'string',
  'NextRecordName': 'string',
  'NextRecordType': 'string',
  'ResourceRecordSets': [
    {
      'Failover': 'string',
      'Name': 'string',
      'Region': 'string',
      'ResourceRecords': [
        {
          'Value': 'string'
        },
      ],
      'SetIdentifier': 'string',
      'TTL': 123,
      'Type': 'string',
      'Weight': 123,
      'XniftyComment': 'string',
      'XniftyHealthCheckConfig': {
        'FullyQualifiedDomainName': 'string',
        'IPAddress': 'string',
        'Port': 123,
        'Protocol': 'string',
        'ResourcePath': 'string'
      }
    },
  ]
}
```

#### Response Structure

- (*dict*) –
  - **IsTruncated** (*boolean*) –
  - **MaxItems** (*integer*) –
  - **NextRecordIdentifier** (*string*) –
  - **NextRecordName** (*string*) –

- **NextRecordType** (*string*) –
- **ResourceRecordSets** (*list*) –
  - \* (*dict*) –
    - **Failover** (*string*) –
    - **Name** (*string*) –
    - **Region** (*string*) –
    - **ResourceRecords** (*list*) –
    - (*dict*) –
    - **Value** (*string*) –
    - **SetIdentifier** (*string*) –
    - **TTL** (*integer*) –
    - **Type** (*string*) –
    - **Weight** (*integer*) –
    - **XniftyComment** (*string*) –
    - **XniftyHealthCheckConfig** (*dict*) –
    - **FullyQualifiedDomainName** (*string*) –
    - **IPAddress** (*string*) –
    - **Port** (*integer*) –
    - **Protocol** (*string*) –
    - **ResourcePath** (*string*) –

## 1.2.2 Client Exceptions

Client exceptions are available on a client instance via the `exceptions` property. For more detailed instructions and examples on the exact usage of client exceptions, see the error handling [user guide](#).

This client has no modeled exception classes.

## 1.3 ess

### 1.3.1 Client

**class** `ess.Client`

A low-level client representing NIFCLOUD ESS

```
client = session.create_client('ess')
```

These are the available methods:

`ess / Client / can_paginate`

#### `can_paginate`

`ess.Client.can_paginate(operation_name)`

Check if an operation can be paginated.

**Parameters** `operation_name` (*string*) – The operation name. This is the same name as the method name on the client. For example, if the method name is `create_foo`, and you'd normally invoke the operation as `client.create_foo(**kwargs)`, if the `create_foo` operation can be paginated, you can use the call `client.get_paginator("create_foo")`.

**Returns** True if the operation can be paginated, False otherwise.



*ess* / Client / close

## close

`ess.Client.close()`  
Closes underlying endpoint connections.

*ess* / Client / delete\_identity

## delete\_identity

`ess.Client.delete_identity(**kwargs)`  
See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_identity(
    Identity='string'
)
```

**Parameters** `Identity` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
    'DeleteIdentityResult': 'string',
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

### Response Structure

- (*dict*) –
  - **DeleteIdentityResult** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*ess* / Client / get\_delivery\_log

## get\_delivery\_log

`ess.Client.get_delivery_log(**kwargs)`  
See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.get_delivery_log(
    EndDate=datetime(2015, 1, 1),
    MaxItems=123,
    NextToken='string',
    StartDate=datetime(2015, 1, 1),
    Status=123
)
```

**Parameters**

- **EndDate** (*datetime*) – [REQUIRED]
- **MaxItems** (*integer*) –
- **NextToken** (*string*) –
- **StartDate** (*datetime*) – [REQUIRED]
- **Status** (*integer*) –

**Return type** dict**Returns****Response Syntax**

```
{
    'Log': 'string',
    'LogCount': 'string',
    'NextToken': 'string',
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

**Response Structure**

- (*dict*) –
  - **Log** (*string*) –
  - **LogCount** (*string*) –
  - **NextToken** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*ess* / Client / get\_identity\_dkim\_attributes**get\_identity\_dkim\_attributes***ess*.Client.**get\_identity\_dkim\_attributes** (\*\**kwargs*)See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.get_identity_dkim_attributes(
    Identities=[
        'string',
    ]
)
```

**Parameters** **Identities** (*list*) – [REQUIRED]

- (*string*) –

**Return type** dict**Returns****Response Syntax**

```
{
    'DkimAttributes': [
        {
            'Key': 'string',
            'Value': {
                'DkimEnabled': True|False,
                'DkimTokens': [
```

(continues on next page)

(continued from previous page)

```

        'string',
    ],
    'DkimVerificationStatus': 'string'
}
},
],
'ResponseMetadata': {
    'RequestId': 'string'
}
}

```

**Response Structure**

- (dict) –
  - **DkimAttributes** (list) –
    - \* (dict) –
      - **Key** (string) –
      - **Value** (dict) –
      - **DkimEnabled** (boolean) –
      - **DkimTokens** (list) –
      - (string) –
      - **DkimVerificationStatus** (string) –
  - **ResponseMetadata** (dict) –
    - \* **RequestId** (string) –

`ess / Client / get_identity_verification_attributes`

**get\_identity\_verification\_attributes**

`ess.Client.get_identity_verification_attributes (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```

response = client.get_identity_verification_attributes(
    Identities=[
        'string',
    ]
)

```

**Parameters** **Identities** (list) – [REQUIRED]

- (string) –

**Return type** dict

**Returns**

**Response Syntax**

```

{
    'ResponseMetadata': {
        'RequestId': 'string'
    },
    'VerificationAttributes': [
        {
            'Key': 'string',
            'Value': {
                'VerificationStatus': 'string',
            }
        }
    ]
}

```

(continues on next page)

(continued from previous page)

```

        'VerificationToken': 'string'
    },
],
}

```

**Response Structure**

- *(dict)* –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –
  - **VerificationAttributes** (*list*) –
    - \* (*dict*) –
      - **Key** (*string*) –
      - **Value** (*dict*) –
      - **VerificationStatus** (*string*) –
      - **VerificationToken** (*string*) –

*ess* / Client / `get_paginator`**get\_paginator**`ess.Client.get_paginator(operation_name)`

Create a paginator for an operation.

**Parameters** `operation_name` (*string*) – The operation name. This is the same name as the method name on the client. For example, if the method name is `create_foo`, and you'd normally invoke the operation as `client.create_foo(**kwargs)`, if the `create_foo` operation can be paginated, you can use the call `client.get_paginator("create_foo")`.

**Raises** **OperationNotPageableError** – Raised if the operation is not pageable. You can use the `client.can_paginate` method to check if an operation is pageable.

**Return type** `L{botocore.paginate.Paginator}`

**Returns** A paginator object.

*ess* / Client / `get_send_quota`**get\_send\_quota**`ess.Client.get_send_quota()`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.get_send_quota()
```

**Return type** `dict`**Returns****Response Syntax**

```

{
    'Max24HourSend': 123.0,
    'MaxSendRate': 123.0,
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}

```

(continues on next page)

(continued from previous page)

```

    },
    'SentLast24Hours': 123.0
}

```

**Response Structure**

- (*dict*) –
  - **Max24HourSend** (*float*) –
  - **MaxSendRate** (*float*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –
  - **SentLast24Hours** (*float*) –

*ess* / Client / `get_send_statistics`**get\_send\_statistics**`ess.Client.get_send_statistics()`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.get_send_statistics()
```

**Return type** dict**Returns****Response Syntax**

```

{
    'ResponseMetadata': {
        'RequestId': 'string'
    },
    'SendDataPoints': [
        {
            'Bounces': 123,
            'Complaints': 123,
            'DeliveryAttempts': 123,
            'Rejects': 123,
            'Timestamp': datetime(2015, 1, 1)
        },
    ]
}

```

**Response Structure**

- (*dict*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –
  - **SendDataPoints** (*list*) –
    - \* (*dict*) –
      - **Bounces** (*integer*) –
      - **Complaints** (*integer*) –
      - **DeliveryAttempts** (*integer*) –
      - **Rejects** (*integer*) –
      - **Timestamp** (*datetime*) –

*ess* / Client / `get_waiter`

## get\_waiter

`ess.Client.get_waiter(waiter_name)`

Returns an object that can wait for some condition.

**Parameters** `waiter_name` (*str*) – The name of the waiter to get. See the waiters section of the service docs for a list of available waiters.

**Returns** The specified waiter object.

**Return type** `botocore.waiter.Waiter`

*ess* / Client / list\_identities

## list\_identities

`ess.Client.list_identities(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.list_identities(
    IdentityType='EmailAddress'|'Domain',
    MaxItems=123,
    NextToken='string'
)
```

### Parameters

- **IdentityType** (*string*) –
- **MaxItems** (*integer*) –
- **NextToken** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
  'Identities': [
    'string',
  ],
  'NextToken': 'string',
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

### Response Structure

- (*dict*) –
  - **Identities** (*list*) –
    - \* (*string*) –
  - **NextToken** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*ess* / Client / send\_email

## send\_email

`ess.Client.send_email(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```

response = client.send_email(
    Destination={
        'ListOfRequestBccAddresses': [
            'string',
        ],
        'ListOfRequestCcAddresses': [
            'string',
        ],
        'ListOfRequestToAddresses': [
            'string',
        ]
    },
    Message={
        'RequestBody': {
            'RequestHtml': {
                'Charset': 'string',
                'Data': 'string'
            },
            'RequestText': {
                'Charset': 'string',
                'Data': 'string'
            }
        },
        'RequestSubject': {
            'Charset': 'string',
            'Data': 'string'
        }
    },
    ReplyToAddresses=[
        'string',
    ],
    ReturnPath='string',
    Source='string'
)

```

### Parameters

- **Destination** (*dict*) –
  - **ListOfRequestBccAddresses** (*list*) –
    - \* (*string*) –
  - **ListOfRequestCcAddresses** (*list*) –
    - \* (*string*) –
  - **ListOfRequestToAddresses** (*list*) –
    - \* (*string*) –
- **Message** (*dict*) –
  - **RequestBody** (*dict*) –
    - \* **RequestHtml** (*dict*) –
      - **Charset** (*string*) –
      - **Data** (*string*) –
    - \* **RequestText** (*dict*) –
      - **Charset** (*string*) –

- **Data** (*string*) –
- **RequestSubject** (*dict*) –
  - \* **Charset** (*string*) –
  - \* **Data** (*string*) –
- **ReplyToAddresses** (*list*) –
  - (*string*) –
- **ReturnPath** (*string*) –
- **Source** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'MessageId': 'string',
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

#### Response Structure

- (*dict*) –
  - **MessageId** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*ess* / Client / send\_raw\_email

### send\_raw\_email

`ess.Client.send_raw_email(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.send_raw_email(
    Destinations=[
        'string',
    ],
    RawMessage={
        'Data': 'string'
    },
    Source='string'
)
```

#### Parameters

- **Destinations** (*list*) –
  - (*string*) –
- **RawMessage** (*dict*) – [REQUIRED]
  - **Data** (*string*) – [REQUIRED]
- **Source** (*string*) –

**Return type** dict

**Returns**

#### Response Syntax



```
{
    'MessageId': 'string',
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

**Response Structure**

- (*dict*) –
  - **MessageId** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*ess* / Client / set\_identity\_dkim\_enabled

**set\_identity\_dkim\_enabled**

`ess.Client.set_identity_dkim_enabled(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.set_identity_dkim_enabled(
    DkimEnabled='true'|'false',
    Identity='string'
)
```

**Parameters**

- **DkimEnabled** (*string*) – [REQUIRED]
- **Identity** (*string*) – [REQUIRED]

**Return type** dict

**Returns****Response Syntax**

```
{
    'ResponseMetadata': {
        'RequestId': 'string'
    },
    'SetIdentityDkimEnabledResult': 'string'
}
```

**Response Structure**

- (*dict*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –
  - **SetIdentityDkimEnabledResult** (*string*) –

*ess* / Client / verify\_domain\_dkim

**verify\_domain\_dkim**

`ess.Client.verify_domain_dkim(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.verify_domain_dkim(
    Domain='string'
)
```

**Parameters** `Domain` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'DkimTokens': [
        'string',
    ],
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

#### Response Structure

- (*dict*) –
  - **DkimTokens** (*list*) –
    - \* (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*ess* / Client / `verify_domain_identity`

### `verify_domain_identity`

`ess.Client.verify_domain_identity` (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.verify_domain_identity(
    Domain='string'
)
```

**Parameters** `Domain` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'ResponseMetadata': {
        'RequestId': 'string'
    },
    'VerificationToken': 'string'
}
```

#### Response Structure

- (*dict*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –
  - **VerificationToken** (*string*) –

*ess* / Client / `verify_email_identity`

## verify\_email\_identity

`ess.Client.verify_email_identity(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.verify_email_identity(
    EmailAddress='string'
)
```

**Parameters** `EmailAddress` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
    'ResponseMetadata': {
        'RequestId': 'string'
    },
    'VerifyEmailIdentityResult': 'string'
}
```

### Response Structure

- (*dict*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –
  - **VerifyEmailIdentityResult** (*string*) –

## 1.3.2 Client Exceptions

Client exceptions are available on a client instance via the `exceptions` property. For more detailed instructions and examples on the exact usage of client exceptions, see the error handling [user guide](#).

This client has no modeled exception classes.

## 1.4 hatoba

### 1.4.1 Client

**class** `hatoba.Client`

A low-level client representing NIFCLOUD Kubernetes Service Hatoba

```
client = session.create_client('hatoba')
```

These are the available methods:

*hatoba* / Client / `attach_disk`

## attach\_disk

hatoba.Client.**attach\_disk** (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.attach_disk(  
    DiskName='string',  
    NodeName='string'  
)
```

### Parameters

- **DiskName** (*string*) – [REQUIRED]
- **NodeName** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'Disk': {  
        'Attachments': [  
            {  
                'AttachTime': 'string',  
                'DevicePath': 'string',  
                'NodeName': 'string',  
                'Status': 'string'  
            },  
        ],  
        'AvailabilityZone': 'string',  
        'Cluster': {  
            'Name': 'string'  
        },  
        'CreateTime': 'string',  
        'Description': 'string',  
        'Name': 'string',  
        'Nrn': 'string',  
        'Size': 123,  
        'Status': 'string',  
        'Tags': [  
            {  
                'Id': 'string',  
                'Key': 'string',  
                'Value': 'string'  
            },  
        ],  
        'Type': 'string'  
    }  
}
```

### Response Structure

- (*dict*) –
  - **Disk** (*dict*) –
    - \* **Attachments** (*list*) –
      - (*dict*) –
        - **AttachTime** (*string*) –
        - **DevicePath** (*string*) –

- **NodeName** (*string*) –
- **Status** (*string*) –
- \* **AvailabilityZone** (*string*) –
- \* **Cluster** (*dict*) –
  - **Name** (*string*) –
- \* **CreateTime** (*string*) –
- \* **Description** (*string*) –
- \* **Name** (*string*) –
- \* **Nrn** (*string*) –
- \* **Size** (*integer*) –
- \* **Status** (*string*) –
- \* **Tags** (*list*) –
  - (*dict*) –
  - **Id** (*string*) –
  - **Key** (*string*) –
  - **Value** (*string*) –
- \* **Type** (*string*) –

*hatoba* / Client / authorize\_firewall\_group

## authorize\_firewall\_group

`hatoba.Client.authorize_firewall_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.authorize_firewall_group(
    FirewallGroupName='string',
    Rules=[
        {
            'CidrIp': 'string',
            'Description': 'string',
            'Direction': 'IN'|'OUT',
            'FromPort': 123,
            'Protocol': 'ANY'|'TCP'|'UDP'|'ICMP'|'SSH'|'HTTP'|'HTTPS'|'RDP'|'GRE'|
↪ 'ESP'|'AH'|'VRRP'|'L2TP',
            'ToPort': 123
        },
    ]
)
```

### Parameters

- **FirewallGroupName** (*string*) – [REQUIRED]
- **Rules** (*list*) – [REQUIRED]
  - (*dict*) –
    - \* **CidrIp** (*string*) – [REQUIRED]
    - \* **Description** (*string*) –
    - \* **Direction** (*string*) –
    - \* **FromPort** (*integer*) –
    - \* **Protocol** (*string*) –
    - \* **ToPort** (*integer*) –

Return type dict

Returns

### Response Syntax

```
{
  'FirewallGroup': {
    'Description': 'string',
    'Name': 'string',
    'Nrn': 'string',
    'Rules': [
      {
        'CidrIp': 'string',
        'Description': 'string',
        'Direction': 'string',
        'FromPort': 123,
        'Id': 'string',
        'Protocol': 'string',
        'Status': 'string',
        'ToPort': 123
      },
    ],
    'Tags': [
      {
        'Id': 'string',
        'Key': 'string',
        'Value': 'string'
      },
    ],
  }
}
```

#### Response Structure

- (dict) –
  - **FirewallGroup** (dict) –
    - \* **Description** (string) –
    - \* **Name** (string) –
    - \* **Nrn** (string) –
    - \* **Rules** (list) –
      - (dict) –
      - **CidrIp** (string) –
      - **Description** (string) –
      - **Direction** (string) –
      - **FromPort** (integer) –
      - **Id** (string) –
      - **Protocol** (string) –
      - **Status** (string) –
      - **ToPort** (integer) –
    - \* **Tags** (list) –
      - (dict) –
      - **Id** (string) –
      - **Key** (string) –
      - **Value** (string) –

*hatoba* / Client / can\_paginate

#### can\_paginate

`hatoba.Client.can_paginate(operation_name)`

Check if an operation can be paginated.

**Parameters** `operation_name` (*string*) – The operation name. This is the same name as the method name on the client. For example, if the method name is `create_foo`, and you'd normally invoke the operation as `client.create_foo(**kwargs)`, if the `create_foo` operation can be paginated, you can use the call `client.get_paginator("create_foo")`.

**Returns** True if the operation can be paginated, False otherwise.

*hatoba* / Client / close

## close

`hatoba.Client.close()`

Closes underlying endpoint connections.

*hatoba* / Client / create\_cluster

## create\_cluster

`hatoba.Client.create_cluster(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.create_cluster(
    Cluster={
        'Description': 'string',
        'FirewallGroup': 'string',
        'KubernetesVersion': 'v1.23.3'|'v1.23.9'|'v1.24.3',
        'ListOfRequestLocations': [
            'string',
        ],
        'ListOfRequestNodePools': [
            {
                'InstanceType': 'c-medium'|'e-medium'|'medium'|'c-medium4'|'e-
↪medium4'|'medium4'|'c-medium8'|'e-medium8'|'medium8'|'e-medium16'|'medium16'|'e-
↪medium24'|'medium24'|'c-large'|'e-large'|'large'|'c-large8'|'e-large8'|'large8'|
↪'e-large16'|'large16'|'e-large24'|'large24'|'e-large32'|'large32'|'e-extra-
↪large8'|'extra-large8'|'e-extra-large16'|'extra-large16'|'e-extra-large24'|
↪'extra-large24'|'e-extra-large32'|'extra-large32'|'e-extra-large48'|'extra-
↪large48'|'e-double-large16'|'double-large16'|'e-double-large24'|'double-large24
↪'|'e-double-large32'|'double-large32'|'e-double-large48'|'double-large48'|'e-
↪double-large64'|'double-large64'|'e-double-large96'|'double-large96',
                'ListOfRequestTags': [
                    {
                        'Key': 'string',
                        'Value': 'string'
                    },
                ],
                'Name': 'string',
                'NodeCount': 123
            },
        ],
        'ListOfRequestTags': [
            {
                'Key': 'string',
                'Value': 'string'
            },
        ],
    },
    ListOfRequestTags=[
        {
            'Key': 'string',
            'Value': 'string'
        },
    ],
)
```

(continues on next page)

(continued from previous page)

```

    },
  ],
  'Name': 'string',
  'RequestAddonsConfig': {
    'RequestHttpLoadBalancing': {
      'Disabled': True|False
    }
  },
  'RequestNetworkConfig': {
    'NetworkId': 'string'
  }
}
)

```

**Parameters Cluster** (*dict*) – [REQUIRED]

- **Description** (*string*) –
- **FirewallGroup** (*string*) – [REQUIRED]
- **KubernetesVersion** (*string*) –
- **ListOfRequestLocations** (*list*) – [REQUIRED]
  - (*string*) –
- **ListOfRequestNodePools** (*list*) – [REQUIRED]
  - (*dict*) –
    - \* **InstanceType** (*string*) – [REQUIRED]
    - \* **ListOfRequestTags** (*list*) –
      - (*dict*) –
      - **Key** (*string*) –
      - **Value** (*string*) –
    - \* **Name** (*string*) – [REQUIRED]
    - \* **NodeCount** (*integer*) –
- **ListOfRequestTags** (*list*) –
  - (*dict*) –
    - \* **Key** (*string*) –
    - \* **Value** (*string*) –
- **Name** (*string*) – [REQUIRED]
- **RequestAddonsConfig** (*dict*) –
  - **RequestHttpLoadBalancing** (*dict*) –
    - \* **Disabled** (*boolean*) –
- **RequestNetworkConfig** (*dict*) –
  - **NetworkId** (*string*) –

**Return type** dict**Returns****Response Syntax**

```

{
  'Cluster': {
    'AddonsConfig': {
      'HttpLoadBalancing': {
        'Disabled': True|False
      }
    },
    'CreateTime': 'string',
    'Description': 'string',
    'FirewallGroup': 'string',
    'InitialKubernetesVersion': 'string',

```

(continues on next page)



(continued from previous page)

```

'InitialNodeCount': 123,
'KubernetesVersion': 'string',
'Locations': [
    'string',
],
'Name': 'string',
'NetworkConfig': {
    'NetworkId': 'string'
},
'NodeCount': 123,
'NodePools': [
    {
        'InitialNodeCount': 123,
        'InstanceType': 'string',
        'Name': 'string',
        'NodeCount': 123,
        'Nodes': [
            {
                'AvailabilityZone': 'string',
                'Name': 'string',
                'PrivateIpAddress': 'string',
                'PublicIpAddress': 'string',
                'Status': 'string'
            },
        ],
        'Nrn': 'string',
        'Status': 'string',
        'Tags': [
            {
                'Id': 'string',
                'Key': 'string',
                'Value': 'string'
            },
        ],
    },
],
'Nrn': 'string',
'Status': 'string',
'Tags': [
    {
        'Id': 'string',
        'Key': 'string',
        'Value': 'string'
    },
],
]
}

```

**Response Structure**

- (dict) –
  - **Cluster** (dict) –
    - \* **AddonsConfig** (dict) –
      - **HttpLoadBalancing** (dict) –
      - **Disabled** (boolean) –
    - \* **CreateTime** (string) –
    - \* **Description** (string) –
    - \* **FirewallGroup** (string) –

- \* **InitialKubernetesVersion** (*string*) –
- \* **InitialNodeCount** (*integer*) –
- \* **KubernetesVersion** (*string*) –
- \* **Locations** (*list*) –
  - (*string*) –
- \* **Name** (*string*) –
- \* **NetworkConfig** (*dict*) –
  - **NetworkId** (*string*) –
- \* **NodeCount** (*integer*) –
- \* **NodePools** (*list*) –
  - (*dict*) –
  - **InitialNodeCount** (*integer*) –
  - **InstanceType** (*string*) –
  - **Name** (*string*) –
  - **NodeCount** (*integer*) –
  - **Nodes** (*list*) –
  - (*dict*) –
  - **AvailabilityZone** (*string*) –
  - **Name** (*string*) –
  - **PrivateIpAddress** (*string*) –
  - **PublicIpAddress** (*string*) –
  - **Status** (*string*) –
  - **Nrn** (*string*) –
  - **Status** (*string*) –
  - **Tags** (*list*) –
  - (*dict*) –
  - **Id** (*string*) –
  - **Key** (*string*) –
  - **Value** (*string*) –
- \* **Nrn** (*string*) –
- \* **Status** (*string*) –
- \* **Tags** (*list*) –
  - (*dict*) –
  - **Id** (*string*) –
  - **Key** (*string*) –
  - **Value** (*string*) –

*hatoba* / Client / create\_disk

## create\_disk

`hatoba.Client.create_disk(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.create_disk(  
    Disk={  
        'AvailabilityZone': 'string',  
        'Description': 'string',  
        'ListOfRequestTags': [  
            {  
                'Key': 'string',  
                'Value': 'string'
```

(continues on next page)

(continued from previous page)

```

        },
    ],
    'Name': 'string',
    'Size': 123,
    'Type': 'standard-flash-a'|'standard-flash-b'|'high-speed-flash-a'|'high-
↪speed-flash-b'
    }
)

```

**Parameters `Disk` (*dict*) – [REQUIRED]**

- **AvailabilityZone** (*string*) –
- **Description** (*string*) –
- **ListOfRequestTags** (*list*) –
  - (*dict*) –
    - \* **Key** (*string*) –
    - \* **Value** (*string*) –
- **Name** (*string*) – [REQUIRED]
- **Size** (*integer*) – [REQUIRED]
- **Type** (*string*) – [REQUIRED]

**Return type** dict**Returns****Response Syntax**

```

{
    'Disk': {
        'Attachments': [
            {
                'AttachTime': 'string',
                'DevicePath': 'string',
                'NodeName': 'string',
                'Status': 'string'
            },
        ],
        'AvailabilityZone': 'string',
        'Cluster': {
            'Name': 'string'
        },
        'CreateTime': 'string',
        'Description': 'string',
        'Name': 'string',
        'Nrn': 'string',
        'Size': 123,
        'Status': 'string',
        'Tags': [
            {
                'Id': 'string',
                'Key': 'string',
                'Value': 'string'
            },
        ],
        'Type': 'string'
    }
}

```

**Response Structure**

- (*dict*) –

- **Disk** (*dict*) –
  - \* **Attachments** (*list*) –
    - (*dict*) –
    - **AttachTime** (*string*) –
    - **DevicePath** (*string*) –
    - **NodeName** (*string*) –
    - **Status** (*string*) –
  - \* **AvailabilityZone** (*string*) –
  - \* **Cluster** (*dict*) –
    - **Name** (*string*) –
  - \* **CreateTime** (*string*) –
  - \* **Description** (*string*) –
  - \* **Name** (*string*) –
  - \* **Nrn** (*string*) –
  - \* **Size** (*integer*) –
  - \* **Status** (*string*) –
  - \* **Tags** (*list*) –
    - (*dict*) –
    - **Id** (*string*) –
    - **Key** (*string*) –
    - **Value** (*string*) –
  - \* **Type** (*string*) –

*hatoba* / Client / create\_firewall\_group

## create\_firewall\_group

`hatoba.Client.create_firewall_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.create_firewall_group(
    FirewallGroup={
        'Description': 'string',
        'ListOfRequestTags': [
            {
                'Key': 'string',
                'Value': 'string'
            },
        ],
        'Name': 'string'
    }
)
```

**Parameters** **FirewallGroup** (*dict*) – [REQUIRED]

- **Description** (*string*) –
- **ListOfRequestTags** (*list*) –
  - (*dict*) –
    - \* **Key** (*string*) –
    - \* **Value** (*string*) –
- **Name** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
  'FirewallGroup': {
    'Description': 'string',
    'Name': 'string',
    'Nrn': 'string',
    'Rules': [
      {
        'CidrIp': 'string',
        'Description': 'string',
        'Direction': 'string',
        'FromPort': 123,
        'Id': 'string',
        'Protocol': 'string',
        'Status': 'string',
        'ToPort': 123
      },
    ],
    'Tags': [
      {
        'Id': 'string',
        'Key': 'string',
        'Value': 'string'
      },
    ],
  }
}
```

#### Response Structure

- *(dict)* –
  - **FirewallGroup** (*dict*) –
    - \* **Description** (*string*) –
    - \* **Name** (*string*) –
    - \* **Nrn** (*string*) –
    - \* **Rules** (*list*) –
      - (*dict*) –
      - **CidrIp** (*string*) –
      - **Description** (*string*) –
      - **Direction** (*string*) –
      - **FromPort** (*integer*) –
      - **Id** (*string*) –
      - **Protocol** (*string*) –
      - **Status** (*string*) –
      - **ToPort** (*integer*) –
    - \* **Tags** (*list*) –
      - (*dict*) –
      - **Id** (*string*) –
      - **Key** (*string*) –
      - **Value** (*string*) –

*hatoba* / Client / create\_node\_pool

#### create\_node\_pool

`hatoba.Client.create_node_pool(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

## Request Syntax

```
response = client.create_node_pool(
    ClusterName='string',
    NodePool={
        'InstanceType': 'c-medium'|'e-medium'|'medium'|'c-medium4'|'e-medium4'|
        ↪ 'medium4'|'c-medium8'|'e-medium8'|'medium8'|'e-medium16'|'medium16'|'e-medium24'
        ↪ '| 'medium24'|'c-large'|'e-large'|'large'|'c-large8'|'e-large8'|'large8'|'e-
        ↪ large16'|'large16'|'e-large24'|'large24'|'e-large32'|'large32'|'e-extra-large8'|
        ↪ 'extra-large8'|'e-extra-large16'|'extra-large16'|'e-extra-large24'|'extra-
        ↪ large24'|'e-extra-large32'|'extra-large32'|'e-extra-large48'|'extra-large48'|'e-
        ↪ double-large16'|'double-large16'|'e-double-large24'|'double-large24'|'e-double-
        ↪ large32'|'double-large32'|'e-double-large48'|'double-large48'|'e-double-large64'
        ↪ '| 'double-large64'|'e-double-large96'|'double-large96',
        'ListOfRequestTags': [
            {
                'Key': 'string',
                'Value': 'string'
            },
        ],
        'Name': 'string',
        'NodeCount': 123
    }
)
```

### Parameters

- **ClusterName** (*string*) – [REQUIRED]
- **NodePool** (*dict*) – [REQUIRED]
  - **InstanceType** (*string*) – [REQUIRED]
  - **ListOfRequestTags** (*list*) –
    - \* (*dict*) –
      - **Key** (*string*) –
      - **Value** (*string*) –
  - **Name** (*string*) – [REQUIRED]
  - **NodeCount** (*integer*) –

**Return type** dict

**Returns**

### Response Syntax

```
{
    'NodePool': {
        'InitialNodeCount': 123,
        'InstanceType': 'string',
        'Name': 'string',
        'NodeCount': 123,
        'Nodes': [
            {
                'AvailabilityZone': 'string',
                'Name': 'string',
                'PrivateIpAddress': 'string',
                'PublicIpAddress': 'string',
                'Status': 'string'
            },
        ],
        'Nrn': 'string',
        'Status': 'string',
    },
}
```

(continues on next page)

(continued from previous page)

```

        'Tags': [
            {
                'Id': 'string',
                'Key': 'string',
                'Value': 'string'
            },
        ]
    }
}

```

**Response Structure**

- *(dict)* –
  - **NodePool** (*dict*) –
    - \* **InitialNodeCount** (*integer*) –
    - \* **InstanceType** (*string*) –
    - \* **Name** (*string*) –
    - \* **NodeCount** (*integer*) –
    - \* **Nodes** (*list*) –
      - *(dict)* –
      - **AvailabilityZone** (*string*) –
      - **Name** (*string*) –
      - **PrivateIpAddress** (*string*) –
      - **PublicIpAddress** (*string*) –
      - **Status** (*string*) –
    - \* **Nrn** (*string*) –
    - \* **Status** (*string*) –
    - \* **Tags** (*list*) –
      - *(dict)* –
      - **Id** (*string*) –
      - **Key** (*string*) –
      - **Value** (*string*) –

*hatoba* / Client / create\_snapshot**create\_snapshot**`hatoba.Client.create_snapshot (**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```

response = client.create_snapshot(
    Snapshot={
        'Description': 'string',
        'ExpirationTime': 'string',
        'ListOfRequestTags': [
            {
                'Key': 'string',
                'Value': 'string'
            },
        ],
        'Name': 'string',
        'RequestCluster': {
            'Name': 'string'

```

(continues on next page)

(continued from previous page)

```

    }
  }
)

```

**Parameters** `Snapshot` (*dict*) – [REQUIRED]

- **Description** (*string*) –
- **ExpirationTime** (*string*) –
- **ListOfRequestTags** (*list*) –
  - (*dict*) –
    - \* **Key** (*string*) –
    - \* **Value** (*string*) –
- **Name** (*string*) – [REQUIRED]
- **RequestCluster** (*dict*) – [REQUIRED]
  - **Name** (*string*) – [REQUIRED]

**Return type** `dict`**Returns****Response Syntax**

```

{
  'Snapshot': {
    'Cluster': {
      'KubernetesVersion': 'string',
      'Name': 'string',
      'NodePools': [
        {
          'InstanceType': 'string',
          'Name': 'string',
          'NodeCount': 123
        },
      ],
    },
    'CreateTime': 'string',
    'Description': 'string',
    'ExpirationTime': 'string',
    'Name': 'string',
    'Nrn': 'string',
    'ResourceVersion': 'string',
    'Status': 'string',
    'Tags': [
      {
        'Id': 'string',
        'Key': 'string',
        'Value': 'string'
      },
    ],
  }
}

```

**Response Structure**

- (*dict*) –
  - **Snapshot** (*dict*) –
    - \* **Cluster** (*dict*) –
      - **KubernetesVersion** (*string*) –
      - **Name** (*string*) –
      - **NodePools** (*list*) –



- (dict) –
- **InstanceType** (string) –
- **Name** (string) –
- **NodeCount** (integer) –
- \* **CreateTime** (string) –
- \* **Description** (string) –
- \* **ExpirationTime** (string) –
- \* **Name** (string) –
- \* **Nrn** (string) –
- \* **ResourceVersion** (string) –
- \* **Status** (string) –
- \* **Tags** (list) –
  - (dict) –
  - **Id** (string) –
  - **Key** (string) –
  - **Value** (string) –

*hatoba* / Client / create\_tags

## create\_tags

`hatoba.Client.create_tags(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.create_tags(
    Tags=[
        {
            'Key': 'string',
            'Nrn': 'string',
            'Value': 'string'
        },
    ]
)
```

### Parameters **Tags** (list) – [REQUIRED]

- (dict) –
  - **Key** (string) – [REQUIRED]
  - **Nrn** (string) – [REQUIRED]
  - **Value** (string) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```
{
    'Tags': [
        {
            'Id': 'string',
            'Key': 'string',
            'Nrn': 'string',
            'Value': 'string'
        },
    ]
}
```

**Response Structure**

- *(dict)* –
  - **Tags** (*list*) –
    - \* *(dict)* –
      - **Id** (*string*) –
      - **Key** (*string*) –
      - **Nrn** (*string*) –
      - **Value** (*string*) –

*hatoba* / Client / delete\_cluster

**delete\_cluster**

`hatoba.Client.delete_cluster(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.delete_cluster(  
    ClusterName='string'  
)
```

**Parameters** **ClusterName** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

**Response Syntax**

```
{  
    'Cluster': {  
        'AddonsConfig': {  
            'HttpLoadBalancing': {  
                'Disabled': True|False  
            }  
        },  
        'CreateTime': 'string',  
        'Description': 'string',  
        'FirewallGroup': 'string',  
        'InitialKubernetesVersion': 'string',  
        'InitialNodeCount': 123,  
        'KubernetesVersion': 'string',  
        'Locations': [  
            'string',  
        ],  
        'Name': 'string',  
        'NetworkConfig': {  
            'NetworkId': 'string'  
        },  
        'NodeCount': 123,  
        'NodePools': [  
            {  
                'InitialNodeCount': 123,  
                'InstanceType': 'string',  
                'Name': 'string',  
                'NodeCount': 123,  
                'Nodes': [  
                    {
```

(continues on next page)

(continued from previous page)

```

        'AvailabilityZone': 'string',
        'Name': 'string',
        'PrivateIpAddress': 'string',
        'PublicIpAddress': 'string',
        'Status': 'string'
    },
    ],
    'Nrn': 'string',
    'Status': 'string',
    'Tags': [
        {
            'Id': 'string',
            'Key': 'string',
            'Value': 'string'
        },
    ]
    },
    ],
    'Nrn': 'string',
    'Status': 'string',
    'Tags': [
        {
            'Id': 'string',
            'Key': 'string',
            'Value': 'string'
        },
    ],
    ]
}

```

**Response Structure**

- *(dict)* –
  - **Cluster** (*dict*) –
    - \* **AddonsConfig** (*dict*) –
      - **HttpLoadBalancing** (*dict*) –
      - **Disabled** (*boolean*) –
    - \* **CreateTime** (*string*) –
    - \* **Description** (*string*) –
    - \* **FirewallGroup** (*string*) –
    - \* **InitialKubernetesVersion** (*string*) –
    - \* **InitialNodeCount** (*integer*) –
    - \* **KubernetesVersion** (*string*) –
    - \* **Locations** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
    - \* **NetworkConfig** (*dict*) –
      - **NetworkId** (*string*) –
    - \* **NodeCount** (*integer*) –
    - \* **NodePools** (*list*) –
      - (*dict*) –
      - **InitialNodeCount** (*integer*) –
      - **InstanceType** (*string*) –
      - **Name** (*string*) –
      - **NodeCount** (*integer*) –
      - **Nodes** (*list*) –

- (dict) –
- **AvailabilityZone** (string) –
- **Name** (string) –
- **PrivateIpAddress** (string) –
- **PublicIpAddress** (string) –
- **Status** (string) –
- **Nrn** (string) –
- **Status** (string) –
- **Tags** (list) –
- (dict) –
- **Id** (string) –
- **Key** (string) –
- **Value** (string) –
- \* **Nrn** (string) –
- \* **Status** (string) –
- \* **Tags** (list) –
- (dict) –
- **Id** (string) –
- **Key** (string) –
- **Value** (string) –

*hatoba* / Client / delete\_clusters

## delete\_clusters

`hatoba.Client.delete_clusters(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_clusters(  
    Names='string'  
)
```

**Parameters** **Names** (string) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'Clusters': [  
        {  
            'AddonsConfig': {  
                'HttpLoadBalancing': {  
                    'Disabled': True|False  
                }  
            },  
            'CreateTime': 'string',  
            'Description': 'string',  
            'FirewallGroup': 'string',  
            'InitialKubernetesVersion': 'string',  
            'InitialNodeCount': 123,  
            'KubernetesVersion': 'string',  
            'Locations': [  
                'string',  
            ]  
        },  
    ]  
}
```

(continues on next page)

(continued from previous page)

```

    ],
    'Name': 'string',
    'NetworkConfig': {
        'NetworkId': 'string'
    },
    'NodeCount': 123,
    'NodePools': [
        {
            'InitialNodeCount': 123,
            'InstanceType': 'string',
            'Name': 'string',
            'NodeCount': 123,
            'Nodes': [
                {
                    'AvailabilityZone': 'string',
                    'Name': 'string',
                    'PrivateIpAddress': 'string',
                    'PublicIpAddress': 'string',
                    'Status': 'string'
                },
            ],
            'Nrn': 'string',
            'Status': 'string',
            'Tags': [
                {
                    'Id': 'string',
                    'Key': 'string',
                    'Value': 'string'
                },
            ],
        },
    ],
    'Nrn': 'string',
    'Status': 'string',
    'Tags': [
        {
            'Id': 'string',
            'Key': 'string',
            'Value': 'string'
        },
    ],
},
]
}

```

**Response Structure**

- *(dict)* –
  - **Clusters** *(list)* –
    - \* *(dict)* –
      - **AddonsConfig** *(dict)* –
      - **HttpLoadBalancing** *(dict)* –
      - **Disabled** *(boolean)* –
      - **CreateTime** *(string)* –
      - **Description** *(string)* –
      - **FirewallGroup** *(string)* –
      - **InitialKubernetesVersion** *(string)* –

- **InitialNodeCount** (*integer*) –
- **KubernetesVersion** (*string*) –
- **Locations** (*list*) –
- (*string*) –
- **Name** (*string*) –
- **NetworkConfig** (*dict*) –
- **NetworkId** (*string*) –
- **NodeCount** (*integer*) –
- **NodePools** (*list*) –
- (*dict*) –
- **InitialNodeCount** (*integer*) –
- **InstanceType** (*string*) –
- **Name** (*string*) –
- **NodeCount** (*integer*) –
- **Nodes** (*list*) –
- (*dict*) –
- **AvailabilityZone** (*string*) –
- **Name** (*string*) –
- **PrivateIpAddress** (*string*) –
- **PublicIpAddress** (*string*) –
- **Status** (*string*) –
- **Nrn** (*string*) –
- **Status** (*string*) –
- **Tags** (*list*) –
- (*dict*) –
- **Id** (*string*) –
- **Key** (*string*) –
- **Value** (*string*) –
- **Nrn** (*string*) –
- **Status** (*string*) –
- **Tags** (*list*) –
- (*dict*) –
- **Id** (*string*) –
- **Key** (*string*) –
- **Value** (*string*) –

*hatoba* / Client / delete\_disk

## delete\_disk

`hatoba.Client.delete_disk(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_disk(  
    DiskName='string'  
)
```

**Parameters** **DiskName** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
  'Disk': {
    'Attachments': [
      {
        'AttachTime': 'string',
        'DevicePath': 'string',
        'NodeName': 'string',
        'Status': 'string'
      },
    ],
    'AvailabilityZone': 'string',
    'Cluster': {
      'Name': 'string'
    },
    'CreateTime': 'string',
    'Description': 'string',
    'Name': 'string',
    'Nrn': 'string',
    'Size': 123,
    'Status': 'string',
    'Tags': [
      {
        'Id': 'string',
        'Key': 'string',
        'Value': 'string'
      },
    ],
    'Type': 'string'
  }
}
```

### Response Structure

- (dict) –
  - **Disk** (dict) –
    - \* **Attachments** (list) –
      - (dict) –
      - **AttachTime** (string) –
      - **DevicePath** (string) –
      - **NodeName** (string) –
      - **Status** (string) –
    - \* **AvailabilityZone** (string) –
    - \* **Cluster** (dict) –
      - **Name** (string) –
    - \* **CreateTime** (string) –
    - \* **Description** (string) –
    - \* **Name** (string) –
    - \* **Nrn** (string) –
    - \* **Size** (integer) –
    - \* **Status** (string) –
    - \* **Tags** (list) –
      - (dict) –
      - **Id** (string) –
      - **Key** (string) –
      - **Value** (string) –
    - \* **Type** (string) –

*hatoba* / Client / delete\_disks

## delete\_disks

`hatoba.Client.delete_disks(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_disks(
    Names='string'
)
```

**Parameters** `Names` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
  'Disks': [
    {
      'Attachments': [
        {
          'AttachTime': 'string',
          'DevicePath': 'string',
          'NodeName': 'string',
          'Status': 'string'
        },
      ],
      'AvailabilityZone': 'string',
      'Cluster': {
        'Name': 'string'
      },
      'CreateTime': 'string',
      'Description': 'string',
      'Name': 'string',
      'Nrn': 'string',
      'Size': 123,
      'Status': 'string',
      'Tags': [
        {
          'Id': 'string',
          'Key': 'string',
          'Value': 'string'
        },
      ],
      'Type': 'string'
    },
  ]
}
```

### Response Structure

- (*dict*) –
  - **Disks** (*list*) –
    - \* (*dict*) –
      - **Attachments** (*list*) –
      - (*dict*) –



- **AttachTime** (*string*) –
- **DevicePath** (*string*) –
- **NodeName** (*string*) –
- **Status** (*string*) –
- **AvailabilityZone** (*string*) –
- **Cluster** (*dict*) –
- **Name** (*string*) –
- **CreateTime** (*string*) –
- **Description** (*string*) –
- **Name** (*string*) –
- **Nrn** (*string*) –
- **Size** (*integer*) –
- **Status** (*string*) –
- **Tags** (*list*) –
- (*dict*) –
- **Id** (*string*) –
- **Key** (*string*) –
- **Value** (*string*) –
- **Type** (*string*) –

*hatoba* / Client / delete\_firewall\_group

## delete\_firewall\_group

`hatoba.Client.delete_firewall_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_firewall_group(
    FirewallGroupName='string'
)
```

**Parameters** `FirewallGroupName` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
  'FirewallGroup': {
    'Description': 'string',
    'Name': 'string',
    'Nrn': 'string',
    'Rules': [
      {
        'CidrIp': 'string',
        'Description': 'string',
        'Direction': 'string',
        'FromPort': 123,
        'Id': 'string',
        'Protocol': 'string',
        'Status': 'string',
        'ToPort': 123
      },
    ],
  },
}
```

(continues on next page)

(continued from previous page)

```

        'Tags': [
            {
                'Id': 'string',
                'Key': 'string',
                'Value': 'string'
            },
        ]
    }
}

```

**Response Structure**

- *(dict)* –
  - **FirewallGroup** (*dict*) –
    - \* **Description** (*string*) –
    - \* **Name** (*string*) –
    - \* **Nrn** (*string*) –
    - \* **Rules** (*list*) –
      - (*dict*) –
      - **CidrIp** (*string*) –
      - **Description** (*string*) –
      - **Direction** (*string*) –
      - **FromPort** (*integer*) –
      - **Id** (*string*) –
      - **Protocol** (*string*) –
      - **Status** (*string*) –
      - **ToPort** (*integer*) –
    - \* **Tags** (*list*) –
      - (*dict*) –
      - **Id** (*string*) –
      - **Key** (*string*) –
      - **Value** (*string*) –

*hatoba* / Client / delete\_firewall\_groups**delete\_firewall\_groups**`hatoba.Client.delete_firewall_groups (**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```

response = client.delete_firewall_groups(
    Names='string'
)

```

**Parameters** **Names** (*string*) –**Return type** dict**Returns****Response Syntax**

```

{
    'FirewallGroups': [
        {

```

(continues on next page)

(continued from previous page)

```

        'Description': 'string',
        'Name': 'string',
        'Nrn': 'string',
        'Rules': [
            {
                'CidrIp': 'string',
                'Description': 'string',
                'Direction': 'string',
                'FromPort': 123,
                'Id': 'string',
                'Protocol': 'string',
                'Status': 'string',
                'ToPort': 123
            },
        ],
        'Tags': [
            {
                'Id': 'string',
                'Key': 'string',
                'Value': 'string'
            },
        ],
    },
]
}

```

**Response Structure**

- (dict) –
  - **FirewallGroups** (list) –
    - \* (dict) –
      - **Description** (string) –
      - **Name** (string) –
      - **Nrn** (string) –
      - **Rules** (list) –
      - (dict) –
      - **CidrIp** (string) –
      - **Description** (string) –
      - **Direction** (string) –
      - **FromPort** (integer) –
      - **Id** (string) –
      - **Protocol** (string) –
      - **Status** (string) –
      - **ToPort** (integer) –
      - **Tags** (list) –
      - (dict) –
      - **Id** (string) –
      - **Key** (string) –
      - **Value** (string) –

*hatoba* / Client / delete\_node\_pool

**delete\_node\_pool**

`hatoba.Client.delete_node_pool(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

## Request Syntax

```
response = client.delete_node_pool(  
    ClusterName='string',  
    NodePoolName='string'  
)
```

### Parameters

- **ClusterName** (*string*) – [REQUIRED]
- **NodePoolName** (*string*) – [REQUIRED]

**Return type** dict

### Returns

## Response Syntax

```
{  
    'NodePool': {  
        'InitialNodeCount': 123,  
        'InstanceType': 'string',  
        'Name': 'string',  
        'NodeCount': 123,  
        'Nodes': [  
            {  
                'AvailabilityZone': 'string',  
                'Name': 'string',  
                'PrivateIpAddress': 'string',  
                'PublicIpAddress': 'string',  
                'Status': 'string'  
            },  
        ],  
        'Nrn': 'string',  
        'Status': 'string',  
        'Tags': [  
            {  
                'Id': 'string',  
                'Key': 'string',  
                'Value': 'string'  
            },  
        ],  
    },  
}
```

## Response Structure

- (*dict*) –
  - **NodePool** (*dict*) –
    - \* **InitialNodeCount** (*integer*) –
    - \* **InstanceType** (*string*) –
    - \* **Name** (*string*) –
    - \* **NodeCount** (*integer*) –
    - \* **Nodes** (*list*) –
      - (*dict*) –
      - **AvailabilityZone** (*string*) –
      - **Name** (*string*) –
      - **PrivateIpAddress** (*string*) –
      - **PublicIpAddress** (*string*) –
      - **Status** (*string*) –
    - \* **Nrn** (*string*) –

- \* **Status** (*string*) –
- \* **Tags** (*list*) –
  - (*dict*) –
  - **Id** (*string*) –
  - **Key** (*string*) –
  - **Value** (*string*) –

*hatoba* / Client / delete\_node\_pools

## delete\_node\_pools

`hatoba.Client.delete_node_pools(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_node_pools(
    ClusterName='string',
    Names='string'
)
```

### Parameters

- **ClusterName** (*string*) – [REQUIRED]
- **Names** (*string*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```
{
  'NodePools': [
    {
      'InitialNodeCount': 123,
      'InstanceType': 'string',
      'Name': 'string',
      'NodeCount': 123,
      'Nodes': [
        {
          'AvailabilityZone': 'string',
          'Name': 'string',
          'PrivateIpAddress': 'string',
          'PublicIpAddress': 'string',
          'Status': 'string'
        },
      ],
      'Nrn': 'string',
      'Status': 'string',
      'Tags': [
        {
          'Id': 'string',
          'Key': 'string',
          'Value': 'string'
        },
      ],
    },
  ],
}
```

**Response Structure**

- *(dict)* –
  - **NodePools** (*list*) –
    - \* *(dict)* –
      - **InitialNodeCount** (*integer*) –
      - **InstanceType** (*string*) –
      - **Name** (*string*) –
      - **NodeCount** (*integer*) –
      - **Nodes** (*list*) –
      - *(dict)* –
      - **AvailabilityZone** (*string*) –
      - **Name** (*string*) –
      - **PrivateIpAddress** (*string*) –
      - **PublicIpAddress** (*string*) –
      - **Status** (*string*) –
      - **Nrn** (*string*) –
      - **Status** (*string*) –
      - **Tags** (*list*) –
      - *(dict)* –
      - **Id** (*string*) –
      - **Key** (*string*) –
      - **Value** (*string*) –

*hatoba* / Client / delete\_snapshot

**delete\_snapshot**

`hatoba.Client.delete_snapshot(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.delete_snapshot(  
    SnapshotName='string'  
)
```

**Parameters** **SnapshotName** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

**Response Syntax**

```
{  
    'Snapshot': {  
        'Cluster': {  
            'KubernetesVersion': 'string',  
            'Name': 'string',  
            'NodePools': [  
                {  
                    'InstanceType': 'string',  
                    'Name': 'string',  
                    'NodeCount': 123  
                },  
            ],  
        },  
        'CreateTime': 'string',  
    },  
}
```

(continues on next page)

(continued from previous page)

```

        'Description': 'string',
        'ExpirationTime': 'string',
        'Name': 'string',
        'Nrn': 'string',
        'ResourceVersion': 'string',
        'Status': 'string',
        'Tags': [
            {
                'Id': 'string',
                'Key': 'string',
                'Value': 'string'
            },
        ]
    }
}

```

**Response Structure**

- *(dict)* –
  - **Snapshot** (*dict*) –
    - \* **Cluster** (*dict*) –
      - **KubernetesVersion** (*string*) –
      - **Name** (*string*) –
      - **NodePools** (*list*) –
      - (*dict*) –
      - **InstanceType** (*string*) –
      - **Name** (*string*) –
      - **NodeCount** (*integer*) –
    - \* **CreateTime** (*string*) –
    - \* **Description** (*string*) –
    - \* **ExpirationTime** (*string*) –
    - \* **Name** (*string*) –
    - \* **Nrn** (*string*) –
    - \* **ResourceVersion** (*string*) –
    - \* **Status** (*string*) –
    - \* **Tags** (*list*) –
      - (*dict*) –
      - **Id** (*string*) –
      - **Key** (*string*) –
      - **Value** (*string*) –

*hatoba* / Client / delete\_snapshots**delete\_snapshots**`hatoba.Client.delete_snapshots (**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```

response = client.delete_snapshots(
    Names='string'
)

```

**Parameters** **Names** (*string*) –**Return type** dict

## Returns

### Response Syntax

```
{
  'Snapshots': [
    {
      'Cluster': {
        'KubernetesVersion': 'string',
        'Name': 'string',
        'NodePools': [
          {
            'InstanceType': 'string',
            'Name': 'string',
            'NodeCount': 123
          },
        ]
      },
      'CreateTime': 'string',
      'Description': 'string',
      'ExpirationTime': 'string',
      'Name': 'string',
      'Nrn': 'string',
      'ResourceVersion': 'string',
      'Status': 'string',
      'Tags': [
        {
          'Id': 'string',
          'Key': 'string',
          'Value': 'string'
        },
      ]
    },
  ]
}
```

### Response Structure

- *(dict)* –
  - **Snapshots** (*list*) –
    - \* *(dict)* –
      - **Cluster** (*dict*) –
      - **KubernetesVersion** (*string*) –
      - **Name** (*string*) –
      - **NodePools** (*list*) –
      - *(dict)* –
      - **InstanceType** (*string*) –
      - **Name** (*string*) –
      - **NodeCount** (*integer*) –
      - **CreateTime** (*string*) –
      - **Description** (*string*) –
      - **ExpirationTime** (*string*) –
      - **Name** (*string*) –
      - **Nrn** (*string*) –
      - **ResourceVersion** (*string*) –
      - **Status** (*string*) –
      - **Tags** (*list*) –
      - *(dict)* –



- **Id** (*string*) –
- **Key** (*string*) –
- **Value** (*string*) –

*hatoba* / Client / delete\_tags

## delete\_tags

`hatoba.Client.delete_tags(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_tags(
    Ids='string'
)
```

**Parameters** **Ids** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
    'Tags': [
        {
            'Id': 'string',
            'Key': 'string',
            'Nrn': 'string',
            'Value': 'string'
        },
    ]
}
```

### Response Structure

- (*dict*) –
  - **Tags** (*list*) –
    - \* (*dict*) –
      - **Id** (*string*) –
      - **Key** (*string*) –
      - **Nrn** (*string*) –
      - **Value** (*string*) –

*hatoba* / Client / detach\_disk

## detach\_disk

`hatoba.Client.detach_disk(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.detach_disk(
    DiskName='string',
    NodeName='string'
)
```

**Parameters**

- **DiskName** (*string*) – [REQUIRED]
- **NodeName** (*string*) – [REQUIRED]

**Return type** dict**Returns****Response Syntax**

```
{
  'Disk': {
    'Attachments': [
      {
        'AttachTime': 'string',
        'DevicePath': 'string',
        'NodeName': 'string',
        'Status': 'string'
      },
    ],
    'AvailabilityZone': 'string',
    'Cluster': {
      'Name': 'string'
    },
    'CreateTime': 'string',
    'Description': 'string',
    'Name': 'string',
    'Nrn': 'string',
    'Size': 123,
    'Status': 'string',
    'Tags': [
      {
        'Id': 'string',
        'Key': 'string',
        'Value': 'string'
      },
    ],
    'Type': 'string'
  }
}
```

**Response Structure**

- (*dict*) –
  - **Disk** (*dict*) –
    - \* **Attachments** (*list*) –
      - (*dict*) –
      - **AttachTime** (*string*) –
      - **DevicePath** (*string*) –
      - **NodeName** (*string*) –
      - **Status** (*string*) –
    - \* **AvailabilityZone** (*string*) –
    - \* **Cluster** (*dict*) –
      - **Name** (*string*) –
    - \* **CreateTime** (*string*) –
    - \* **Description** (*string*) –
    - \* **Name** (*string*) –
    - \* **Nrn** (*string*) –
    - \* **Size** (*integer*) –
    - \* **Status** (*string*) –

- \* **Tags** (*list*) –
  - (*dict*) –
  - **Id** (*string*) –
  - **Key** (*string*) –
  - **Value** (*string*) –
- \* **Type** (*string*) –

*hatoba* / Client / get\_cluster

## get\_cluster

`hatoba.Client.get_cluster(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.get_cluster(
    ClusterName='string'
)
```

**Parameters** `ClusterName` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
  'Cluster': {
    'AddonsConfig': {
      'HttpLoadBalancing': {
        'Disabled': True|False
      }
    },
    'CreateTime': 'string',
    'Description': 'string',
    'FirewallGroup': 'string',
    'InitialKubernetesVersion': 'string',
    'InitialNodeCount': 123,
    'KubernetesVersion': 'string',
    'Locations': [
      'string',
    ],
    'Name': 'string',
    'NetworkConfig': {
      'NetworkId': 'string'
    },
    'NodeCount': 123,
    'NodePools': [
      {
        'InitialNodeCount': 123,
        'InstanceType': 'string',
        'Name': 'string',
        'NodeCount': 123,
        'Nodes': [
          {
            'AvailabilityZone': 'string',
            'Name': 'string',
```

(continues on next page)

(continued from previous page)

```

        'PrivateIpAddress': 'string',
        'PublicIpAddress': 'string',
        'Status': 'string'
    },
    ],
    'Nrn': 'string',
    'Status': 'string',
    'Tags': [
        {
            'Id': 'string',
            'Key': 'string',
            'Value': 'string'
        },
    ],
    ],
    },
    ],
    'Nrn': 'string',
    'Status': 'string',
    'Tags': [
        {
            'Id': 'string',
            'Key': 'string',
            'Value': 'string'
        },
    ],
    ],
    }
}

```

**Response Structure**

- *(dict)* –
  - **Cluster** *(dict)* –
    - \* **AddonsConfig** *(dict)* –
      - **HttpLoadBalancing** *(dict)* –
      - **Disabled** *(boolean)* –
    - \* **CreateTime** *(string)* –
    - \* **Description** *(string)* –
    - \* **FirewallGroup** *(string)* –
    - \* **InitialKubernetesVersion** *(string)* –
    - \* **InitialNodeCount** *(integer)* –
    - \* **KubernetesVersion** *(string)* –
    - \* **Locations** *(list)* –
      - *(string)* –
    - \* **Name** *(string)* –
    - \* **NetworkConfig** *(dict)* –
      - **NetworkId** *(string)* –
    - \* **NodeCount** *(integer)* –
    - \* **NodePools** *(list)* –
      - *(dict)* –
      - **InitialNodeCount** *(integer)* –
      - **InstanceType** *(string)* –
      - **Name** *(string)* –
      - **NodeCount** *(integer)* –
      - **Nodes** *(list)* –
      - *(dict)* –
      - **AvailabilityZone** *(string)* –

- **Name** (*string*) –
- **PrivateIpAddress** (*string*) –
- **PublicIpAddress** (*string*) –
- **Status** (*string*) –
- **Nrn** (*string*) –
- **Status** (*string*) –
- **Tags** (*list*) –
- (*dict*) –
- **Id** (*string*) –
- **Key** (*string*) –
- **Value** (*string*) –
- \* **Nrn** (*string*) –
- \* **Status** (*string*) –
- \* **Tags** (*list*) –
- (*dict*) –
- **Id** (*string*) –
- **Key** (*string*) –
- **Value** (*string*) –

*hatoba* / Client / get\_cluster\_credentials

## get\_cluster\_credentials

`hatoba.Client.get_cluster_credentials(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.get_cluster_credentials(
    ClusterName='string'
)
```

**Parameters** **ClusterName** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
    'Credentials': 'string'
}
```

### Response Structure

- (*dict*) –
  - **Credentials** (*string*) –

*hatoba* / Client / get\_disk

## get\_disk

`hatoba.Client.get_disk(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.get_disk(  
    DiskName='string'  
)
```

**Parameters** **DiskName** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{  
    'Disk': {  
        'Attachments': [  
            {  
                'AttachTime': 'string',  
                'DevicePath': 'string',  
                'NodeName': 'string',  
                'Status': 'string'  
            },  
        ],  
        'AvailabilityZone': 'string',  
        'Cluster': {  
            'Name': 'string'  
        },  
        'CreateTime': 'string',  
        'Description': 'string',  
        'Name': 'string',  
        'Nrn': 'string',  
        'Size': 123,  
        'Status': 'string',  
        'Tags': [  
            {  
                'Id': 'string',  
                'Key': 'string',  
                'Value': 'string'  
            },  
        ],  
        'Type': 'string'  
    }  
}
```

#### Response Structure

- (*dict*) –
  - **Disk** (*dict*) –
    - \* **Attachments** (*list*) –
      - (*dict*) –
      - **AttachTime** (*string*) –
      - **DevicePath** (*string*) –
      - **NodeName** (*string*) –
      - **Status** (*string*) –
    - \* **AvailabilityZone** (*string*) –
    - \* **Cluster** (*dict*) –
      - **Name** (*string*) –
    - \* **CreateTime** (*string*) –
    - \* **Description** (*string*) –
    - \* **Name** (*string*) –
    - \* **Nrn** (*string*) –

- \* **Size** (*integer*) –
- \* **Status** (*string*) –
- \* **Tags** (*list*) –
  - (*dict*) –
  - **Id** (*string*) –
  - **Key** (*string*) –
  - **Value** (*string*) –
- \* **Type** (*string*) –

*hatoba* / Client / get\_firewall\_group

## get\_firewall\_group

`hatoba.Client.get_firewall_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.get_firewall_group(
    FirewallGroupName='string'
)
```

**Parameters** `FirewallGroupName` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
  'FirewallGroup': {
    'Description': 'string',
    'Name': 'string',
    'Nrn': 'string',
    'Rules': [
      {
        'CidrIp': 'string',
        'Description': 'string',
        'Direction': 'string',
        'FromPort': 123,
        'Id': 'string',
        'Protocol': 'string',
        'Status': 'string',
        'ToPort': 123
      },
    ],
    'Tags': [
      {
        'Id': 'string',
        'Key': 'string',
        'Value': 'string'
      },
    ],
  }
}
```

### Response Structure

- (*dict*) –
  - **FirewallGroup** (*dict*) –

- \* **Description** (*string*) –
- \* **Name** (*string*) –
- \* **Nrn** (*string*) –
- \* **Rules** (*list*) –
  - (*dict*) –
  - **CidrIp** (*string*) –
  - **Description** (*string*) –
  - **Direction** (*string*) –
  - **FromPort** (*integer*) –
  - **Id** (*string*) –
  - **Protocol** (*string*) –
  - **Status** (*string*) –
  - **ToPort** (*integer*) –
- \* **Tags** (*list*) –
  - (*dict*) –
  - **Id** (*string*) –
  - **Key** (*string*) –
  - **Value** (*string*) –

*hatoba* / Client / get\_load\_balancer

## get\_load\_balancer

`hatoba.Client.get_load_balancer(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.get_load_balancer(  
    InstancePort=123,  
    LoadBalancerName='string',  
    LoadBalancerPort=123  
)
```

### Parameters

- **InstancePort** (*integer*) – [REQUIRED]
- **LoadBalancerName** (*string*) – [REQUIRED]
- **LoadBalancerPort** (*integer*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```
{  
    'LoadBalancers': {  
        'AccountingType': 123,  
        'AvailabilityZones': [  
            'string',  
        ],  
        'Clusters': [  
            {  
                'Name': 'string',  
                'NodePools': [  
                    {  
                        'Name': 'string',  
                        'NodeCount': 123,  

```

(continues on next page)



(continued from previous page)

```

        'Nodes': [
            {
                'AvailabilityZone': 'string',
                'HealthCheckState': 'string',
                'Name': 'string',
                'PublicIpAddress': 'string'
            },
        ]
    },
]
},
'CreatedTime': 'string',
'Description': 'string',
'DnsName': 'string',
'Filter': {
    'FilterType': 'string',
    'IpAddresses': 'string'
},
'HealthCheck': {
    'HealthyThreshold': 123,
    'Interval': 123,
    'Target': 'string',
    'Timeout': 123,
    'UnhealthyThreshold': 123
},
'ListenerDescriptions': [
    {
        'Listener': {
            'BalancingType': 'string',
            'InstancePort': 'string',
            'LoadBalancerPort': 'string',
            'Protocol': 'string',
            'SslCertificateId': 'string'
        }
    },
],
'LoadBalancerName': 'string',
'NetworkVolume': 'string',
'NextMonthAccountingType': 123,
'Option': {
    'SessionStickinessPolicy': {
        'Enabled': True|False,
        'ExpirationPeriod': 123
    },
    'SorryPage': {
        'Enabled': True|False,
        'StatusCode': 123
    }
},
'PolicyType': 'string'
}

```

**Response Structure**

- (dict) –
  - LoadBalancers (dict) –

- \* **AccountingType** (*integer*) –
- \* **AvailabilityZones** (*list*) –
  - (*string*) –
- \* **Clusters** (*list*) –
  - (*dict*) –
  - **Name** (*string*) –
  - **NodePools** (*list*) –
  - (*dict*) –
  - **Name** (*string*) –
  - **NodeCount** (*integer*) –
  - **Nodes** (*list*) –
  - (*dict*) –
  - **AvailabilityZone** (*string*) –
  - **HealthCheckState** (*string*) –
  - **Name** (*string*) –
  - **PublicIpAddress** (*string*) –
- \* **CreatedTime** (*string*) –
- \* **Description** (*string*) –
- \* **DnsName** (*string*) –
- \* **Filter** (*dict*) –
  - **FilterType** (*string*) –
  - **IpAddresses** (*string*) –
- \* **HealthCheck** (*dict*) –
  - **HealthyThreshold** (*integer*) –
  - **Interval** (*integer*) –
  - **Target** (*string*) –
  - **Timeout** (*integer*) –
  - **UnhealthyThreshold** (*integer*) –
- \* **ListenerDescriptions** (*list*) –
  - (*dict*) –
  - **Listener** (*dict*) –
  - **BalancingType** (*string*) –
  - **InstancePort** (*string*) –
  - **LoadBalancerPort** (*string*) –
  - **Protocol** (*string*) –
  - **SslCertificateId** (*string*) –
- \* **LoadBalancerName** (*string*) –
- \* **NetworkVolume** (*string*) –
- \* **NextMonthAccountingType** (*integer*) –
- \* **Option** (*dict*) –
  - **SessionStickinessPolicy** (*dict*) –
  - **Enabled** (*boolean*) –
  - **ExpirationPeriod** (*integer*) –
  - **SorryPage** (*dict*) –
  - **Enabled** (*boolean*) –
  - **StatusCode** (*integer*) –
- \* **PolicyType** (*string*) –

*hatoba* / Client / `get_node_pool`

## `get_node_pool`

`hatoba.Client.get_node_pool(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

## Request Syntax

```
response = client.get_node_pool(
    ClusterName='string',
    NodePoolName='string'
)
```

### Parameters

- **ClusterName** (*string*) – [REQUIRED]
- **NodePoolName** (*string*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```
{
  'NodePool': {
    'InitialNodeCount': 123,
    'InstanceType': 'string',
    'Name': 'string',
    'NodeCount': 123,
    'Nodes': [
      {
        'AvailabilityZone': 'string',
        'Name': 'string',
        'PrivateIpAddress': 'string',
        'PublicIpAddress': 'string',
        'Status': 'string'
      },
    ],
    'Nrn': 'string',
    'Status': 'string',
    'Tags': [
      {
        'Id': 'string',
        'Key': 'string',
        'Value': 'string'
      },
    ],
  }
}
```

### Response Structure

- (*dict*) –
  - **NodePool** (*dict*) –
    - \* **InitialNodeCount** (*integer*) –
    - \* **InstanceType** (*string*) –
    - \* **Name** (*string*) –
    - \* **NodeCount** (*integer*) –
    - \* **Nodes** (*list*) –
      - (*dict*) –
        - **AvailabilityZone** (*string*) –
        - **Name** (*string*) –
        - **PrivateIpAddress** (*string*) –
        - **PublicIpAddress** (*string*) –
        - **Status** (*string*) –
    - \* **Nrn** (*string*) –

- \* **Status** (*string*) –
- \* **Tags** (*list*) –
  - (*dict*) –
  - **Id** (*string*) –
  - **Key** (*string*) –
  - **Value** (*string*) –

*hatoba* / Client / get\_paginator

## get\_paginator

`hatoba.Client.get_paginator(operation_name)`

Create a paginator for an operation.

**Parameters** `operation_name` (*string*) – The operation name. This is the same name as the method name on the client. For example, if the method name is `create_foo`, and you'd normally invoke the operation as `client.create_foo(**kwargs)`, if the `create_foo` operation can be paginated, you can use the call `client.get_paginator("create_foo")`.

**Raises** `OperationNotPageableError` – Raised if the operation is not pageable. You can use the `client.can_paginate` method to check if an operation is pageable.

**Return type** `L{botocore.paginate.Paginator}`

**Returns** A paginator object.

*hatoba* / Client / get\_server\_config

## get\_server\_config

`hatoba.Client.get_server_config()`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.get_server_config()
```

**Return type** `dict`

**Returns**

### Response Syntax

```
{
  'ServerConfig': {
    'DefaultKubernetesVersion': 'string',
    'ValidKubernetesVersions': [
      'string',
    ]
  }
}
```

### Response Structure

- (*dict*) –
  - **ServerConfig** (*dict*) –
    - \* **DefaultKubernetesVersion** (*string*) –
    - \* **ValidKubernetesVersions** (*list*) –
      - (*string*) –

*hatoba* / Client / get\_snapshot

## get\_snapshot

hatoba.Client.get\_snapshot(\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.get_snapshot(
    SnapshotName='string'
)
```

**Parameters** **SnapshotName** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
  'Snapshot': {
    'Cluster': {
      'KubernetesVersion': 'string',
      'Name': 'string',
      'NodePools': [
        {
          'InstanceType': 'string',
          'Name': 'string',
          'NodeCount': 123
        }
      ]
    },
    'CreateTime': 'string',
    'Description': 'string',
    'ExpirationTime': 'string',
    'Name': 'string',
    'Nrn': 'string',
    'ResourceVersion': 'string',
    'Status': 'string',
    'Tags': [
      {
        'Id': 'string',
        'Key': 'string',
        'Value': 'string'
      }
    ]
  }
}
```

### Response Structure

- (*dict*) –
  - **Snapshot** (*dict*) –
    - \* **Cluster** (*dict*) –
      - **KubernetesVersion** (*string*) –
      - **Name** (*string*) –
      - **NodePools** (*list*) –
      - (*dict*) –
      - **InstanceType** (*string*) –
      - **Name** (*string*) –
      - **NodeCount** (*integer*) –

- \* **CreateTime** (*string*) –
- \* **Description** (*string*) –
- \* **ExpirationTime** (*string*) –
- \* **Name** (*string*) –
- \* **Nrn** (*string*) –
- \* **ResourceVersion** (*string*) –
- \* **Status** (*string*) –
- \* **Tags** (*list*) –
  - (*dict*) –
  - **Id** (*string*) –
  - **Key** (*string*) –
  - **Value** (*string*) –

*hatoba* / Client / get\_waiter

## get\_waiter

`hatoba.Client.get_waiter(waiter_name)`

Returns an object that can wait for some condition.

**Parameters** **waiter\_name** (*str*) – The name of the waiter to get. See the waiters section of the service docs for a list of available waiters.

**Returns** The specified waiter object.

**Return type** `botocore.waiter.Waiter`

*hatoba* / Client / list\_clusters

## list\_clusters

`hatoba.Client.list_clusters(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.list_clusters(  
    Filters='string'  
)
```

**Parameters** **Filters** (*string*) –

**Return type** `dict`

**Returns**

### Response Syntax

```
{  
    'Clusters': [  
        {  
            'AddonsConfig': {  
                'HttpLoadBalancing': {  
                    'Disabled': True|False  
                }  
            },  
            'CreateTime': 'string',  
            'Description': 'string',  
            'FirewallGroup': 'string',  
            'InitialKubernetesVersion': 'string',
```

(continues on next page)

(continued from previous page)

```

        'InitialNodeCount': 123,
        'KubernetesVersion': 'string',
        'Locations': [
            'string',
        ],
        'Name': 'string',
        'NetworkConfig': {
            'NetworkId': 'string'
        },
        'NodeCount': 123,
        'NodePools': [
            {
                'InitialNodeCount': 123,
                'InstanceType': 'string',
                'Name': 'string',
                'NodeCount': 123,
                'Nodes': [
                    {
                        'AvailabilityZone': 'string',
                        'Name': 'string',
                        'PrivateIpAddress': 'string',
                        'PublicIpAddress': 'string',
                        'Status': 'string'
                    },
                ],
                'Nrn': 'string',
                'Status': 'string',
                'Tags': [
                    {
                        'Id': 'string',
                        'Key': 'string',
                        'Value': 'string'
                    },
                ],
            },
        ],
        'Nrn': 'string',
        'Status': 'string',
        'Tags': [
            {
                'Id': 'string',
                'Key': 'string',
                'Value': 'string'
            },
        ],
    },
]
}

```

**Response Structure**

- *(dict)* –
  - **Clusters** *(list)* –
    - \* *(dict)* –
      - **AddonsConfig** *(dict)* –
      - **HttpLoadBalancing** *(dict)* –
      - **Disabled** *(boolean)* –
      - **CreateTime** *(string)* –

- **Description** (*string*) –
- **FirewallGroup** (*string*) –
- **InitialKubernetesVersion** (*string*) –
- **InitialNodeCount** (*integer*) –
- **KubernetesVersion** (*string*) –
- **Locations** (*list*) –
- (*string*) –
- **Name** (*string*) –
- **NetworkConfig** (*dict*) –
- **NetworkId** (*string*) –
- **NodeCount** (*integer*) –
- **NodePools** (*list*) –
- (*dict*) –
- **InitialNodeCount** (*integer*) –
- **InstanceType** (*string*) –
- **Name** (*string*) –
- **NodeCount** (*integer*) –
- **Nodes** (*list*) –
- (*dict*) –
- **AvailabilityZone** (*string*) –
- **Name** (*string*) –
- **PrivateIpAddress** (*string*) –
- **PublicIpAddress** (*string*) –
- **Status** (*string*) –
- **Nrn** (*string*) –
- **Status** (*string*) –
- **Tags** (*list*) –
- (*dict*) –
- **Id** (*string*) –
- **Key** (*string*) –
- **Value** (*string*) –
- **Nrn** (*string*) –
- **Status** (*string*) –
- **Tags** (*list*) –
- (*dict*) –
- **Id** (*string*) –
- **Key** (*string*) –
- **Value** (*string*) –

*hatoba* / Client / list\_disks

## list\_disks

`hatoba.Client.list_disks(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.list_disks(  
    Filters='string'  
)
```

**Parameters** **Filters** (*string*)–

**Return type** dict

**Returns**



## Response Syntax

```
{
  'Disks': [
    {
      'Attachments': [
        {
          'AttachTime': 'string',
          'DevicePath': 'string',
          'NodeName': 'string',
          'Status': 'string'
        },
      ],
      'AvailabilityZone': 'string',
      'Cluster': {
        'Name': 'string'
      },
      'CreateTime': 'string',
      'Description': 'string',
      'Name': 'string',
      'Nrn': 'string',
      'Size': 123,
      'Status': 'string',
      'Tags': [
        {
          'Id': 'string',
          'Key': 'string',
          'Value': 'string'
        },
      ],
      'Type': 'string'
    },
  ],
}
```

## Response Structure

- (dict) –
  - **Disks** (list) –
    - \* (dict) –
      - **Attachments** (list) –
        - (dict) –
          - **AttachTime** (string) –
          - **DevicePath** (string) –
          - **NodeName** (string) –
          - **Status** (string) –
        - **AvailabilityZone** (string) –
        - **Cluster** (dict) –
          - **Name** (string) –
        - **CreateTime** (string) –
        - **Description** (string) –
        - **Name** (string) –
        - **Nrn** (string) –
        - **Size** (integer) –
        - **Status** (string) –
        - **Tags** (list) –
          - (dict) –
            - **Id** (string) –

- **Key** (*string*) –
- **Value** (*string*) –
- **Type** (*string*) –

*hatoba* / Client / list\_firewall\_groups

## list\_firewall\_groups

`hatoba.Client.list_firewall_groups(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.list_firewall_groups(  
    Filters='string'  
)
```

**Parameters** **Filters** (*string*) –

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'FirewallGroups': [  
        {  
            'Description': 'string',  
            'Name': 'string',  
            'Nrn': 'string',  
            'Rules': [  
                {  
                    'CidrIp': 'string',  
                    'Description': 'string',  
                    'Direction': 'string',  
                    'FromPort': 123,  
                    'Id': 'string',  
                    'Protocol': 'string',  
                    'Status': 'string',  
                    'ToPort': 123  
                },  
            ],  
            'Tags': [  
                {  
                    'Id': 'string',  
                    'Key': 'string',  
                    'Value': 'string'  
                },  
            ]  
        },  
    ]  
}
```

### Response Structure

- (*dict*) –
  - **FirewallGroups** (*list*) –
    - \* (*dict*) –
      - **Description** (*string*) –
      - **Name** (*string*) –

- **Nrn** (*string*) –
- **Rules** (*list*) –
- (*dict*) –
- **CidrIp** (*string*) –
- **Description** (*string*) –
- **Direction** (*string*) –
- **FromPort** (*integer*) –
- **Id** (*string*) –
- **Protocol** (*string*) –
- **Status** (*string*) –
- **ToPort** (*integer*) –
- **Tags** (*list*) –
- (*dict*) –
- **Id** (*string*) –
- **Key** (*string*) –
- **Value** (*string*) –

*hatoba* / Client / list\_load\_balancers

## list\_load\_balancers

`hatoba.Client.list_load_balancers()`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.list_load_balancers()
```

**Return type** dict

**Returns**

### Response Syntax

```
{
  'LoadBalancers': [
    {
      'AccountingType': 123,
      'AvailabilityZones': [
        'string',
      ],
      'Clusters': [
        {
          'Name': 'string',
          'NodePools': [
            {
              'Name': 'string',
              'NodeCount': 123,
              'Nodes': [
                {
                  'AvailabilityZone': 'string',
                  'HealthCheckState': 'string',
                  'Name': 'string',
                  'PublicIpAddress': 'string'
                },
              ],
            },
          ],
        },
      ],
    },
  ],
}
```

(continues on next page)

(continued from previous page)

```

        ],
        'CreatedTime': 'string',
        'Description': 'string',
        'DnsName': 'string',
        'Filter': {
            'FilterType': 'string',
            'IpAddresses': 'string'
        },
        'HealthCheck': {
            'HealthyThreshold': 123,
            'Interval': 123,
            'Target': 'string',
            'Timeout': 123,
            'UnhealthyThreshold': 123
        },
        'ListenerDescriptions': [
            {
                'Listener': {
                    'BalancingType': 'string',
                    'InstancePort': 'string',
                    'LoadBalancerPort': 'string',
                    'Protocol': 'string',
                    'SslCertificateId': 'string'
                }
            }
        ],
        'LoadBalancerName': 'string',
        'NetworkVolume': 'string',
        'NextMonthAccountingType': 123,
        'Option': {
            'SessionStickinessPolicy': {
                'Enabled': True|False,
                'ExpirationPeriod': 123
            },
            'SorryPage': {
                'Enabled': True|False,
                'StatusCode': 123
            }
        },
        'PolicyType': 'string'
    },
]
}

```

**Response Structure**

- (dict) –
  - **LoadBalancers** (list) –
    - \* (dict) –
      - **AccountingType** (integer) –
      - **AvailabilityZones** (list) –
      - (string) –
      - **Clusters** (list) –
      - (dict) –
      - **Name** (string) –

- **NodePools** (*list*) –
- (*dict*) –
- **Name** (*string*) –
- **NodeCount** (*integer*) –
- **Nodes** (*list*) –
- (*dict*) –
- **AvailabilityZone** (*string*) –
- **HealthCheckState** (*string*) –
- **Name** (*string*) –
- **PublicIpAddress** (*string*) –
- **CreatedTime** (*string*) –
- **Description** (*string*) –
- **DnsName** (*string*) –
- **Filter** (*dict*) –
- **FilterType** (*string*) –
- **IpAddresses** (*string*) –
- **HealthCheck** (*dict*) –
- **HealthyThreshold** (*integer*) –
- **Interval** (*integer*) –
- **Target** (*string*) –
- **Timeout** (*integer*) –
- **UnhealthyThreshold** (*integer*) –
- **ListenerDescriptions** (*list*) –
- (*dict*) –
- **Listener** (*dict*) –
- **BalancingType** (*string*) –
- **InstancePort** (*string*) –
- **LoadBalancerPort** (*string*) –
- **Protocol** (*string*) –
- **SslCertificateId** (*string*) –
- **LoadBalancerName** (*string*) –
- **NetworkVolume** (*string*) –
- **NextMonthAccountingType** (*integer*) –
- **Option** (*dict*) –
- **SessionStickinessPolicy** (*dict*) –
- **Enabled** (*boolean*) –
- **ExpirationPeriod** (*integer*) –
- **SorryPage** (*dict*) –
- **Enabled** (*boolean*) –
- **StatusCode** (*integer*) –
- **PolicyType** (*string*) –

*hatoba* / Client / list\_node\_pools

## list\_node\_pools

`hatoba.Client.list_node_pools(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.list_node_pools(
    ClusterName='string',
```

(continues on next page)

(continued from previous page)

```
Filters='string'
)
```

**Parameters**

- **ClusterName** (*string*) – [REQUIRED]
- **Filters** (*string*) –

**Return type** dict**Returns****Response Syntax**

```
{
  'NodePools': [
    {
      'InitialNodeCount': 123,
      'InstanceType': 'string',
      'Name': 'string',
      'NodeCount': 123,
      'Nodes': [
        {
          'AvailabilityZone': 'string',
          'Name': 'string',
          'PrivateIpAddress': 'string',
          'PublicIpAddress': 'string',
          'Status': 'string'
        },
      ],
      'Nrn': 'string',
      'Status': 'string',
      'Tags': [
        {
          'Id': 'string',
          'Key': 'string',
          'Value': 'string'
        },
      ],
    },
  ],
}
```

**Response Structure**

- (*dict*) –
  - **NodePools** (*list*) –
    - \* (*dict*) –
      - **InitialNodeCount** (*integer*) –
      - **InstanceType** (*string*) –
      - **Name** (*string*) –
      - **NodeCount** (*integer*) –
      - **Nodes** (*list*) –
        - (*dict*) –
          - **AvailabilityZone** (*string*) –
          - **Name** (*string*) –
          - **PrivateIpAddress** (*string*) –
          - **PublicIpAddress** (*string*) –
          - **Status** (*string*) –
          - **Nrn** (*string*) –

- **Status** (*string*) –
- **Tags** (*list*) –
- (*dict*) –
- **Id** (*string*) –
- **Key** (*string*) –
- **Value** (*string*) –

*hatoba* / Client / list\_snapshots

## list\_snapshots

`hatoba.Client.list_snapshots(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.list_snapshots(
    Filters='string'
)
```

**Parameters** **Filters** (*string*) –

**Return type** dict

**Returns**

### Response Syntax

```
{
  'Snapshots': [
    {
      'Cluster': {
        'KubernetesVersion': 'string',
        'Name': 'string',
        'NodePools': [
          {
            'InstanceType': 'string',
            'Name': 'string',
            'NodeCount': 123
          },
        ],
      },
      'CreateTime': 'string',
      'Description': 'string',
      'ExpirationTime': 'string',
      'Name': 'string',
      'Nrn': 'string',
      'ResourceVersion': 'string',
      'Status': 'string',
      'Tags': [
        {
          'Id': 'string',
          'Key': 'string',
          'Value': 'string'
        },
      ],
    },
  ],
}
```

**Response Structure**

- *(dict)* –
  - **Snapshots** (*list*) –
    - \* *(dict)* –
      - **Cluster** (*dict*) –
      - **KubernetesVersion** (*string*) –
      - **Name** (*string*) –
      - **NodePools** (*list*) –
      - *(dict)* –
      - **InstanceType** (*string*) –
      - **Name** (*string*) –
      - **NodeCount** (*integer*) –
      - **CreateTime** (*string*) –
      - **Description** (*string*) –
      - **ExpirationTime** (*string*) –
      - **Name** (*string*) –
      - **Nrn** (*string*) –
      - **ResourceVersion** (*string*) –
      - **Status** (*string*) –
      - **Tags** (*list*) –
      - *(dict)* –
      - **Id** (*string*) –
      - **Key** (*string*) –
      - **Value** (*string*) –

*hatoba* / Client / list\_tags

**list\_tags**

`hatoba.Client.list_tags(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.list_tags(  
    Nrn='string'  
)
```

**Parameters** **Nrn** (*string*) –

**Return type** dict

**Returns**

**Response Syntax**

```
{  
    'Tags': [  
        {  
            'Id': 'string',  
            'Key': 'string',  
            'Nrn': 'string',  
            'Value': 'string'  
        },  
    ]  
}
```

**Response Structure**



- (dict) –
  - **Tags** (list) –
    - \* (dict) –
      - **Id** (string) –
      - **Key** (string) –
      - **Nrn** (string) –
      - **Value** (string) –

*hatoba* / Client / reboot\_node

## reboot\_node

`hatoba.Client.reboot_node(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.reboot_node(
    ClusterName='string',
    Force=True|False,
    NodeName='string',
    NodePoolName='string'
)
```

### Parameters

- **ClusterName** (string) – [REQUIRED]
- **Force** (boolean) –
- **NodeName** (string) – [REQUIRED]
- **NodePoolName** (string) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```
{
  'Node': {
    'AvailabilityZone': 'string',
    'Name': 'string',
    'PrivateIpAddress': 'string',
    'PublicIpAddress': 'string',
    'Status': 'string'
  }
}
```

### Response Structure

- (dict) –
  - **Node** (dict) –
    - \* **AvailabilityZone** (string) –
    - \* **Name** (string) –
    - \* **PrivateIpAddress** (string) –
    - \* **PublicIpAddress** (string) –
    - \* **Status** (string) –

*hatoba* / Client / restore\_cluster\_from\_snapshot

## restore\_cluster\_from\_snapshot

hatoba.Client.restore\_cluster\_from\_snapshot (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.restore_cluster_from_snapshot(
    Cluster={
        'Description': 'string',
        'FirewallGroup': 'string',
        'ListOfRequestLocations': [
            'string',
        ],
        'ListOfRequestTags': [
            {
                'Key': 'string',
                'Value': 'string'
            },
        ],
        'Name': 'string',
        'RequestAddonsConfig': {
            'RequestHttpLoadBalancing': {
                'Disabled': True|False
            }
        },
        'RequestNetworkConfig': {
            'NetworkId': 'string'
        }
    },
    SnapshotName='string'
)
```

### Parameters

- **Cluster** (*dict*) – [REQUIRED]
  - **Description** (*string*) –
  - **FirewallGroup** (*string*) – [REQUIRED]
  - **ListOfRequestLocations** (*list*) – [REQUIRED]
    - \* (*string*) –
  - **ListOfRequestTags** (*list*) –
    - \* (*dict*) –
      - **Key** (*string*) –
      - **Value** (*string*) –
  - **Name** (*string*) – [REQUIRED]
  - **RequestAddonsConfig** (*dict*) –
    - \* **RequestHttpLoadBalancing** (*dict*) –
      - **Disabled** (*boolean*) –
  - **RequestNetworkConfig** (*dict*) –
    - \* **NetworkId** (*string*) –
- **SnapshotName** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
    'Cluster': {
```

(continues on next page)

(continued from previous page)

```

    'AddonsConfig': {
        'HttpLoadBalancing': {
            'Disabled': True|False
        }
    },
    'CreateTime': 'string',
    'Description': 'string',
    'FirewallGroup': 'string',
    'InitialKubernetesVersion': 'string',
    'InitialNodeCount': 123,
    'KubernetesVersion': 'string',
    'Locations': [
        'string',
    ],
    'Name': 'string',
    'NetworkConfig': {
        'NetworkId': 'string'
    },
    'NodeCount': 123,
    'NodePools': [
        {
            'InitialNodeCount': 123,
            'InstanceType': 'string',
            'Name': 'string',
            'NodeCount': 123,
            'Nodes': [
                {
                    'AvailabilityZone': 'string',
                    'Name': 'string',
                    'PrivateIpAddress': 'string',
                    'PublicIpAddress': 'string',
                    'Status': 'string'
                }
            ],
            'Nrn': 'string',
            'Status': 'string',
            'Tags': [
                {
                    'Id': 'string',
                    'Key': 'string',
                    'Value': 'string'
                }
            ]
        }
    ],
    'Nrn': 'string',
    'Status': 'string',
    'Tags': [
        {
            'Id': 'string',
            'Key': 'string',
            'Value': 'string'
        }
    ]
}

```

**Response Structure**

- *(dict)* –
  - **Cluster** *(dict)* –
    - \* **AddonsConfig** *(dict)* –
      - **HttpLoadBalancing** *(dict)* –
      - **Disabled** *(boolean)* –
    - \* **CreateTime** *(string)* –
    - \* **Description** *(string)* –
    - \* **FirewallGroup** *(string)* –
    - \* **InitialKubernetesVersion** *(string)* –
    - \* **InitialNodeCount** *(integer)* –
    - \* **KubernetesVersion** *(string)* –
    - \* **Locations** *(list)* –
      - *(string)* –
    - \* **Name** *(string)* –
    - \* **NetworkConfig** *(dict)* –
      - **NetworkId** *(string)* –
    - \* **NodeCount** *(integer)* –
    - \* **NodePools** *(list)* –
      - *(dict)* –
      - **InitialNodeCount** *(integer)* –
      - **InstanceType** *(string)* –
      - **Name** *(string)* –
      - **NodeCount** *(integer)* –
      - **Nodes** *(list)* –
      - *(dict)* –
      - **AvailabilityZone** *(string)* –
      - **Name** *(string)* –
      - **PrivateIpAddress** *(string)* –
      - **PublicIpAddress** *(string)* –
      - **Status** *(string)* –
      - **Nrn** *(string)* –
      - **Status** *(string)* –
      - **Tags** *(list)* –
      - *(dict)* –
      - **Id** *(string)* –
      - **Key** *(string)* –
      - **Value** *(string)* –
    - \* **Nrn** *(string)* –
    - \* **Status** *(string)* –
    - \* **Tags** *(list)* –
      - *(dict)* –
      - **Id** *(string)* –
      - **Key** *(string)* –
      - **Value** *(string)* –

*hatoba* / Client / revoke\_firewall\_group

**revoke\_firewall\_group**

`hatoba.Client.revoke_firewall_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.revoke_firewall_group(
    FirewallGroupName='string',
    Ids='string'
)
```

**Parameters**

- **FirewallGroupName** (*string*) – [REQUIRED]
- **Ids** (*string*) –

**Return type** dict**Returns****Response Syntax**

```
{
  'FirewallGroup': {
    'Description': 'string',
    'Name': 'string',
    'Nrn': 'string',
    'Rules': [
      {
        'CidrIp': 'string',
        'Description': 'string',
        'Direction': 'string',
        'FromPort': 123,
        'Id': 'string',
        'Protocol': 'string',
        'Status': 'string',
        'ToPort': 123
      },
    ],
    'Tags': [
      {
        'Id': 'string',
        'Key': 'string',
        'Value': 'string'
      },
    ],
  }
}
```

**Response Structure**

- (*dict*) –
  - **FirewallGroup** (*dict*) –
    - \* **Description** (*string*) –
    - \* **Name** (*string*) –
    - \* **Nrn** (*string*) –
    - \* **Rules** (*list*) –
      - (*dict*) –
      - **CidrIp** (*string*) –
      - **Description** (*string*) –
      - **Direction** (*string*) –
      - **FromPort** (*integer*) –
      - **Id** (*string*) –
      - **Protocol** (*string*) –
      - **Status** (*string*) –
      - **ToPort** (*integer*) –
    - \* **Tags** (*list*) –

- *(dict)* –
- **Id** (*string*) –
- **Key** (*string*) –
- **Value** (*string*) –

*hatoba* / Client / `set_node_pool_size`

## `set_node_pool_size`

`hatoba.Client.set_node_pool_size(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.set_node_pool_size(
    ClusterName='string',
    NodeCount=123,
    NodePoolName='string'
)
```

### Parameters

- **ClusterName** (*string*) – [REQUIRED]
- **NodeCount** (*integer*) – [REQUIRED]
- **NodePoolName** (*string*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```
{
  'NodePool': {
    'InitialNodeCount': 123,
    'InstanceType': 'string',
    'Name': 'string',
    'NodeCount': 123,
    'Nodes': [
      {
        'AvailabilityZone': 'string',
        'Name': 'string',
        'PrivateIpAddress': 'string',
        'PublicIpAddress': 'string',
        'Status': 'string'
      },
    ],
    'Nrn': 'string',
    'Status': 'string',
    'Tags': [
      {
        'Id': 'string',
        'Key': 'string',
        'Value': 'string'
      },
    ],
  }
}
```

### Response Structure

- *(dict)* –

- **NodePool** (*dict*) –
  - \* **InitialNodeCount** (*integer*) –
  - \* **InstanceType** (*string*) –
  - \* **Name** (*string*) –
  - \* **NodeCount** (*integer*) –
  - \* **Nodes** (*list*) –
    - (*dict*) –
    - **AvailabilityZone** (*string*) –
    - **Name** (*string*) –
    - **PrivateIpAddress** (*string*) –
    - **PublicIpAddress** (*string*) –
    - **Status** (*string*) –
  - \* **Nrn** (*string*) –
  - \* **Status** (*string*) –
  - \* **Tags** (*list*) –
    - (*dict*) –
    - **Id** (*string*) –
    - **Key** (*string*) –
    - **Value** (*string*) –

*hatoba* / Client / update\_cluster

## update\_cluster

`hatoba.Client.update_cluster(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.update_cluster(
    Cluster={
        'Description': 'string',
        'KubernetesVersion': 'v1.23.3'|'v1.23.9'|'v1.24.3',
        'ListOfRequestTags': [
            {
                'Key': 'string',
                'Value': 'string'
            },
        ],
        'Name': 'string',
        'RequestAddonsConfig': {
            'RequestHttpLoadBalancing': {
                'Disabled': True|False
            }
        }
    },
    ClusterName='string'
)
```

### Parameters

- **Cluster** (*dict*) –
  - **Description** (*string*) –
  - **KubernetesVersion** (*string*) –
  - **ListOfRequestTags** (*list*) –
    - \* (*dict*) –
      - **Key** (*string*) –

- **Value** (*string*) –
- **Name** (*string*) –
- **RequestAddonsConfig** (*dict*) –
  - \* **RequestHttpLoadBalancing** (*dict*) –
    - **Disabled** (*boolean*) –
- **ClusterName** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'Cluster': {
    'AddonsConfig': {
      'HttpLoadBalancing': {
        'Disabled': True|False
      }
    },
    'CreateTime': 'string',
    'Description': 'string',
    'FirewallGroup': 'string',
    'InitialKubernetesVersion': 'string',
    'InitialNodeCount': 123,
    'KubernetesVersion': 'string',
    'Locations': [
      'string',
    ],
    'Name': 'string',
    'NetworkConfig': {
      'NetworkId': 'string'
    },
    'NodeCount': 123,
    'NodePools': [
      {
        'InitialNodeCount': 123,
        'InstanceType': 'string',
        'Name': 'string',
        'NodeCount': 123,
        'Nodes': [
          {
            'AvailabilityZone': 'string',
            'Name': 'string',
            'PrivateIpAddress': 'string',
            'PublicIpAddress': 'string',
            'Status': 'string'
          },
        ],
        'Nrn': 'string',
        'Status': 'string',
        'Tags': [
          {
            'Id': 'string',
            'Key': 'string',
            'Value': 'string'
          },
        ],
      },
    ],
  },
}
```

(continues on next page)



(continued from previous page)

```

        'Nrn': 'string',
        'Status': 'string',
        'Tags': [
            {
                'Id': 'string',
                'Key': 'string',
                'Value': 'string'
            },
        ]
    }
}

```

**Response Structure**

- *(dict)* –
  - **Cluster** (*dict*) –
    - \* **AddonsConfig** (*dict*) –
      - **HttpLoadBalancing** (*dict*) –
      - **Disabled** (*boolean*) –
    - \* **CreateTime** (*string*) –
    - \* **Description** (*string*) –
    - \* **FirewallGroup** (*string*) –
    - \* **InitialKubernetesVersion** (*string*) –
    - \* **InitialNodeCount** (*integer*) –
    - \* **KubernetesVersion** (*string*) –
    - \* **Locations** (*list*) –
      - (*string*) –
    - \* **Name** (*string*) –
    - \* **NetworkConfig** (*dict*) –
      - **NetworkId** (*string*) –
    - \* **NodeCount** (*integer*) –
    - \* **NodePools** (*list*) –
      - (*dict*) –
      - **InitialNodeCount** (*integer*) –
      - **InstanceType** (*string*) –
      - **Name** (*string*) –
      - **NodeCount** (*integer*) –
      - **Nodes** (*list*) –
      - (*dict*) –
      - **AvailabilityZone** (*string*) –
      - **Name** (*string*) –
      - **PrivateIpAddress** (*string*) –
      - **PublicIpAddress** (*string*) –
      - **Status** (*string*) –
      - **Nrn** (*string*) –
      - **Status** (*string*) –
      - **Tags** (*list*) –
      - (*dict*) –
      - **Id** (*string*) –
      - **Key** (*string*) –
      - **Value** (*string*) –
    - \* **Nrn** (*string*) –
    - \* **Status** (*string*) –
    - \* **Tags** (*list*) –
      - (*dict*) –

- **Id** (*string*) –
- **Key** (*string*) –
- **Value** (*string*) –

*hatoba* / Client / update\_disk

## update\_disk

`hatoba.Client.update_disk(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.update_disk(  
    Disk={  
        'Description': 'string',  
        'ListOfRequestTags': [  
            {  
                'Key': 'string',  
                'Value': 'string'  
            },  
        ],  
        'Name': 'string',  
        'Size': 123  
    },  
    DiskName='string'  
)
```

### Parameters

- **Disk** (*dict*) –
  - **Description** (*string*) –
  - **ListOfRequestTags** (*list*) –
    - \* (*dict*) –
      - **Key** (*string*) –
      - **Value** (*string*) –
  - **Name** (*string*) –
  - **Size** (*integer*) –
- **DiskName** (*string*) – **[REQUIRED]**

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'Disk': {  
        'Attachments': [  
            {  
                'AttachTime': 'string',  
                'DevicePath': 'string',  
                'NodeName': 'string',  
                'Status': 'string'  
            },  
        ],  
        'AvailabilityZone': 'string',  
        'Cluster': {  
            'Name': 'string'  
        },  
    },  
}
```

(continues on next page)

(continued from previous page)

```

        'CreateTime': 'string',
        'Description': 'string',
        'Name': 'string',
        'Nrn': 'string',
        'Size': 123,
        'Status': 'string',
        'Tags': [
            {
                'Id': 'string',
                'Key': 'string',
                'Value': 'string'
            },
        ],
        'Type': 'string'
    }
}

```

**Response Structure**

- *(dict)* –
  - **Disk** *(dict)* –
    - \* **Attachments** *(list)* –
      - *(dict)* –
      - **AttachTime** *(string)* –
      - **DevicePath** *(string)* –
      - **NodeName** *(string)* –
      - **Status** *(string)* –
    - \* **AvailabilityZone** *(string)* –
    - \* **Cluster** *(dict)* –
      - **Name** *(string)* –
    - \* **CreateTime** *(string)* –
    - \* **Description** *(string)* –
    - \* **Name** *(string)* –
    - \* **Nrn** *(string)* –
    - \* **Size** *(integer)* –
    - \* **Status** *(string)* –
    - \* **Tags** *(list)* –
      - *(dict)* –
      - **Id** *(string)* –
      - **Key** *(string)* –
      - **Value** *(string)* –
    - \* **Type** *(string)* –

*hatoba* / Client / update\_firewall\_group

**update\_firewall\_group**

`hatoba.Client.update_firewall_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```

response = client.update_firewall_group(
    FirewallGroup={
        'Description': 'string',

```

(continues on next page)

(continued from previous page)

```

        'ListOfRequestTags': [
            {
                'Key': 'string',
                'Value': 'string'
            },
        ],
        'Name': 'string'
    },
    FirewallGroupName='string'
)

```

**Parameters**

- **FirewallGroup** (*dict*) –
  - **Description** (*string*) –
  - **ListOfRequestTags** (*list*) –
    - \* (*dict*) –
      - **Key** (*string*) –
      - **Value** (*string*) –
  - **Name** (*string*) –
- **FirewallGroupName** (*string*) – [REQUIRED]

**Return type** dict**Returns****Response Syntax**

```

{
    'FirewallGroup': {
        'Description': 'string',
        'Name': 'string',
        'Nrn': 'string',
        'Rules': [
            {
                'CidrIp': 'string',
                'Description': 'string',
                'Direction': 'string',
                'FromPort': 123,
                'Id': 'string',
                'Protocol': 'string',
                'Status': 'string',
                'ToPort': 123
            },
        ],
        'Tags': [
            {
                'Id': 'string',
                'Key': 'string',
                'Value': 'string'
            },
        ],
    }
}

```

**Response Structure**

- (*dict*) –
  - **FirewallGroup** (*dict*) –
    - \* **Description** (*string*) –

- \* **Name** (*string*) –
- \* **Nrn** (*string*) –
- \* **Rules** (*list*) –
  - (*dict*) –
  - **CidrIp** (*string*) –
  - **Description** (*string*) –
  - **Direction** (*string*) –
  - **FromPort** (*integer*) –
  - **Id** (*string*) –
  - **Protocol** (*string*) –
  - **Status** (*string*) –
  - **ToPort** (*integer*) –
- \* **Tags** (*list*) –
  - (*dict*) –
  - **Id** (*string*) –
  - **Key** (*string*) –
  - **Value** (*string*) –

*hatoba* / Client / update\_node\_pool

## update\_node\_pool

`hatoba.Client.update_node_pool (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.update_node_pool(
    ClusterName='string',
    NodePool={
        'ListOfRequestTags': [
            {
                'Key': 'string',
                'Value': 'string'
            },
        ]
    },
    NodePoolName='string'
)
```

### Parameters

- **ClusterName** (*string*) – [REQUIRED]
- **NodePool** (*dict*) –
  - **ListOfRequestTags** (*list*) –
    - \* (*dict*) –
      - **Key** (*string*) –
      - **Value** (*string*) –
- **NodePoolName** (*string*) – [REQUIRED]

Return type `dict`

### Returns

#### Response Syntax

```
{
    'NodePool': {
        'InitialNodeCount': 123,
```

(continues on next page)

(continued from previous page)

```
'InstanceType': 'string',
'Name': 'string',
'NodeCount': 123,
'Nodes': [
    {
        'AvailabilityZone': 'string',
        'Name': 'string',
        'PrivateIpAddress': 'string',
        'PublicIpAddress': 'string',
        'Status': 'string'
    },
],
'Nrn': 'string',
'Status': 'string',
'Tags': [
    {
        'Id': 'string',
        'Key': 'string',
        'Value': 'string'
    },
]
}
```

### Response Structure

- (dict) –
  - **NodePool** (dict) –
    - \* **InitialNodeCount** (integer) –
    - \* **InstanceType** (string) –
    - \* **Name** (string) –
    - \* **NodeCount** (integer) –
    - \* **Nodes** (list) –
      - (dict) –
      - **AvailabilityZone** (string) –
      - **Name** (string) –
      - **PrivateIpAddress** (string) –
      - **PublicIpAddress** (string) –
      - **Status** (string) –
    - \* **Nrn** (string) –
    - \* **Status** (string) –
    - \* **Tags** (list) –
      - (dict) –
      - **Id** (string) –
      - **Key** (string) –
      - **Value** (string) –

*hatoba* / Client / update\_snapshot

### update\_snapshot

`hatoba.Client.update_snapshot(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```

response = client.update_snapshot(
    Snapshot={
        'Description': 'string',
        'ExpirationTime': 'string',
        'ListOfRequestTags': [
            {
                'Key': 'string',
                'Value': 'string'
            },
        ],
        'Name': 'string'
    },
    SnapshotName='string'
)

```

### Parameters

- **Snapshot** (*dict*) –
  - **Description** (*string*) –
  - **ExpirationTime** (*string*) –
  - **ListOfRequestTags** (*list*) –
    - \* (*dict*) –
      - **Key** (*string*) –
      - **Value** (*string*) –
  - **Name** (*string*) –
- **SnapshotName** (*string*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```

{
  'Snapshot': {
    'Cluster': {
      'KubernetesVersion': 'string',
      'Name': 'string',
      'NodePools': [
        {
          'InstanceType': 'string',
          'Name': 'string',
          'NodeCount': 123
        },
      ],
    },
    'CreateTime': 'string',
    'Description': 'string',
    'ExpirationTime': 'string',
    'Name': 'string',
    'Nrn': 'string',
    'ResourceVersion': 'string',
    'Status': 'string',
    'Tags': [
      {
        'Id': 'string',
        'Key': 'string',
        'Value': 'string'
      },
    ],
  }
}

```

(continues on next page)

(continued from previous page)

```
}  
}
```

**Response Structure**

- *(dict)* –
  - **Snapshot** *(dict)* –
    - \* **Cluster** *(dict)* –
      - **KubernetesVersion** *(string)* –
      - **Name** *(string)* –
      - **NodePools** *(list)* –
      - *(dict)* –
      - **InstanceType** *(string)* –
      - **Name** *(string)* –
      - **NodeCount** *(integer)* –
    - \* **CreateTime** *(string)* –
    - \* **Description** *(string)* –
    - \* **ExpirationTime** *(string)* –
    - \* **Name** *(string)* –
    - \* **Nrn** *(string)* –
    - \* **ResourceVersion** *(string)* –
    - \* **Status** *(string)* –
    - \* **Tags** *(list)* –
      - *(dict)* –
      - **Id** *(string)* –
      - **Key** *(string)* –
      - **Value** *(string)* –

*hatoba* / Client / update\_tags**update\_tags**`hatoba.Client.update_tags (**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.update_tags(  
    Tags=[  
        {  
            'Key': 'string',  
            'Nrn': 'string',  
            'Value': 'string'  
        },  
    ],  
)
```

**Parameters** **Tags** *(list)* – [REQUIRED]

- *(dict)* –
  - **Key** *(string)* – [REQUIRED]
  - **Nrn** *(string)* – [REQUIRED]
  - **Value** *(string)* – [REQUIRED]

**Return type** dict**Returns****Response Syntax**



```
{
  'Tags': [
    {
      'Id': 'string',
      'Key': 'string',
      'Nrn': 'string',
      'Value': 'string'
    },
  ],
}
```

#### Response Structure

- *(dict)* –
  - **Tags** (*list*) –
    - \* *(dict)* –
      - **Id** (*string*) –
      - **Key** (*string*) –
      - **Nrn** (*string*) –
      - **Value** (*string*) –

## 1.4.2 Client Exceptions

Client exceptions are available on a client instance via the `exceptions` property. For more detailed instructions and examples on the exact usage of client exceptions, see the error handling [user guide](#).

This client has no modeled exception classes.

## 1.4.3 Waiters

Waiters are available on a client instance via the `get_waiter` method. For more detailed instructions and examples on the usage or waiters, see the waiters [user guide](#).

The available waiters are:

[hatoba](#) / Waiter / ClusterDeleted

### ClusterDeleted

```
class hatoba.Waiter.ClusterDeleted
```

```
waiter = client.get_waiter('cluster_deleted')
```

**wait** (*\*\*kwargs*)

Polls `hatoba.Client.get_cluster()` every 30 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

#### Request Syntax

```
waiter.wait(
    ClusterName='string',
    WaiterConfig={
```

(continues on next page)

(continued from previous page)

```

        'Delay': 123,
        'MaxAttempts': 123
    }
)

```

**Parameters**

- **ClusterName** (*string*) – **[REQUIRED]**
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 30
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None*hatoba* / Waiter / ClusterRunning**ClusterRunning****class** *hatoba.Waiter.ClusterRunning*

```
waiter = client.get_waiter('cluster_running')
```

**wait** (*\*\*kwargs*)

Polls *hatoba.Client.get\_cluster()* every 60 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```

waiter.wait(
    ClusterName='string',
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)

```

**Parameters**

- **ClusterName** (*string*) – **[REQUIRED]**
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 60
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None*hatoba* / Waiter / FirewallRuleAuthorized

## FirewallRuleAuthorized

**class** hatoba.Waiter.**FirewallRuleAuthorized**

```
waiter = client.get_waiter('firewall_rule_authorized')
```

**wait** (*\*\*kwargs*)

Polls `hatoba.Client.get_firewall_group()` every 20 seconds until a successful state is reached. An error is returned after 20 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    FirewallGroupName='string',
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **FirewallGroupName** (*string*) – [REQUIRED]
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 20

**Returns** None

*hatoba* / Waiter / SnapshotAvailable

## SnapshotAvailable

**class** hatoba.Waiter.**SnapshotAvailable**

```
waiter = client.get_waiter('snapshot_available')
```

**wait** (*\*\*kwargs*)

Polls `hatoba.Client.get_snapshot()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    SnapshotName='string',
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

**Parameters**

- **SnapshotName** (*string*) – **[REQUIRED]**
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None

## 1.5 nas

### 1.5.1 Client

**class** nas.**Client**

A low-level client representing NIFCLOUD NAS

```
client = session.create_client('nas')
```

These are the available methods:

*nas* / Client / `authorize_nas_security_group_ingress`

#### `authorize_nas_security_group_ingress`

`nas.Client.authorize_nas_security_group_ingress` (\*\*kwargs)See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.authorize_nas_security_group_ingress(  
    CIDRIP='string',  
    NASSecurityGroupName='string',  
    SecurityGroupName='string'  
)
```

**Parameters**

- **CIDRIP** (*string*) –
- **NASSecurityGroupName** (*string*) – **[REQUIRED]**
- **SecurityGroupName** (*string*) –

**Return type** dict**Returns****Response Syntax**

```
{  
    'NASSecurityGroup': {  
        'AvailabilityZone': 'string',  
        'IPRanges': [  
            {  
                'CIDRIP': 'string',  
                'Status': 'string'  
            },  
            ...  
        ]  
    },  
    ...  
}
```

(continues on next page)

(continued from previous page)

```

    ],
    'NASSecurityGroupDescription': 'string',
    'NASSecurityGroupName': 'string',
    'OwnerId': 'string',
    'SecurityGroups': [
        {
            'SecurityGroupName': 'string',
            'SecurityGroupOwnerId': 'string',
            'Status': 'string'
        },
    ]
},
'ResponseMetadata': {
    'RequestId': 'string'
}
}

```

**Response Structure**

- *(dict)* –
  - **NASSecurityGroup** (*dict*) –
    - \* **AvailabilityZone** (*string*) –
    - \* **IPRanges** (*list*) –
      - *(dict)* –
      - **CIDRIP** (*string*) –
      - **Status** (*string*) –
    - \* **NASSecurityGroupDescription** (*string*) –
    - \* **NASSecurityGroupName** (*string*) –
    - \* **OwnerId** (*string*) –
    - \* **SecurityGroups** (*list*) –
      - *(dict)* –
      - **SecurityGroupName** (*string*) –
      - **SecurityGroupOwnerId** (*string*) –
      - **Status** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*nas* / Client / can\_paginate**can\_paginate***nas*.Client.**can\_paginate**(*operation\_name*)

Check if an operation can be paginated.

**Parameters** *operation\_name* (*string*) – The operation name. This is the same name as the method name on the client. For example, if the method name is `create_foo`, and you'd normally invoke the operation as `client.create_foo(**kwargs)`, if the `create_foo` operation can be paginated, you can use the call `client.get_paginator("create_foo")`.

**Returns** True if the operation can be paginated, False otherwise.

*nas* / Client / clear\_nas\_session

## clear\_nas\_session

`nas.Client.clear_nas_session(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.clear_nas_session(  
    NASInstanceIdentifier='string',  
    SessionClearType='0'|'1'  
)
```

### Parameters

- **NASInstanceIdentifier** (*string*) – [REQUIRED]
- **SessionClearType** (*string*) –

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'NASInstance': {  
        'AllocatedStorage': 123,  
        'AuthenticationType': 123,  
        'AvailabilityZone': 'string',  
        'Endpoint': {  
            'Address': 'string',  
            'PrivateAddress': 'string'  
        },  
        'MasterUsername': 'string',  
        'NASInstanceClass': 'string',  
        'NASInstanceDescription': 'string',  
        'NASInstanceErrorInfo': {  
            'NASInstanceErrorCode': 'string',  
            'NASInstanceErrorMessage': 'string'  
        },  
        'NASInstanceIdentifier': 'string',  
        'NASInstanceStatus': 'string',  
        'NASInstanceType': 123,  
        'NASSecurityGroups': [  
            {  
                'NASSecurityGroupName': 'string',  
                'Status': 'string'  
            },  
            ...  
        ],  
        'NetworkId': 'string',  
        'NoRootSquash': True|False,  
        'Protocol': 'string',  
        'StorageType': 123,  
        'UpgradeRequired': True|False  
    },  
    'ResponseMetadata': {  
        'RequestId': 'string'  
    }  
}
```

### Response Structure

- (*dict*) –
  - **NASInstance** (*dict*) –

- \* **AllocatedStorage** (*integer*) –
- \* **AuthenticationType** (*integer*) –
- \* **AvailabilityZone** (*string*) –
- \* **Endpoint** (*dict*) –
  - **Address** (*string*) –
  - **PrivateAddress** (*string*) –
- \* **MasterUsername** (*string*) –
- \* **NASInstanceClass** (*string*) –
- \* **NASInstanceDescription** (*string*) –
- \* **NASInstanceErrorInfo** (*dict*) –
  - **NASInstanceErrorCode** (*string*) –
  - **NASInstanceErrorMessage** (*string*) –
- \* **NASInstanceIdentifier** (*string*) –
- \* **NASInstanceStatus** (*string*) –
- \* **NASInstanceType** (*integer*) –
- \* **NASSecurityGroups** (*list*) –
  - (*dict*) –
  - **NASSecurityGroupName** (*string*) –
  - **Status** (*string*) –
- \* **NetworkId** (*string*) –
- \* **NoRootSquash** (*boolean*) –
- \* **Protocol** (*string*) –
- \* **StorageType** (*integer*) –
- \* **UpgradeRequired** (*boolean*) –
- **ResponseMetadata** (*dict*) –
  - \* **RequestId** (*string*) –

*nas* / Client / close

## close

`nas.Client.close()`  
Closes underlying endpoint connections.

*nas* / Client / create\_nas\_instance

## create\_nas\_instance

`nas.Client.create_nas_instance(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.create_nas_instance(
    AllocatedStorage=123,
    AvailabilityZone='string',
    MasterPrivateAddress='string',
    MasterUserPassword='string',
    MasterUsername='string',
    NASInstanceDescription='string',
    NASInstanceIdentifier='string',
    NASInstanceType=123,
    NASSecurityGroups=[
        'string',
```

(continues on next page)

(continued from previous page)

```

    ],
    NetworkId='string',
    Protocol='nfs'|'cifs'
)

```

**Parameters**

- **AllocatedStorage** (*integer*) – [REQUIRED]
- **AvailabilityZone** (*string*) –
- **MasterPrivateAddress** (*string*) –
- **MasterUserPassword** (*string*) –
- **MasterUsername** (*string*) –
- **NASInstanceDescription** (*string*) –
- **NASInstanceIdentifier** (*string*) – [REQUIRED]
- **NASInstanceType** (*integer*) –
- **NASSecurityGroups** (*list*) –
  - (*string*) –
- **NetworkId** (*string*) –
- **Protocol** (*string*) – [REQUIRED]

**Return type** dict**Returns****Response Syntax**

```

{
  'NASInstance': {
    'AllocatedStorage': 123,
    'AuthenticationType': 123,
    'AvailabilityZone': 'string',
    'Endpoint': {
      'Address': 'string',
      'PrivateAddress': 'string'
    },
    'MasterUsername': 'string',
    'NASInstanceClass': 'string',
    'NASInstanceDescription': 'string',
    'NASInstanceErrorInfo': {
      'NASInstanceErrorCode': 'string',
      'NASInstanceErrorMessage': 'string'
    },
    'NASInstanceIdentifier': 'string',
    'NASInstanceStatus': 'string',
    'NASInstanceType': 123,
    'NASSecurityGroups': [
      {
        'NASSecurityGroupName': 'string',
        'Status': 'string'
      },
    ],
    'NetworkId': 'string',
    'NoRootSquash': True|False,
    'Protocol': 'string',
    'StorageType': 123,
    'UpgradeRequired': True|False
  },
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}

```

(continues on next page)



(continued from previous page)

```
}
}
```

**Response Structure**

- *(dict)* –
  - **NASInstance** (*dict*) –
    - \* **AllocatedStorage** (*integer*) –
    - \* **AuthenticationType** (*integer*) –
    - \* **AvailabilityZone** (*string*) –
    - \* **Endpoint** (*dict*) –
      - **Address** (*string*) –
      - **PrivateAddress** (*string*) –
    - \* **MasterUsername** (*string*) –
    - \* **NASInstanceClass** (*string*) –
    - \* **NASInstanceDescription** (*string*) –
    - \* **NASInstanceErrorInfo** (*dict*) –
      - **NASInstanceErrorCode** (*string*) –
      - **NASInstanceErrorMessage** (*string*) –
    - \* **NASInstanceIdentifier** (*string*) –
    - \* **NASInstanceStatus** (*string*) –
    - \* **NASInstanceType** (*integer*) –
    - \* **NASSecurityGroups** (*list*) –
      - (*dict*) –
      - **NASSecurityGroupName** (*string*) –
      - **Status** (*string*) –
    - \* **NetworkId** (*string*) –
    - \* **NoRootSquash** (*boolean*) –
    - \* **Protocol** (*string*) –
    - \* **StorageType** (*integer*) –
    - \* **UpgradeRequired** (*boolean*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*nas* / Client / `create_nas_security_group`

**create\_nas\_security\_group**

`nas.Client.create_nas_security_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.create_nas_security_group(
    AvailabilityZone='string',
    NASSecurityGroupDescription='string',
    NASSecurityGroupName='string'
)
```

**Parameters**

- **AvailabilityZone** (*string*) – [REQUIRED]
- **NASSecurityGroupDescription** (*string*) –
- **NASSecurityGroupName** (*string*) – [REQUIRED]

**Return type** dict

## Returns

### Response Syntax

```
{
  'NASSecurityGroup': {
    'AvailabilityZone': 'string',
    'IPRanges': [
      {
        'CIDRIP': 'string',
        'Status': 'string'
      },
    ],
    'NASSecurityGroupDescription': 'string',
    'NASSecurityGroupName': 'string',
    'OwnerId': 'string',
    'SecurityGroups': [
      {
        'SecurityGroupName': 'string',
        'SecurityGroupOwnerId': 'string',
        'Status': 'string'
      },
    ],
  },
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

### Response Structure

- (*dict*) –
  - **NASSecurityGroup** (*dict*) –
    - \* **AvailabilityZone** (*string*) –
    - \* **IPRanges** (*list*) –
      - (*dict*) –
      - **CIDRIP** (*string*) –
      - **Status** (*string*) –
    - \* **NASSecurityGroupDescription** (*string*) –
    - \* **NASSecurityGroupName** (*string*) –
    - \* **OwnerId** (*string*) –
    - \* **SecurityGroups** (*list*) –
      - (*dict*) –
      - **SecurityGroupName** (*string*) –
      - **SecurityGroupOwnerId** (*string*) –
      - **Status** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*nas* / Client / delete\_nas\_instance

### delete\_nas\_instance

`nas.Client.delete_nas_instance(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_nas_instance(
    NASInstanceIdentifier='string'
)
```

**Parameters** **NASInstanceIdentifier** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'NASInstance': {
    'AllocatedStorage': 123,
    'AuthenticationType': 123,
    'AvailabilityZone': 'string',
    'Endpoint': {
      'Address': 'string',
      'PrivateAddress': 'string'
    },
    'MasterUsername': 'string',
    'NASInstanceClass': 'string',
    'NASInstanceDescription': 'string',
    'NASInstanceErrorInfo': {
      'NASInstanceErrorCode': 'string',
      'NASInstanceErrorMessage': 'string'
    },
    'NASInstanceIdentifier': 'string',
    'NASInstanceStatus': 'string',
    'NASInstanceType': 123,
    'NASSecurityGroups': [
      {
        'NASSecurityGroupName': 'string',
        'Status': 'string'
      }
    ],
    'NetworkId': 'string',
    'NoRootSquash': True|False,
    'Protocol': 'string',
    'StorageType': 123,
    'UpgradeRequired': True|False
  },
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

#### Response Structure

- (*dict*) –
  - **NASInstance** (*dict*) –
    - \* **AllocatedStorage** (*integer*) –
    - \* **AuthenticationType** (*integer*) –
    - \* **AvailabilityZone** (*string*) –
    - \* **Endpoint** (*dict*) –
      - **Address** (*string*) –
      - **PrivateAddress** (*string*) –
    - \* **MasterUsername** (*string*) –
    - \* **NASInstanceClass** (*string*) –

- \* **NASInstanceDescription** (*string*) –
- \* **NASInstanceErrorInfo** (*dict*) –
  - **NASInstanceErrorCode** (*string*) –
  - **NASInstanceErrorMessage** (*string*) –
- \* **NASInstanceIdentifier** (*string*) –
- \* **NASInstanceStatus** (*string*) –
- \* **NASInstanceType** (*integer*) –
- \* **NASSecurityGroups** (*list*) –
  - (*dict*) –
  - **NASSecurityGroupName** (*string*) –
  - **Status** (*string*) –
- \* **NetworkId** (*string*) –
- \* **NoRootSquash** (*boolean*) –
- \* **Protocol** (*string*) –
- \* **StorageType** (*integer*) –
- \* **UpgradeRequired** (*boolean*) –
- **ResponseMetadata** (*dict*) –
  - \* **RequestId** (*string*) –

*nas* / Client / delete\_nas\_security\_group

### delete\_nas\_security\_group

`nas.Client.delete_nas_security_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.delete_nas_security_group(
    NASSecurityGroupName='string'
)
```

**Parameters** **NASSecurityGroupName** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

#### Response Structure

- (*dict*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*nas* / Client / describe\_nas\_instances

### describe\_nas\_instances

`nas.Client.describe_nas_instances(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

## Request Syntax

```
response = client.describe_nas_instances(
    NASInstanceIdentifier='string'
)
```

**Parameters** **NASInstanceIdentifier** (*string*) –

**Return type** dict

**Returns**

## Response Syntax

```
{
  'NASInstances': [
    {
      'AllocatedStorage': 123,
      'AuthenticationType': 123,
      'AvailabilityZone': 'string',
      'Endpoint': {
        'Address': 'string',
        'PrivateAddress': 'string'
      },
      'MasterUsername': 'string',
      'NASInstanceClass': 'string',
      'NASInstanceDescription': 'string',
      'NASInstanceErrorInfo': {
        'NASInstanceErrorCode': 'string',
        'NASInstanceErrorMessage': 'string'
      },
      'NASInstanceIdentifier': 'string',
      'NASInstanceStatus': 'string',
      'NASInstanceType': 123,
      'NASSecurityGroups': [
        {
          'NASSecurityGroupName': 'string',
          'Status': 'string'
        }
      ],
      'NetworkId': 'string',
      'NoRootSquash': True|False,
      'Protocol': 'string',
      'StorageType': 123,
      'UpgradeRequired': True|False
    },
  ],
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

## Response Structure

- (*dict*) –
  - **NASInstances** (*list*) –
    - \* (*dict*) –
      - **AllocatedStorage** (*integer*) –
      - **AuthenticationType** (*integer*) –
      - **AvailabilityZone** (*string*) –
      - **Endpoint** (*dict*) –

- **Address** (*string*) –
- **PrivateAddress** (*string*) –
- **MasterUsername** (*string*) –
- **NASInstanceClass** (*string*) –
- **NASInstanceDescription** (*string*) –
- **NASInstanceErrorInfo** (*dict*) –
- **NASInstanceErrorCode** (*string*) –
- **NASInstanceErrorMessage** (*string*) –
- **NASInstanceIdentifier** (*string*) –
- **NASInstanceStatus** (*string*) –
- **NASInstanceType** (*integer*) –
- **NASSecurityGroups** (*list*) –
- (*dict*) –
- **NASSecurityGroupName** (*string*) –
- **Status** (*string*) –
- **NetworkId** (*string*) –
- **NoRootSquash** (*boolean*) –
- **Protocol** (*string*) –
- **StorageType** (*integer*) –
- **UpgradeRequired** (*boolean*) –
- **ResponseMetadata** (*dict*) –
  - \* **RequestId** (*string*) –

*nas* / Client / describe\_nas\_security\_groups

## describe\_nas\_security\_groups

`nas.Client.describe_nas_security_groups(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_nas_security_groups(  
    NASSecurityGroupName='string'  
)
```

**Parameters** **NASSecurityGroupName** (*string*) –

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'NASSecurityGroups': [  
        {  
            'AvailabilityZone': 'string',  
            'IPRanges': [  
                {  
                    'CIDRIP': 'string',  
                    'Status': 'string'  
                },  
            ],  
            'NASSecurityGroupDescription': 'string',  
            'NASSecurityGroupName': 'string',  
            'OwnerId': 'string',  
            'SecurityGroups': [  

```

(continues on next page)

(continued from previous page)

```

        {
            'SecurityGroupName': 'string',
            'SecurityGroupOwnerId': 'string',
            'Status': 'string'
        },
    ],
    ],
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}

```

**Response Structure**

- (dict) –
  - **NASecurityGroups** (list) –
    - \* (dict) –
      - **AvailabilityZone** (string) –
      - **IPRanges** (list) –
      - (dict) –
      - **CIDRIP** (string) –
      - **Status** (string) –
      - **NASecurityGroupDescription** (string) –
      - **NASecurityGroupName** (string) –
      - **OwnerId** (string) –
      - **SecurityGroups** (list) –
      - (dict) –
      - **SecurityGroupName** (string) –
      - **SecurityGroupOwnerId** (string) –
      - **Status** (string) –
  - **ResponseMetadata** (dict) –
    - \* **RequestId** (string) –

*nas* / Client / get\_metric\_statistics

**get\_metric\_statistics**

`nas.Client.get_metric_statistics(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```

response = client.get_metric_statistics(
    Dimensions=[
        {
            'Name': 'string',
            'Value': 'string'
        },
    ],
    EndTime=datetime(2015, 1, 1),
    MetricName='FreeStorageSpace'|'UsedStorageSpace'|'ReadIOPS'|'WriteIOPS'|
    ↳ 'ReadThroughput'|'WriteThroughput'|'ActiveConnections'|'GlobalReadTraffic'|
    ↳ 'PrivateReadTraffic'|'GlobalWriteTraffic'|'PrivateWriteTraffic',
    StartTime=datetime(2015, 1, 1)
)

```

**Parameters**

- **Dimensions** (*list*) – [REQUIRED]
  - (*dict*) –
    - \* **Name** (*string*) – [REQUIRED]
    - \* **Value** (*string*) – [REQUIRED]
- **EndTime** (*datetime*) –
- **MetricName** (*string*) – [REQUIRED]
- **StartTime** (*datetime*) –

**Return type** dict**Returns****Response Syntax**

```
{
  'Datapoints': [
    {
      'SampleCount': 123,
      'Sum': 123.0,
      'TargetName': 'string',
      'Timestamp': datetime(2015, 1, 1)
    },
  ],
  'Label': 'string',
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

**Response Structure**

- (*dict*) –
  - **Datapoints** (*list*) –
    - \* (*dict*) –
      - **SampleCount** (*integer*) –
      - **Sum** (*float*) –
      - **TargetName** (*string*) –
      - **Timestamp** (*datetime*) –
  - **Label** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*nas* / Client / `get_paginator`**get\_paginator**`nas.Client.get_paginator(operation_name)`

Create a paginator for an operation.

**Parameters** **operation\_name** (*string*) – The operation name. This is the same name as the method name on the client. For example, if the method name is `create_foo`, and you'd normally invoke the operation as `client.create_foo(**kwargs)`, if the `create_foo` operation can be paginated, you can use the call `client.get_paginator("create_foo")`.

**Raises** **OperationNotPageableError** – Raised if the operation is not pageable. You can use the `client.can_paginate` method to check if an operation is pageable.

**Return type** L{botocore.paginator.Paginator}

**Returns** A paginator object.



*nas* / Client / `get_waiter`

## `get_waiter`

`nas.Client.get_waiter(waiter_name)`

Returns an object that can wait for some condition.

**Parameters** `waiter_name` (*str*) – The name of the waiter to get. See the waiters section of the service docs for a list of available waiters.

**Returns** The specified waiter object.

**Return type** `botocore.waiter.Waiter`

*nas* / Client / `modify_nas_instance`

## `modify_nas_instance`

`nas.Client.modify_nas_instance(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.modify_nas_instance(
    AllocatedStorage=123,
    AuthenticationType=123,
    MasterPrivateAddress='string',
    MasterUserPassword='string',
    NASInstanceDescription='string',
    NASInstanceIdentifier='string',
    NASSecurityGroups=[
        'string',
    ],
    NetworkId='string',
    NewNASInstanceIdentifier='string',
    NoRootSquash=True|False
)
```

### Parameters

- **AllocatedStorage** (*integer*) –
- **AuthenticationType** (*integer*) –
- **MasterPrivateAddress** (*string*) –
- **MasterUserPassword** (*string*) –
- **NASInstanceDescription** (*string*) –
- **NASInstanceIdentifier** (*string*) – [REQUIRED]
- **NASSecurityGroups** (*list*) –
  - (*string*) –
- **NetworkId** (*string*) –
- **NewNASInstanceIdentifier** (*string*) –
- **NoRootSquash** (*boolean*) –

**Return type** `dict`

**Returns**

### Response Syntax

```
{
  'NASInstance': {
    'AllocatedStorage': 123,
```

(continues on next page)

(continued from previous page)

```

    'AuthenticationType': 123,
    'AvailabilityZone': 'string',
    'Endpoint': {
        'Address': 'string',
        'PrivateAddress': 'string'
    },
    'MasterUsername': 'string',
    'NASInstanceClass': 'string',
    'NASInstanceDescription': 'string',
    'NASInstanceErrorInfo': {
        'NASInstanceErrorCode': 'string',
        'NASInstanceErrorMessage': 'string'
    },
    'NASInstanceIdentifier': 'string',
    'NASInstanceStatus': 'string',
    'NASInstanceType': 123,
    'NASSecurityGroups': [
        {
            'NASSecurityGroupName': 'string',
            'Status': 'string'
        },
    ],
    'NetworkId': 'string',
    'NoRootSquash': True|False,
    'Protocol': 'string',
    'StorageType': 123,
    'UpgradeRequired': True|False
},
'ResponseMetadata': {
    'RequestId': 'string'
}
}

```

### Response Structure

- (dict) –
  - **NASInstance** (dict) –
    - \* **AllocatedStorage** (integer) –
    - \* **AuthenticationType** (integer) –
    - \* **AvailabilityZone** (string) –
    - \* **Endpoint** (dict) –
      - **Address** (string) –
      - **PrivateAddress** (string) –
    - \* **MasterUsername** (string) –
    - \* **NASInstanceClass** (string) –
    - \* **NASInstanceDescription** (string) –
    - \* **NASInstanceErrorInfo** (dict) –
      - **NASInstanceErrorCode** (string) –
      - **NASInstanceErrorMessage** (string) –
    - \* **NASInstanceIdentifier** (string) –
    - \* **NASInstanceStatus** (string) –
    - \* **NASInstanceType** (integer) –
    - \* **NASSecurityGroups** (list) –
      - (dict) –
      - **NASSecurityGroupName** (string) –
      - **Status** (string) –

- \* **NetworkId** (*string*) –
- \* **NoRootSquash** (*boolean*) –
- \* **Protocol** (*string*) –
- \* **StorageType** (*integer*) –
- \* **UpgradeRequired** (*boolean*) –
- **ResponseMetadata** (*dict*) –
- \* **RequestId** (*string*) –

*nas* / Client / modify\_nas\_security\_group

## modify\_nas\_security\_group

`nas.Client.modify_nas_security_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.modify_nas_security_group(
    NASSecurityGroupDescription='string',
    NASSecurityGroupName='string',
    NewNASSecurityGroupName='string'
)
```

### Parameters

- **NASSecurityGroupDescription** (*string*) –
- **NASSecurityGroupName** (*string*) – [REQUIRED]
- **NewNASSecurityGroupName** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
    'NASSecurityGroup': {
        'AvailabilityZone': 'string',
        'IPRanges': [
            {
                'CIDRIP': 'string',
                'Status': 'string'
            },
        ],
        'NASSecurityGroupDescription': 'string',
        'NASSecurityGroupName': 'string',
        'OwnerId': 'string',
        'SecurityGroups': [
            {
                'SecurityGroupName': 'string',
                'SecurityGroupOwnerId': 'string',
                'Status': 'string'
            },
        ],
    },
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

### Response Structure

- (*dict*) –
  - **NASecurityGroup** (*dict*) –
    - \* **AvailabilityZone** (*string*) –
    - \* **IPRanges** (*list*) –
      - (*dict*) –
      - **CIDRIP** (*string*) –
      - **Status** (*string*) –
    - \* **NASecurityGroupDescription** (*string*) –
    - \* **NASecurityGroupName** (*string*) –
    - \* **OwnerId** (*string*) –
    - \* **SecurityGroups** (*list*) –
      - (*dict*) –
      - **SecurityGroupName** (*string*) –
      - **SecurityGroupOwnerId** (*string*) –
      - **Status** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*nas* / Client / revoke\_nas\_security\_group\_ingress

## revoke\_nas\_security\_group\_ingress

`nas.Client.revoke_nas_security_group_ingress(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.revoke_nas_security_group_ingress(  
    CIDRIP='string',  
    NASSecurityGroupName='string',  
    SecurityGroupName='string'  
)
```

### Parameters

- **CIDRIP** (*string*) –
- **NASSecurityGroupName** (*string*) – [REQUIRED]
- **SecurityGroupName** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{  
    'NASSecurityGroup': {  
        'AvailabilityZone': 'string',  
        'IPRanges': [  
            {  
                'CIDRIP': 'string',  
                'Status': 'string'  
            },  
        ],  
        'NASSecurityGroupDescription': 'string',  
        'NASSecurityGroupName': 'string',  
        'OwnerId': 'string',  
        'SecurityGroups': [  
            {  
                'CIDRIP': 'string',  
                'Status': 'string'  
            },  
        ],  
        'SecurityGroupName': 'string',  
        'SecurityGroupOwnerId': 'string'  
    },  
    'ResponseMetadata': {  
        'RequestId': 'string'  
    }
```

(continues on next page)

(continued from previous page)

```

        'SecurityGroupName': 'string',
        'SecurityGroupOwnerId': 'string',
        'Status': 'string'
    },
]
},
'ResponseMetadata': {
    'RequestId': 'string'
}
}

```

**Response Structure**

- (dict) –
  - **NASSecurityGroup** (dict) –
    - \* **AvailabilityZone** (string) –
    - \* **IPRanges** (list) –
      - (dict) –
      - **CIDRIP** (string) –
      - **Status** (string) –
    - \* **NASSecurityGroupDescription** (string) –
    - \* **NASSecurityGroupName** (string) –
    - \* **OwnerId** (string) –
    - \* **SecurityGroups** (list) –
      - (dict) –
      - **SecurityGroupName** (string) –
      - **SecurityGroupOwnerId** (string) –
      - **Status** (string) –
  - **ResponseMetadata** (dict) –
    - \* **RequestId** (string) –

*nas* / Client / upgrade\_nas\_instance

**upgrade\_nas\_instance**

`nas.Client.upgrade_nas_instance(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```

response = client.upgrade_nas_instance(
    NASInstanceIdentifier='string'
)

```

**Parameters** **NASInstanceIdentifier** (string) – [REQUIRED]

**Return type** dict

**Returns**

**Response Syntax**

```

{
    'NASInstance': {
        'AllocatedStorage': 123,
        'AuthenticationType': 123,
        'AvailabilityZone': 'string',
        'Endpoint': {

```

(continues on next page)

(continued from previous page)

```

        'Address': 'string',
        'PrivateAddress': 'string'
    },
    'MasterUsername': 'string',
    'NASInstanceClass': 'string',
    'NASInstanceDescription': 'string',
    'NASInstanceErrorInfo': {
        'NASInstanceErrorCode': 'string',
        'NASInstanceErrorMessage': 'string'
    },
    'NASInstanceIdentifier': 'string',
    'NASInstanceStatus': 'string',
    'NASInstanceType': 123,
    'NASSecurityGroups': [
        {
            'NASSecurityGroupName': 'string',
            'Status': 'string'
        }
    ],
    'NetworkId': 'string',
    'NoRootSquash': True|False,
    'Protocol': 'string',
    'StorageType': 123,
    'UpgradeRequired': True|False
},
'ResponseMetadata': {
    'RequestId': 'string'
}
}

```

### Response Structure

- (dict) –
  - **NASInstance** (dict) –
    - \* **AllocatedStorage** (integer) –
    - \* **AuthenticationType** (integer) –
    - \* **AvailabilityZone** (string) –
    - \* **Endpoint** (dict) –
      - **Address** (string) –
      - **PrivateAddress** (string) –
    - \* **MasterUsername** (string) –
    - \* **NASInstanceClass** (string) –
    - \* **NASInstanceDescription** (string) –
    - \* **NASInstanceErrorInfo** (dict) –
      - **NASInstanceErrorCode** (string) –
      - **NASInstanceErrorMessage** (string) –
    - \* **NASInstanceIdentifier** (string) –
    - \* **NASInstanceStatus** (string) –
    - \* **NASInstanceType** (integer) –
    - \* **NASSecurityGroups** (list) –
      - (dict) –
      - **NASSecurityGroupName** (string) –
      - **Status** (string) –
    - \* **NetworkId** (string) –
    - \* **NoRootSquash** (boolean) –
    - \* **Protocol** (string) –

- \* **StorageType** (*integer*) –
- \* **UpgradeRequired** (*boolean*) –
- **ResponseMetadata** (*dict*) –
- \* **RequestId** (*string*) –

## 1.5.2 Client Exceptions

Client exceptions are available on a client instance via the `exceptions` property. For more detailed instructions and examples on the exact usage of client exceptions, see the error handling [user guide](#).

This client has no modeled exception classes.

## 1.5.3 Waiters

Waiters are available on a client instance via the `get_waiter` method. For more detailed instructions and examples on the usage of waiters, see the waiters [user guide](#).

The available waiters are:

*nas* / Waiter / NASInstanceAvailable

### NASInstanceAvailable

**class** `nas.Waiter.NASInstanceAvailable`

```
waiter = client.get_waiter('nas_instance_available')
```

**wait** (*\*\*kwargs*)

Polls `nas.Client.describe_nas_instances()` every 40 seconds until a successful state is reached. An error is returned after 80 failed checks.

See also: [AWS API Documentation](#)

#### Request Syntax

```
waiter.wait(
    NASInstanceIdentifier='string',
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

#### Parameters

- **NASInstanceIdentifier** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 40
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 80

**Returns** None

*nas* / Waiter / NASInstanceDeleted

## NASInstanceDeleted

**class** nas.Waiter.NASInstanceDeleted

```
waiter = client.get_waiter('nas_instance_deleted')
```

**wait** (*\*\*kwargs*)

Polls `nas.Client.describe_nas_instances()` every 40 seconds until a successful state is reached. An error is returned after 80 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    NASInstanceIdentifier='string',
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **NASInstanceIdentifier** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 40
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 80

**Returns** None

*nas* / Waiter / NASInstanceExists

## NASInstanceExists

**class** nas.Waiter.NASInstanceExists

```
waiter = client.get_waiter('nas_instance_exists')
```

**wait** (*\*\*kwargs*)

Polls `nas.Client.describe_nas_instances()` every 40 seconds until a successful state is reached. An error is returned after 80 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    NASInstanceIdentifier='string',
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```



**Parameters**

- **NASInstanceIdentifier** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 40
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 80

**Returns** None*nas* / Waiter / NASInstanceFailed**NASInstanceFailed****class** nas.Waiter.NASInstanceFailed

```
waiter = client.get_waiter('nas_instance_failed')
```

**wait** (*\*\*kwargs*)

Polls *nas.Client.describe\_nas\_instances()* every 40 seconds until a successful state is reached. An error is returned after 80 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```
waiter.wait(
    NASInstanceIdentifier='string',
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

**Parameters**

- **NASInstanceIdentifier** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 40
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 80

**Returns** None*nas* / Waiter / NASInstanceStorageFull**NASInstanceStorageFull****class** nas.Waiter.NASInstanceStorageFull

```
waiter = client.get_waiter('nas_instance_storage_full')
```

**wait** (\*\*kwargs)

Polls `nas.Client.describe_nas_instances()` every 40 seconds until a successful state is reached. An error is returned after 80 failed checks.

See also: [AWS API Documentation](#)

#### Request Syntax

```
waiter.wait(  
    NASInstanceIdentifier='string',  
    WaiterConfig={  
        'Delay': 123,  
        'MaxAttempts': 123  
    }  
)
```

#### Parameters

- **NASInstanceIdentifier** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 40
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 80

**Returns** None

*nas* / Waiter / NASSecurityGroupDeleted

### NASSecurityGroupDeleted

**class** nas.Waiter.NASSecurityGroupDeleted

```
waiter = client.get_waiter('nas_security_group_deleted')
```

**wait** (\*\*kwargs)

Polls `nas.Client.describe_nas_security_groups()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

#### Request Syntax

```
waiter.wait(  
    NASSecurityGroupName='string',  
    WaiterConfig={  
        'Delay': 123,  
        'MaxAttempts': 123  
    }  
)
```

#### Parameters

- **NASSecurityGroupName** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.

- **Delay** (*integer*) –

The amount of time in seconds to wait between attempts. Default: 20

- **MaxAttempts** (*integer*) –

The maximum number of attempts to be made. Default: 40

**Returns** None

*nas* / Waiter / NASSecurityGroupExists

## NASSecurityGroupExists

**class** `nas.Waiter.NASSecurityGroupExists`

```
waiter = client.get_waiter('nas_security_group_exists')
```

**wait** (*\*\*kwargs*)

Polls `nas.Client.describe_nas_security_groups()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    NASSecurityGroupName='string',
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **NASSecurityGroupName** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None

*nas* / Waiter / NASSecurityGroupIPRangesAuthFailed

## NASSecurityGroupIPRangesAuthFailed

**class** `nas.Waiter.NASSecurityGroupIPRangesAuthFailed`

```
waiter = client.get_waiter('nas_security_group_ip_ranges_auth_failed')
```

**wait** (*\*\*kwargs*)

Polls `nas.Client.describe_nas_security_groups()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(  
    NASSecurityGroupName='string',  
    WaiterConfig={  
        'Delay': 123,  
        'MaxAttempts': 123  
    }  
)
```

#### Parameters

- **NASSecurityGroupName** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None

*nas* / Waiter / NASSecurityGroupIPRangesAuthorized

### NASSecurityGroupIPRangesAuthorized

**class** *nas.Waiter.NASSecurityGroupIPRangesAuthorized*

```
waiter = client.get_waiter('nas_security_group_ip_ranges_authorized')
```

**wait** (*\*\*kwargs*)

Polls *nas.Client.describe\_nas\_security\_groups()* every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(  
    NASSecurityGroupName='string',  
    WaiterConfig={  
        'Delay': 123,  
        'MaxAttempts': 123  
    }  
)
```

#### Parameters

- **NASSecurityGroupName** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None

*nas* / Waiter / NASSecurityGroupIPRangesEmptied

## NASSecurityGroupIPRangesEmptied

**class** `nas.Waiter.NASSecurityGroupIPRangesEmptied`

```
waiter = client.get_waiter('nas_security_group_ip_ranges_emptied')
```

**wait** (*\*\*kwargs*)

Polls `nas.Client.describe_nas_security_groups()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    NASSecurityGroupName='string',
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **NASSecurityGroupName** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None

*nas* / Waiter / NASSecurityGroupIPRangesRevokeFailed

## NASSecurityGroupIPRangesRevokeFailed

**class** `nas.Waiter.NASSecurityGroupIPRangesRevokeFailed`

```
waiter = client.get_waiter('nas_security_group_ip_ranges_revoke_failed')
```

**wait** (*\*\*kwargs*)

Polls `nas.Client.describe_nas_security_groups()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(  
    NASSecurityGroupName='string',  
    WaiterConfig={  
        'Delay': 123,  
        'MaxAttempts': 123  
    }  
)
```

**Parameters**

- **NASSecurityGroupName** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None*nas* / Waiter / NASSecurityGroupSecurityGroupsAuthFailed**NASSecurityGroupSecurityGroupsAuthFailed****class** *nas*.Waiter.NASSecurityGroupSecurityGroupsAuthFailed

```
waiter = client.get_waiter('nas_security_group_security_groups_auth_failed')
```

**wait** (*\*\*kwargs*)

Polls *nas.Client.describe\_nas\_security\_groups()* every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```
waiter.wait(  
    NASSecurityGroupName='string',  
    WaiterConfig={  
        'Delay': 123,  
        'MaxAttempts': 123  
    }  
)
```

**Parameters**

- **NASSecurityGroupName** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None*nas* / Waiter / NASSecurityGroupSecurityGroupsAuthorized

## NASecurityGroupSecurityGroupsAuthorized

**class** `nas.Waiter.NASecurityGroupSecurityGroupsAuthorized`

```
waiter = client.get_waiter('nas_security_group_security_groups_authorized')
```

**wait** (*\*\*kwargs*)

Polls `nas.Client.describe_nas_security_groups()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    NASSecurityGroupName='string',
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **NASecurityGroupName** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None

*nas* / Waiter / NASecurityGroupSecurityGroupsEmptyed

## NASecurityGroupSecurityGroupsEmptyed

**class** `nas.Waiter.NASecurityGroupSecurityGroupsEmptyed`

```
waiter = client.get_waiter('nas_security_group_security_groups_emptyed')
```

**wait** (*\*\*kwargs*)

Polls `nas.Client.describe_nas_security_groups()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    NASSecurityGroupName='string',
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

**Parameters**

- **NASecurityGroupName** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None*nas* / Waiter / NASSecurityGroupSecurityGroupsRevokeFailed**NASSecurityGroupSecurityGroupsRevokeFailed****class** *nas*.Waiter.NASSecurityGroupSecurityGroupsRevokeFailed

```
waiter = client.get_waiter('nas_security_group_security_groups_revoke_failed')
```

**wait** (*\*\*kwargs*)

Polls *nas.Client.describe\_nas\_security\_groups()* every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```
waiter.wait(  
    NASSecurityGroupName='string',  
    WaiterConfig={  
        'Delay': 123,  
        'MaxAttempts': 123  
    }  
)
```

**Parameters**

- **NASecurityGroupName** (*string*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None

## 1.6 rdb

### 1.6.1 Client

**class** *rdb*.Client

A low-level client representing NIFCLOUD RDB



```
client = session.create_client('rdb')
```

These are the available methods:

*rdb* / Client / `add_source_identifier_to_subscription`

### `add_source_identifier_to_subscription`

`rdb.Client.add_source_identifier_to_subscription(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.add_source_identifier_to_subscription(
    SourceIdentifier='string',
    SubscriptionName='string'
)
```

#### Parameters

- **SourceIdentifier** (*string*) – [REQUIRED]
- **SubscriptionName** (*string*) – [REQUIRED]

**Return type** dict

#### Returns

#### Response Syntax

```
{
  'EventSubscription': {
    'CustSubscriptionId': 'string',
    'Enabled': True|False,
    'EventCategoriesList': [
      'string',
    ],
    'NiftyDescription': 'string',
    'NiftyEmailAddressesList': [
      'string',
    ],
    'SourceIdsList': [
      'string',
    ],
    'SourceType': 'string',
    'Status': 'string',
    'SubscriptionCreationTime': 'string'
  },
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

#### Response Structure

- (*dict*) –
  - **EventSubscription** (*dict*) –
    - \* **CustSubscriptionId** (*string*) –
    - \* **Enabled** (*boolean*) –
    - \* **EventCategoriesList** (*list*) –
      - (*string*) –
    - \* **NiftyDescription** (*string*) –

- \* **NiftyEmailAddressesList** (*list*) –
  - (*string*) –
- \* **SourceIdsList** (*list*) –
  - (*string*) –
- \* **SourceType** (*string*) –
- \* **Status** (*string*) –
- \* **SubscriptionCreationTime** (*string*) –
- **ResponseMetadata** (*dict*) –
  - \* **RequestId** (*string*) –

*rdb* / Client / `authorize_db_security_group_ingress`

## `authorize_db_security_group_ingress`

`rdb.Client.authorize_db_security_group_ingress` (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.authorize_db_security_group_ingress(  
    CIDRIP='string',  
    DBSecurityGroupName='string',  
    EC2SecurityGroupId='string',  
    EC2SecurityGroupName='string',  
    EC2SecurityGroupOwnerId='string'  
)
```

### Parameters

- **CIDRIP** (*string*) –
- **DBSecurityGroupName** (*string*) – [REQUIRED]
- **EC2SecurityGroupId** (*string*) –
- **EC2SecurityGroupName** (*string*) –
- **EC2SecurityGroupOwnerId** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{  
    'DBSecurityGroup': {  
        'DBSecurityGroupDescription': 'string',  
        'DBSecurityGroupName': 'string',  
        'EC2SecurityGroups': [  
            {  
                'EC2SecurityGroupName': 'string',  
                'EC2SecurityGroupOwnerId': 'string',  
                'Status': 'string'  
            },  
            ...  
        ],  
        'IPRanges': [  
            {  
                'CIDRIP': 'string',  
                'Status': 'string'  
            },  
            ...  
        ],  
        'NiftyAvailabilityZone': 'string',  
    },  
    'ResponseMetadata': {  
        'RequestId': 'string',  
        'HTTPStatusCode': 200,  
        'Headers': {'x-amzn-requestid': 'string'},  
        'NextToken': 'string',  
        'MaxResults': 100,  
        'ActualResults': 100,  
    },  
}
```

(continues on next page)

(continued from previous page)

```

        'OwnerId': 'string'
    },
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}

```

**Response Structure**

- (dict) –
  - DBSecurityGroup (dict) –
    - \* DBSecurityGroupDescription (string) –
    - \* DBSecurityGroupName (string) –
    - \* EC2SecurityGroups (list) –
      - (dict) –
      - EC2SecurityGroupName (string) –
      - EC2SecurityGroupOwnerId (string) –
      - Status (string) –
    - \* IPRanges (list) –
      - (dict) –
      - CIDRIP (string) –
      - Status (string) –
    - \* NiftyAvailabilityZone (string) –
    - \* OwnerId (string) –
  - ResponseMetadata (dict) –
    - \* RequestId (string) –

*rdb* / Client / can\_paginate**can\_paginate**`rdb.Client.can_paginate(operation_name)`

Check if an operation can be paginated.

**Parameters** `operation_name` (string) – The operation name. This is the same name as the method name on the client. For example, if the method name is `create_foo`, and you'd normally invoke the operation as `client.create_foo(**kwargs)`, if the `create_foo` operation can be paginated, you can use the call `client.get_paginator("create_foo")`.

**Returns** True if the operation can be paginated, False otherwise.

*rdb* / Client / close**close**`rdb.Client.close()`

Closes underlying endpoint connections.

*rdb* / Client / copy\_db\_snapshot**copy\_db\_snapshot**`rdb.Client.copy_db_snapshot(**kwargs)`See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.copy_db_snapshot(  
    SourceDBSnapshotIdentifier='string',  
    TargetDBSnapshotIdentifier='string'  
)
```

#### Parameters

- **SourceDBSnapshotIdentifier** (*string*) – [REQUIRED]
- **TargetDBSnapshotIdentifier** (*string*) – [REQUIRED]

**Return type** dict

#### Returns

### Response Syntax

```
{  
    'DBSnapshot': {  
        'AllocatedStorage': 123,  
        'AvailabilityZone': 'string',  
        'DBInstanceIdentifier': 'string',  
        'DBSnapshotIdentifier': 'string',  
        'Engine': 'string',  
        'EngineVersion': 'string',  
        'InstanceCreateTime': datetime(2015, 1, 1),  
        'LicenseModel': 'string',  
        'MasterUsername': 'string',  
        'OptionGroupName': 'string',  
        'Port': 123,  
        'SnapshotCreateTime': datetime(2015, 1, 1),  
        'SnapshotType': 'string',  
        'Status': 'string'  
    },  
    'ResponseMetadata': {  
        'RequestId': 'string'  
    }  
}
```

### Response Structure

- (*dict*) –
  - **DBSnapshot** (*dict*) –
    - \* **AllocatedStorage** (*integer*) –
    - \* **AvailabilityZone** (*string*) –
    - \* **DBInstanceIdentifier** (*string*) –
    - \* **DBSnapshotIdentifier** (*string*) –
    - \* **Engine** (*string*) –
    - \* **EngineVersion** (*string*) –
    - \* **InstanceCreateTime** (*datetime*) –
    - \* **LicenseModel** (*string*) –
    - \* **MasterUsername** (*string*) –
    - \* **OptionGroupName** (*string*) –
    - \* **Port** (*integer*) –
    - \* **SnapshotCreateTime** (*datetime*) –
    - \* **SnapshotType** (*string*) –
    - \* **Status** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*rdb* / Client / create\_db\_instance

## create\_db\_instance

`rdb.Client.create_db_instance(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.create_db_instance(
    AccountingType='1'|'2',
    AllocatedStorage=123,
    AutoMinorVersionUpgrade=True|False,
    AvailabilityZone='string',
    BackupRetentionPeriod=123,
    CharacterSetName='string',
    DBInstanceClass='db.mini'|'db.e-small1'|'db.small1'|'db.e-small12'|'db.small12'|
    ↪ 'db.e-small14'|'db.small14'|'db.e-small18'|'db.small18'|'db.e-small116'|'db.small116'|
    ↪ 'db.e-medium'|'db.medium'|'db.e-medium4'|'db.medium4'|'db.e-medium8'|'db.medium8'
    ↪ '| 'db.e-medium16'|'db.medium16'|'db.e-medium24'|'db.medium24'|'db.e-large'|'db.
    ↪ large'|'db.e-large8'|'db.large8'|'db.e-large16'|'db.large16'|'db.e-large24'|'db.
    ↪ large24'|'db.e-large32'|'db.large32'|'db.e-extra-large8'|'db.extra-large8'|'db.
    ↪ e-extra-large16'|'db.extra-large16'|'db.e-extra-large24'|'db.extra-large24'|'db.
    ↪ e-extra-large32'|'db.extra-large32'|'db.e-extra-large48'|'db.extra-large48'|'db.
    ↪ e-double-large16'|'db.double-large16'|'db.e-double-large24'|'db.double-large24'|
    ↪ 'db.e-double-large32'|'db.double-large32'|'db.e-double-large48'|'db.double-
    ↪ large48'|'db.e-double-large64'|'db.double-large64'|'db.e-double-large96'|'db.
    ↪ double-large96'|'db.triple-large32'|'db.triple-large48'|'db.triple-large64'|'db.
    ↪ triple-large96'|'db.triple-large128'|'db.quad-large64'|'db.quad-large96'|'db.
    ↪ quad-large128'|'db.septa-large128',
    DBInstanceIdentifier='string',
    DBName='string',
    DBParameterGroupName='string',
    DBSecurityGroups=[
        'string',
    ],
    DBSubnetGroupName='string',
    Engine='MySQL'|'postgres',
    EngineVersion='string',
    Iops=123,
    LicenseModel='string',
    MasterUserPassword='string',
    MasterUsername='string',
    MultiAZ=True|False,
    NiftyMasterPrivateAddress='string',
    NiftyMultiAZType=123,
    NiftyNetworkId='string',
    NiftySlavePrivateAddress='string',
    NiftyStorageType=123,
    NiftyVirtualPrivateAddress='string',
    OptionGroupName='string',
    Port=123,
    PreferredBackupWindow='string',
    PreferredMaintenanceWindow='string',
    PubliclyAccessible=True|False,
    VpcSecurityGroupIds=[
        'string',
```

(continues on next page)

(continued from previous page)

```
]
)
```

**Parameters**

- **AccountingType** (*string*) –
- **AllocatedStorage** (*integer*) – [REQUIRED]
- **AutoMinorVersionUpgrade** (*boolean*) –
- **AvailabilityZone** (*string*) –
- **BackupRetentionPeriod** (*integer*) –
- **CharacterSetName** (*string*) –
- **DBInstanceClass** (*string*) – [REQUIRED]
- **DBInstanceIdentifier** (*string*) – [REQUIRED]
- **DBName** (*string*) –
- **DBParameterGroupName** (*string*) –
- **DBSecurityGroups** (*list*) –
  - (*string*) –
- **DBSubnetGroupName** (*string*) –
- **Engine** (*string*) – [REQUIRED]
- **EngineVersion** (*string*) –
- **Iops** (*integer*) –
- **LicenseModel** (*string*) –
- **MasterUserPassword** (*string*) – [REQUIRED]
- **MasterUsername** (*string*) – [REQUIRED]
- **MultiAZ** (*boolean*) –
- **NiftyMasterPrivateAddress** (*string*) –
- **NiftyMultiAZType** (*integer*) –
- **NiftyNetworkId** (*string*) –
- **NiftySlavePrivateAddress** (*string*) –
- **NiftyStorageType** (*integer*) –
- **NiftyVirtualPrivateAddress** (*string*) –
- **OptionGroupName** (*string*) –
- **Port** (*integer*) –
- **PreferredBackupWindow** (*string*) –
- **PreferredMaintenanceWindow** (*string*) –
- **PubliclyAccessible** (*boolean*) –
- **VpcSecurityGroupIds** (*list*) –
  - (*string*) –

**Return type** dict**Returns****Response Syntax**

```
{
  'DBInstance': {
    'AccountingType': 'string',
    'AllocatedStorage': 123,
    'AutoMinorVersionUpgrade': True|False,
    'AvailabilityZone': 'string',
    'BackupRetentionPeriod': 123,
    'BinlogRetentionPeriod': 123,
    'CACertificateIdentifier': 'string',
    'DBInstanceClass': 'string',
    'DBInstanceIdentifier': 'string',
    'DBInstanceStatus': 'string',
```

(continues on next page)

(continued from previous page)

```

'DBName': 'string',
'DBParameterGroups': [
    {
        'DBParameterGroupName': 'string',
        'ParameterApplyStatus': 'string'
    },
],
'DBSecurityGroups': [
    {
        'DBSecurityGroupName': 'string',
        'Status': 'string'
    },
],
'Endpoint': {
    'Address': 'string',
    'NiftyPrivateAddress': 'string',
    'Port': 123
},
'Engine': 'string',
'EngineVersion': 'string',
'ExternalReplicationInfo': {
    'ExternalMasterAddress': 'string',
    'ExternalReplicationMessage': 'string',
    'ExternalReplicationStatus': 'string',
    'ReplicationAddresses': [
        'string',
    ],
    'ReplicationPrivateAddresses': [
        'string',
    ]
},
'InstanceCreateTime': datetime(2015, 1, 1),
'LatestRestorableTime': datetime(2015, 1, 1),
'LicenseModel': 'string',
'MasterUsername': 'string',
'MultiAZ': True|False,
'NextMonthAccountingType': 'string',
'NiftyMasterPrivateAddress': 'string',
'NiftyMultiAZType': 'string',
'NiftyNetworkId': 'string',
'NiftySlavePrivateAddress': 'string',
'NiftyStorageType': 123,
'OptionGroupMemberships': [
    {
        'OptionGroupName': 'string',
        'Status': 'string'
    },
],
'PendingModifiedValues': {
    'AllocatedStorage': 123,
    'BackupRetentionPeriod': 123,
    'DBInstanceClass': 'string',
    'DBInstanceIdentifier': 'string',
    'EngineVersion': 'string',
    'MasterUserPassword': 'string',
    'MultiAZ': True|False,
    'Port': 123
}

```

(continues on next page)

(continued from previous page)

```

    },
    'PreferredBackupWindow': 'string',
    'PreferredMaintenanceWindow': 'string',
    'PubliclyAccessible': True|False,
    'ReadReplicaDBInstanceIdentifiers': [
        'string',
    ],
    'ReadReplicaSourceDBInstanceIdentifier': 'string',
    'SecondaryAvailabilityZone': 'string',
    'StatusInfos': [
        {
            'Message': 'string',
            'Normal': True|False,
            'Status': 'string',
            'StatusType': 'string'
        },
    ],
    'VpcSecurityGroups': 'string'
},
'ResponseMetadata': {
    'RequestId': 'string'
}
}

```

**Response Structure**

- (dict) –
  - **DBInstance** (dict) –
    - \* **AccountingType** (string) –
    - \* **AllocatedStorage** (integer) –
    - \* **AutoMinorVersionUpgrade** (boolean) –
    - \* **AvailabilityZone** (string) –
    - \* **BackupRetentionPeriod** (integer) –
    - \* **BinlogRetentionPeriod** (integer) –
    - \* **CACertificateIdentifier** (string) –
    - \* **DBInstanceClass** (string) –
    - \* **DBInstanceIdentifier** (string) –
    - \* **DBInstanceStatus** (string) –
    - \* **DBName** (string) –
    - \* **DBParameterGroups** (list) –
      - (dict) –
      - **DBParameterGroupName** (string) –
      - **ParameterApplyStatus** (string) –
    - \* **DBSecurityGroups** (list) –
      - (dict) –
      - **DBSecurityGroupName** (string) –
      - **Status** (string) –
    - \* **Endpoint** (dict) –
      - **Address** (string) –
      - **NiftyPrivateAddress** (string) –
      - **Port** (integer) –
    - \* **Engine** (string) –
    - \* **EngineVersion** (string) –
    - \* **ExternalReplicationInfo** (dict) –
      - **ExternalMasterAddress** (string) –
      - **ExternalReplicationMessage** (string) –



- **ExternalReplicationStatus** (*string*) –
- **ReplicationAddresses** (*list*) –
- (*string*) –
- **ReplicationPrivateAddresses** (*list*) –
- (*string*) –
- \* **InstanceCreateTime** (*datetime*) –
- \* **LatestRestorableTime** (*datetime*) –
- \* **LicenseModel** (*string*) –
- \* **MasterUsername** (*string*) –
- \* **MultiAZ** (*boolean*) –
- \* **NextMonthAccountingType** (*string*) –
- \* **NiftyMasterPrivateAddress** (*string*) –
- \* **NiftyMultiAZType** (*string*) –
- \* **NiftyNetworkId** (*string*) –
- \* **NiftySlavePrivateAddress** (*string*) –
- \* **NiftyStorageType** (*integer*) –
- \* **OptionGroupMemberships** (*list*) –
  - (*dict*) –
  - **OptionGroupName** (*string*) –
  - **Status** (*string*) –
- \* **PendingModifiedValues** (*dict*) –
  - **AllocatedStorage** (*integer*) –
  - **BackupRetentionPeriod** (*integer*) –
  - **DBInstanceClass** (*string*) –
  - **DBInstanceIdentifier** (*string*) –
  - **EngineVersion** (*string*) –
  - **MasterUserPassword** (*string*) –
  - **MultiAZ** (*boolean*) –
  - **Port** (*integer*) –
- \* **PreferredBackupWindow** (*string*) –
- \* **PreferredMaintenanceWindow** (*string*) –
- \* **PubliclyAccessible** (*boolean*) –
- \* **ReadReplicaDBInstanceIdentifiers** (*list*) –
  - (*string*) –
- \* **ReadReplicaSourceDBInstanceIdentifier** (*string*) –
- \* **SecondaryAvailabilityZone** (*string*) –
- \* **StatusInfos** (*list*) –
  - (*dict*) –
  - **Message** (*string*) –
  - **Normal** (*boolean*) –
  - **Status** (*string*) –
  - **StatusType** (*string*) –
- \* **VpcSecurityGroups** (*string*) –
- **ResponseMetadata** (*dict*) –
  - \* **RequestId** (*string*) –

*rdb* / Client / create\_db\_instance\_read\_replica

## create\_db\_instance\_read\_replica

`rdb.Client.create_db_instance_read_replica(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```

response = client.create_db_instance_read_replica(
    AccountingType='1'|'2',
    AutoMinorVersionUpgrade=True|False,
    AvailabilityZone='string',
    DBInstanceClass='db.mini'|'db.e-small1'|'db.small1'|'db.e-small12'|'db.small12'|
↪ 'db.e-small14'|'db.small14'|'db.e-small18'|'db.small18'|'db.e-small16'|'db.small16'|
↪ 'db.e-medium'|'db.medium'|'db.e-medium4'|'db.medium4'|'db.e-medium8'|'db.medium8'
↪ '| 'db.e-medium16'|'db.medium16'|'db.e-medium24'|'db.medium24'|'db.e-large'|'db.
↪ large'|'db.e-large8'|'db.large8'|'db.e-large16'|'db.large16'|'db.e-large24'|'db.
↪ large24'|'db.e-large32'|'db.large32'|'db.e-extra-large8'|'db.extra-large8'|'db.
↪ e-extra-large16'|'db.extra-large16'|'db.e-extra-large24'|'db.extra-large24'|'db.
↪ e-extra-large32'|'db.extra-large32'|'db.e-extra-large48'|'db.extra-large48'|'db.
↪ e-double-large16'|'db.double-large16'|'db.e-double-large24'|'db.double-large24'|
↪ 'db.e-double-large32'|'db.double-large32'|'db.e-double-large48'|'db.double-
↪ large48'|'db.e-double-large64'|'db.double-large64'|'db.e-double-large96'|'db.
↪ double-large96'|'db.triple-large32'|'db.triple-large48'|'db.triple-large64'|'db.
↪ triple-large96'|'db.triple-large128'|'db.quad-large64'|'db.quad-large96'|'db.
↪ quad-large128'|'db.septa-large128',
    DBInstanceIdentifier='string',
    Iops=123,
    NiftyReadReplicaPrivateAddress='string',
    NiftyStorageType=123,
    OptionGroupName='string',
    Port=123,
    PubliclyAccessible=True|False,
    SourceDBInstanceIdentifier='string'
)

```

### Parameters

- **AccountingType** (*string*) –
- **AutoMinorVersionUpgrade** (*boolean*) –
- **AvailabilityZone** (*string*) –
- **DBInstanceClass** (*string*) –
- **DBInstanceIdentifier** (*string*) – [REQUIRED]
- **Iops** (*integer*) –
- **NiftyReadReplicaPrivateAddress** (*string*) –
- **NiftyStorageType** (*integer*) –
- **OptionGroupName** (*string*) –
- **Port** (*integer*) –
- **PubliclyAccessible** (*boolean*) –
- **SourceDBInstanceIdentifier** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```

{
    'DBInstance': {
        'AccountingType': 'string',
        'AllocatedStorage': 123,
        'AutoMinorVersionUpgrade': True|False,
        'AvailabilityZone': 'string',
        'BackupRetentionPeriod': 123,
        'BinlogRetentionPeriod': 123,
        'CACertificateIdentifier': 'string',
        'DBInstanceClass': 'string',

```

(continues on next page)

(continued from previous page)

```

'DBInstanceIdentifier': 'string',
'DBInstanceStatus': 'string',
'DBName': 'string',
'DBParameterGroups': [
    {
        'DBParameterGroupName': 'string',
        'ParameterApplyStatus': 'string'
    },
],
'DBSecurityGroups': [
    {
        'DBSecurityGroupName': 'string',
        'Status': 'string'
    },
],
'Endpoint': {
    'Address': 'string',
    'NiftyPrivateAddress': 'string',
    'Port': 123
},
'Engine': 'string',
'EngineVersion': 'string',
'ExternalReplicationInfo': {
    'ExternalMasterAddress': 'string',
    'ExternalReplicationMessage': 'string',
    'ExternalReplicationStatus': 'string',
    'ReplicationAddresses': [
        'string',
    ],
    'ReplicationPrivateAddresses': [
        'string',
    ]
},
'InstanceCreateTime': datetime(2015, 1, 1),
'LatestRestorableTime': datetime(2015, 1, 1),
'LicenseModel': 'string',
'MasterUsername': 'string',
'MultiAZ': True|False,
'NextMonthAccountingType': 'string',
'NiftyMasterPrivateAddress': 'string',
'NiftyMultiAZType': 'string',
'NiftyNetworkId': 'string',
'NiftySlavePrivateAddress': 'string',
'NiftyStorageType': 123,
'OptionGroupMemberships': [
    {
        'OptionGroupName': 'string',
        'Status': 'string'
    },
],
'PendingModifiedValues': {
    'AllocatedStorage': 123,
    'BackupRetentionPeriod': 123,
    'DBInstanceClass': 'string',
    'DBInstanceIdentifier': 'string',
    'EngineVersion': 'string',
    'MasterUserPassword': 'string',

```

(continues on next page)

(continued from previous page)

```

        'MultiAZ': True|False,
        'Port': 123
    },
    'PreferredBackupWindow': 'string',
    'PreferredMaintenanceWindow': 'string',
    'PubliclyAccessible': True|False,
    'ReadReplicaDBInstanceIdentifiers': [
        'string',
    ],
    'ReadReplicaSourceDBInstanceIdentifier': 'string',
    'SecondaryAvailabilityZone': 'string',
    'StatusInfos': [
        {
            'Message': 'string',
            'Normal': True|False,
            'Status': 'string',
            'StatusType': 'string'
        },
    ],
    'VpcSecurityGroups': 'string'
},
'ResponseMetadata': {
    'RequestId': 'string'
}
}

```

### Response Structure

- *(dict)* –
  - **DBInstance** (*dict*) –
    - \* **AccountingType** (*string*) –
    - \* **AllocatedStorage** (*integer*) –
    - \* **AutoMinorVersionUpgrade** (*boolean*) –
    - \* **AvailabilityZone** (*string*) –
    - \* **BackupRetentionPeriod** (*integer*) –
    - \* **BinlogRetentionPeriod** (*integer*) –
    - \* **CACertificateIdentifier** (*string*) –
    - \* **DBInstanceClass** (*string*) –
    - \* **DBInstanceIdentifier** (*string*) –
    - \* **DBInstanceStatus** (*string*) –
    - \* **DBName** (*string*) –
    - \* **DBParameterGroups** (*list*) –
      - (*dict*) –
      - **DBParameterGroupName** (*string*) –
      - **ParameterApplyStatus** (*string*) –
    - \* **DBSecurityGroups** (*list*) –
      - (*dict*) –
      - **DBSecurityGroupName** (*string*) –
      - **Status** (*string*) –
    - \* **Endpoint** (*dict*) –
      - **Address** (*string*) –
      - **NiftyPrivateAddress** (*string*) –
      - **Port** (*integer*) –
    - \* **Engine** (*string*) –
    - \* **EngineVersion** (*string*) –
    - \* **ExternalReplicationInfo** (*dict*) –

- **ExternalMasterAddress** (*string*) –
- **ExternalReplicationMessage** (*string*) –
- **ExternalReplicationStatus** (*string*) –
- **ReplicationAddresses** (*list*) –
- (*string*) –
- **ReplicationPrivateAddresses** (*list*) –
- (*string*) –
- \* **InstanceCreateTime** (*datetime*) –
- \* **LatestRestorableTime** (*datetime*) –
- \* **LicenseModel** (*string*) –
- \* **MasterUsername** (*string*) –
- \* **MultiAZ** (*boolean*) –
- \* **NextMonthAccountingType** (*string*) –
- \* **NiftyMasterPrivateAddress** (*string*) –
- \* **NiftyMultiAZType** (*string*) –
- \* **NiftyNetworkId** (*string*) –
- \* **NiftySlavePrivateAddress** (*string*) –
- \* **NiftyStorageType** (*integer*) –
- \* **OptionGroupMemberships** (*list*) –
  - (*dict*) –
  - **OptionGroupName** (*string*) –
  - **Status** (*string*) –
- \* **PendingModifiedValues** (*dict*) –
  - **AllocatedStorage** (*integer*) –
  - **BackupRetentionPeriod** (*integer*) –
  - **DBInstanceClass** (*string*) –
  - **DBInstanceIdentifier** (*string*) –
  - **EngineVersion** (*string*) –
  - **MasterUserPassword** (*string*) –
  - **MultiAZ** (*boolean*) –
  - **Port** (*integer*) –
- \* **PreferredBackupWindow** (*string*) –
- \* **PreferredMaintenanceWindow** (*string*) –
- \* **PubliclyAccessible** (*boolean*) –
- \* **ReadReplicaDBInstanceIdentifiers** (*list*) –
  - (*string*) –
- \* **ReadReplicaSourceDBInstanceIdentifier** (*string*) –
- \* **SecondaryAvailabilityZone** (*string*) –
- \* **StatusInfos** (*list*) –
  - (*dict*) –
  - **Message** (*string*) –
  - **Normal** (*boolean*) –
  - **Status** (*string*) –
  - **StatusType** (*string*) –
- \* **VpcSecurityGroups** (*string*) –
- **ResponseMetadata** (*dict*) –
  - \* **RequestId** (*string*) –

*rdb* / Client / create\_db\_parameter\_group

## create\_db\_parameter\_group

`rdb.Client.create_db_parameter_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.create_db_parameter_group(  
    DBParameterGroupFamily='string',  
    DBParameterGroupName='string',  
    Description='string'  
)
```

#### Parameters

- **DBParameterGroupFamily** (*string*) – [REQUIRED]
- **DBParameterGroupName** (*string*) – [REQUIRED]
- **Description** (*string*) – [REQUIRED]

Return type dict

#### Returns

### Response Syntax

```
{  
    'DBParameterGroup': {  
        'DBParameterGroupFamily': 'string',  
        'DBParameterGroupName': 'string',  
        'Description': 'string'  
    },  
    'ResponseMetadata': {  
        'RequestId': 'string'  
    }  
}
```

### Response Structure

- (*dict*) –
  - **DBParameterGroup** (*dict*) –
    - \* **DBParameterGroupFamily** (*string*) –
    - \* **DBParameterGroupName** (*string*) –
    - \* **Description** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*rdb* / Client / create\_db\_security\_group

### create\_db\_security\_group

`rdb.Client.create_db_security_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.create_db_security_group(  
    DBSecurityGroupDescription='string',  
    DBSecurityGroupName='string',  
    NiftyAvailabilityZone='string'  
)
```

#### Parameters

- **DBSecurityGroupDescription** (*string*) – [REQUIRED]
- **DBSecurityGroupName** (*string*) – [REQUIRED]
- **NiftyAvailabilityZone** (*string*) – [REQUIRED]

Return type dict

## Returns

### Response Syntax

```
{
  'DBSecurityGroup': {
    'DBSecurityGroupDescription': 'string',
    'DBSecurityGroupName': 'string',
    'EC2SecurityGroups': [
      {
        'EC2SecurityGroupName': 'string',
        'EC2SecurityGroupOwnerId': 'string',
        'Status': 'string'
      },
    ],
    'IPRanges': [
      {
        'CIDRIP': 'string',
        'Status': 'string'
      },
    ],
    'NiftyAvailabilityZone': 'string',
    'OwnerId': 'string'
  },
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

### Response Structure

- (dict) –
  - **DBSecurityGroup** (dict) –
    - \* **DBSecurityGroupDescription** (string) –
    - \* **DBSecurityGroupName** (string) –
    - \* **EC2SecurityGroups** (list) –
      - (dict) –
      - **EC2SecurityGroupName** (string) –
      - **EC2SecurityGroupOwnerId** (string) –
      - **Status** (string) –
    - \* **IPRanges** (list) –
      - (dict) –
      - **CIDRIP** (string) –
      - **Status** (string) –
    - \* **NiftyAvailabilityZone** (string) –
    - \* **OwnerId** (string) –
  - **ResponseMetadata** (dict) –
    - \* **RequestId** (string) –

*rdb* / Client / create\_db\_snapshot

### create\_db\_snapshot

`rdb.Client.create_db_snapshot(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.create_db_snapshot(  
    DBInstanceIdentifier='string',  
    DBSnapshotIdentifier='string'  
)
```

**Parameters**

- **DBInstanceIdentifier** (*string*) – [REQUIRED]
- **DBSnapshotIdentifier** (*string*) – [REQUIRED]

**Return type** dict**Returns****Response Syntax**

```
{  
    'DBSnapshot': {  
        'AllocatedStorage': 123,  
        'AvailabilityZone': 'string',  
        'DBInstanceIdentifier': 'string',  
        'DBSnapshotIdentifier': 'string',  
        'Engine': 'string',  
        'EngineVersion': 'string',  
        'InstanceCreateTime': datetime(2015, 1, 1),  
        'LicenseModel': 'string',  
        'MasterUsername': 'string',  
        'OptionGroupName': 'string',  
        'Port': 123,  
        'SnapshotCreateTime': datetime(2015, 1, 1),  
        'SnapshotType': 'string',  
        'Status': 'string'  
    },  
    'ResponseMetadata': {  
        'RequestId': 'string'  
    }  
}
```

**Response Structure**

- (*dict*) –
  - **DBSnapshot** (*dict*) –
    - \* **AllocatedStorage** (*integer*) –
    - \* **AvailabilityZone** (*string*) –
    - \* **DBInstanceIdentifier** (*string*) –
    - \* **DBSnapshotIdentifier** (*string*) –
    - \* **Engine** (*string*) –
    - \* **EngineVersion** (*string*) –
    - \* **InstanceCreateTime** (*datetime*) –
    - \* **LicenseModel** (*string*) –
    - \* **MasterUsername** (*string*) –
    - \* **OptionGroupName** (*string*) –
    - \* **Port** (*integer*) –
    - \* **SnapshotCreateTime** (*datetime*) –
    - \* **SnapshotType** (*string*) –
    - \* **Status** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*rdb* / Client / create\_event\_subscription



## create\_event\_subscription

`rdb.Client.create_event_subscription(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```

response = client.create_event_subscription(
    Enabled=True|False,
    EventCategories=[
        'string',
    ],
    NiftyDescription='string',
    NiftyEmailAddresses=[
        'string',
    ],
    SourceIds=[
        'string',
    ],
    SourceType='db-instance'|'db-parameter-group'|'db-security-group'|'db-snapshot
    ↪',
    SubscriptionName='string'
)

```

### Parameters

- **Enabled** (*boolean*) –
- **EventCategories** (*list*) –  
– (*string*) –
- **NiftyDescription** (*string*) –
- **NiftyEmailAddresses** (*list*) – [REQUIRED]  
– (*string*) –
- **SourceIds** (*list*) –  
– (*string*) –
- **SourceType** (*string*) –
- **SubscriptionName** (*string*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```

{
    'EventSubscription': {
        'CustSubscriptionId': 'string',
        'Enabled': True|False,
        'EventCategoriesList': [
            'string',
        ],
        'NiftyDescription': 'string',
        'NiftyEmailAddressesList': [
            'string',
        ],
        'SourceIdsList': [
            'string',
        ],
        'SourceType': 'string',
        'Status': 'string',
        'SubscriptionCreationTime': 'string'
    }
}

```

(continues on next page)

(continued from previous page)

```

    },
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}

```

**Response Structure**

- *(dict)* –
  - **EventSubscription** (*dict*) –
    - \* **CustSubscriptionId** (*string*) –
    - \* **Enabled** (*boolean*) –
    - \* **EventCategoriesList** (*list*) –
      - (*string*) –
    - \* **NiftyDescription** (*string*) –
    - \* **NiftyEmailAddressesList** (*list*) –
      - (*string*) –
    - \* **SourceIdsList** (*list*) –
      - (*string*) –
    - \* **SourceType** (*string*) –
    - \* **Status** (*string*) –
    - \* **SubscriptionCreationTime** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*rdb* / Client / delete\_db\_instance**delete\_db\_instance**`rdb.Client.delete_db_instance(**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```

response = client.delete_db_instance(
    DBInstanceIdentifier='string',
    FinalDBSnapshotIdentifier='string',
    SkipFinalSnapshot=True|False
)

```

**Parameters**

- **DBInstanceIdentifier** (*string*) – [REQUIRED]
- **FinalDBSnapshotIdentifier** (*string*) –
- **SkipFinalSnapshot** (*boolean*) –

**Return type** dict**Returns****Response Syntax**

```

{
    'DBInstance': {
        'AccountingType': 'string',
        'AllocatedStorage': 123,
        'AutoMinorVersionUpgrade': True|False,
        'AvailabilityZone': 'string',

```

(continues on next page)

(continued from previous page)

```

'BackupRetentionPeriod': 123,
'BinlogRetentionPeriod': 123,
'CACertificateIdentifier': 'string',
'DBInstanceClass': 'string',
'DBInstanceIdentifier': 'string',
'DBInstanceStatus': 'string',
'DBName': 'string',
'DBParameterGroups': [
    {
        'DBParameterGroupName': 'string',
        'ParameterApplyStatus': 'string'
    },
],
'DBSecurityGroups': [
    {
        'DBSecurityGroupName': 'string',
        'Status': 'string'
    },
],
'Endpoint': {
    'Address': 'string',
    'NiftyPrivateAddress': 'string',
    'Port': 123
},
'Engine': 'string',
'EngineVersion': 'string',
'ExternalReplicationInfo': {
    'ExternalMasterAddress': 'string',
    'ExternalReplicationMessage': 'string',
    'ExternalReplicationStatus': 'string',
    'ReplicationAddresses': [
        'string',
    ],
    'ReplicationPrivateAddresses': [
        'string',
    ]
},
'InstanceCreateTime': datetime(2015, 1, 1),
'LatestRestorableTime': datetime(2015, 1, 1),
'LicenseModel': 'string',
'MasterUsername': 'string',
'MultiAZ': True|False,
'NextMonthAccountingType': 'string',
'NiftyMasterPrivateAddress': 'string',
'NiftyMultiAZType': 'string',
'NiftyNetworkId': 'string',
'NiftySlavePrivateAddress': 'string',
'NiftyStorageType': 123,
'OptionGroupMemberships': [
    {
        'OptionGroupName': 'string',
        'Status': 'string'
    },
],
'PendingModifiedValues': {
    'AllocatedStorage': 123,
    'BackupRetentionPeriod': 123,

```

(continues on next page)

(continued from previous page)

```

        'DBInstanceClass': 'string',
        'DBInstanceIdentifier': 'string',
        'EngineVersion': 'string',
        'MasterUserPassword': 'string',
        'MultiAZ': True|False,
        'Port': 123
    },
    'PreferredBackupWindow': 'string',
    'PreferredMaintenanceWindow': 'string',
    'PubliclyAccessible': True|False,
    'ReadReplicaDBInstanceIdentifiers': [
        'string',
    ],
    'ReadReplicaSourceDBInstanceIdentifier': 'string',
    'SecondaryAvailabilityZone': 'string',
    'StatusInfos': [
        {
            'Message': 'string',
            'Normal': True|False,
            'Status': 'string',
            'StatusType': 'string'
        },
    ],
    'VpcSecurityGroups': 'string'
},
'ResponseMetadata': {
    'RequestId': 'string'
}
}

```

### Response Structure

- (dict) –
  - DBInstance (dict) –
    - \* AccountingType (string) –
    - \* AllocatedStorage (integer) –
    - \* AutoMinorVersionUpgrade (boolean) –
    - \* AvailabilityZone (string) –
    - \* BackupRetentionPeriod (integer) –
    - \* BinlogRetentionPeriod (integer) –
    - \* CACertificateIdentifier (string) –
    - \* DBInstanceClass (string) –
    - \* DBInstanceIdentifier (string) –
    - \* DBInstanceStatus (string) –
    - \* DBName (string) –
    - \* DBParameterGroups (list) –
      - (dict) –
      - DBParameterGroupName (string) –
      - ParameterApplyStatus (string) –
    - \* DBSecurityGroups (list) –
      - (dict) –
      - DBSecurityGroupName (string) –
      - Status (string) –
    - \* Endpoint (dict) –
      - Address (string) –
      - NiftyPrivateAddress (string) –

- **Port** (*integer*) –
- \* **Engine** (*string*) –
- \* **EngineVersion** (*string*) –
- \* **ExternalReplicationInfo** (*dict*) –
  - **ExternalMasterAddress** (*string*) –
  - **ExternalReplicationMessage** (*string*) –
  - **ExternalReplicationStatus** (*string*) –
  - **ReplicationAddresses** (*list*) –
  - (*string*) –
  - **ReplicationPrivateAddresses** (*list*) –
  - (*string*) –
- \* **InstanceCreateTime** (*datetime*) –
- \* **LatestRestorableTime** (*datetime*) –
- \* **LicenseModel** (*string*) –
- \* **MasterUsername** (*string*) –
- \* **MultiAZ** (*boolean*) –
- \* **NextMonthAccountingType** (*string*) –
- \* **NiftyMasterPrivateAddress** (*string*) –
- \* **NiftyMultiAZType** (*string*) –
- \* **NiftyNetworkId** (*string*) –
- \* **NiftySlavePrivateAddress** (*string*) –
- \* **NiftyStorageType** (*integer*) –
- \* **OptionGroupMemberships** (*list*) –
  - (*dict*) –
  - **OptionGroupName** (*string*) –
  - **Status** (*string*) –
- \* **PendingModifiedValues** (*dict*) –
  - **AllocatedStorage** (*integer*) –
  - **BackupRetentionPeriod** (*integer*) –
  - **DBInstanceClass** (*string*) –
  - **DBInstanceIdentifier** (*string*) –
  - **EngineVersion** (*string*) –
  - **MasterUserPassword** (*string*) –
  - **MultiAZ** (*boolean*) –
  - **Port** (*integer*) –
- \* **PreferredBackupWindow** (*string*) –
- \* **PreferredMaintenanceWindow** (*string*) –
- \* **PubliclyAccessible** (*boolean*) –
- \* **ReadReplicaDBInstanceIdentifiers** (*list*) –
  - (*string*) –
- \* **ReadReplicaSourceDBInstanceIdentifier** (*string*) –
- \* **SecondaryAvailabilityZone** (*string*) –
- \* **StatusInfos** (*list*) –
  - (*dict*) –
  - **Message** (*string*) –
  - **Normal** (*boolean*) –
  - **Status** (*string*) –
  - **StatusType** (*string*) –
- \* **VpcSecurityGroups** (*string*) –
- **ResponseMetadata** (*dict*) –
  - \* **RequestId** (*string*) –

*rdb* / Client / delete\_db\_parameter\_group

## delete\_db\_parameter\_group

`rdb.Client.delete_db_parameter_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_db_parameter_group(
    DBParameterGroupName='string'
)
```

**Parameters** `DBParameterGroupName` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

### Response Structure

- (*dict*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*rdb* / Client / delete\_db\_security\_group

## delete\_db\_security\_group

`rdb.Client.delete_db_security_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_db_security_group(
    DBSecurityGroupName='string'
)
```

**Parameters** `DBSecurityGroupName` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

### Response Structure

- (*dict*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*rdb* / Client / delete\_db\_snapshot

## delete\_db\_snapshot

`rdb.Client.delete_db_snapshot(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_db_snapshot(
    DBSnapshotIdentifier='string'
)
```

**Parameters** `DBSnapshotIdentifier` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
    'DBSnapshot': {
        'AllocatedStorage': 123,
        'AvailabilityZone': 'string',
        'DBInstanceIdentifier': 'string',
        'DBSnapshotIdentifier': 'string',
        'Engine': 'string',
        'EngineVersion': 'string',
        'InstanceCreateTime': datetime(2015, 1, 1),
        'LicenseModel': 'string',
        'MasterUsername': 'string',
        'OptionGroupName': 'string',
        'Port': 123,
        'SnapshotCreateTime': datetime(2015, 1, 1),
        'SnapshotType': 'string',
        'Status': 'string'
    },
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

### Response Structure

- (*dict*) –
  - **DBSnapshot** (*dict*) –
    - \* **AllocatedStorage** (*integer*) –
    - \* **AvailabilityZone** (*string*) –
    - \* **DBInstanceIdentifier** (*string*) –
    - \* **DBSnapshotIdentifier** (*string*) –
    - \* **Engine** (*string*) –
    - \* **EngineVersion** (*string*) –
    - \* **InstanceCreateTime** (*datetime*) –
    - \* **LicenseModel** (*string*) –
    - \* **MasterUsername** (*string*) –
    - \* **OptionGroupName** (*string*) –
    - \* **Port** (*integer*) –
    - \* **SnapshotCreateTime** (*datetime*) –
    - \* **SnapshotType** (*string*) –
    - \* **Status** (*string*) –
  - **ResponseMetadata** (*dict*) –

\* **RequestId** (*string*) –

*rdb* / Client / delete\_event\_subscription

## delete\_event\_subscription

`rdb.Client.delete_event_subscription(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_event_subscription(
    SubscriptionName='string'
)
```

**Parameters** **SubscriptionName** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
    'EventSubscription': {
        'CustSubscriptionId': 'string',
        'Enabled': True|False,
        'EventCategoriesList': [
            'string',
        ],
        'NiftyDescription': 'string',
        'NiftyEmailAddressesList': [
            'string',
        ],
        'SourceIdsList': [
            'string',
        ],
        'SourceType': 'string',
        'Status': 'string',
        'SubscriptionCreationTime': 'string'
    },
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

### Response Structure

- (*dict*) –
  - **EventSubscription** (*dict*) –
    - \* **CustSubscriptionId** (*string*) –
    - \* **Enabled** (*boolean*) –
    - \* **EventCategoriesList** (*list*) –
      - (*string*) –
    - \* **NiftyDescription** (*string*) –
    - \* **NiftyEmailAddressesList** (*list*) –
      - (*string*) –
    - \* **SourceIdsList** (*list*) –
      - (*string*) –
    - \* **SourceType** (*string*) –



- \* **Status** (*string*) –
- \* **SubscriptionCreationTime** (*string*) –
- **ResponseMetadata** (*dict*) –
- \* **RequestId** (*string*) –

*rd*b / Client / describe\_certificates

## describe\_certificates

`rd`b.Client.**describe\_certificates** (\*\**kwargs*)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_certificates(
    CertificateIdentifier='string',
    Filter='string',
    FilterName='string',
    FilterValue='string',
    Filters=[
        'string',
    ],
    Marker='string',
    MaxRecords=123
)
```

### Parameters

- **CertificateIdentifier** (*string*) –
- **Filter** (*string*) –
- **FilterName** (*string*) –
- **FilterValue** (*string*) –
- **Filters** (*list*) –
  - (*string*) –
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –

**Return type** dict

**Returns**

### Response Syntax

```
{
    'Certificates': [
        {
            'CertificateIdentifier': 'string',
            'CertificateType': 'string',
            'Thumbprint': 'string',
            'ValidFrom': datetime(2015, 1, 1),
            'ValidTill': datetime(2015, 1, 1)
        },
    ],
    'Marker': 'string',
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

### Response Structure

- *(dict)* –
  - **Certificates** (*list*) –
    - \* *(dict)* –
      - **CertificateIdentifier** (*string*) –
      - **CertificateType** (*string*) –
      - **Thumbprint** (*string*) –
      - **ValidFrom** (*datetime*) –
      - **ValidTill** (*datetime*) –
  - **Marker** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*rdb* / Client / describe\_db\_engine\_versions

## describe\_db\_engine\_versions

`rdb.Client.describe_db_engine_versions(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_db_engine_versions(
    DBParameterGroupFamily='mysql5.6'|'mysql5.7'|'postgres9.3'|'postgres9.6'|
    ↪ 'postgres11'|'postgres13',
    DefaultOnly=True|False,
    Engine='MySQL'|'postgres',
    EngineVersion='string',
    IncludeAll=True|False,
    ListSupportedCharacterSets=True|False,
    Marker='string',
    MaxRecords=123
)
```

### Parameters

- **DBParameterGroupFamily** (*string*) –
- **DefaultOnly** (*boolean*) –
- **Engine** (*string*) –
- **EngineVersion** (*string*) –
- **IncludeAll** (*boolean*) –
- **ListSupportedCharacterSets** (*boolean*) –
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –

**Return type** dict

**Returns**

### Response Syntax

```
{
  'DBEngineVersions': [
    {
      'DBEngineDescription': 'string',
      'DBEngineVersionDescription': 'string',
      'DBParameterGroupFamily': 'string',
      'Engine': 'string',
      'EngineVersion': 'string',
      'Status': 'string',
```

(continues on next page)

(continued from previous page)

```

        'ValidUpgradeTarget': [
            {
                'AutoUpgrade': True|False,
                'DBParameterGroupFamily': 'string',
                'Description': 'string',
                'Engine': 'string',
                'EngineVersion': 'string',
                'IsMajorVersionUpgrade': True|False
            },
        ],
    },
    'Marker': 'string',
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}

```

**Response Structure**

- (dict) –
  - **DBEngineVersions** (list) –
    - \* (dict) –
      - **DBEngineDescription** (string) –
      - **DBEngineVersionDescription** (string) –
      - **DBParameterGroupFamily** (string) –
      - **Engine** (string) –
      - **EngineVersion** (string) –
      - **Status** (string) –
      - **ValidUpgradeTarget** (list) –
      - (dict) –
        - **AutoUpgrade** (boolean) –
        - **DBParameterGroupFamily** (string) –
        - **Description** (string) –
        - **Engine** (string) –
        - **EngineVersion** (string) –
        - **IsMajorVersionUpgrade** (boolean) –
  - **Marker** (string) –
  - **ResponseMetadata** (dict) –
    - \* **RequestId** (string) –

*rdb* / Client / describe\_db\_instances**describe\_db\_instances**`rdb.Client.describe_db_instances(**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```

response = client.describe_db_instances(
    DBInstanceIdentifier='string',
    Filter='string',
    FilterName='string',
    FilterValue='string',

```

(continues on next page)

(continued from previous page)

```

Filters=[
    'string',
],
Marker='string',
MaxRecords=123
)

```

**Parameters**

- **DBInstanceIdentifier** (*string*) –
- **Filter** (*string*) –
- **FilterName** (*string*) –
- **FilterValue** (*string*) –
- **Filters** (*list*) –
  - (*string*) –
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –

**Return type** dict**Returns****Response Syntax**

```

{
    'DBInstances': [
        {
            'AccountingType': 'string',
            'AllocatedStorage': 123,
            'AutoMinorVersionUpgrade': True|False,
            'AvailabilityZone': 'string',
            'BackupRetentionPeriod': 123,
            'BinlogRetentionPeriod': 123,
            'CACertificateIdentifier': 'string',
            'DBInstanceClass': 'string',
            'DBInstanceIdentifier': 'string',
            'DBInstanceStatus': 'string',
            'DBName': 'string',
            'DBParameterGroups': [
                {
                    'DBParameterGroupName': 'string',
                    'ParameterApplyStatus': 'string'
                },
            ],
            'DBSecurityGroups': [
                {
                    'DBSecurityGroupName': 'string',
                    'Status': 'string'
                },
            ],
            'Endpoint': {
                'Address': 'string',
                'NiftyPrivateAddress': 'string',
                'Port': 123
            },
            'Engine': 'string',
            'EngineVersion': 'string',
            'ExternalReplicationInfo': {
                'ExternalMasterAddress': 'string',

```

(continues on next page)

(continued from previous page)

```

        'ExternalReplicationMessage': 'string',
        'ExternalReplicationStatus': 'string',
        'ReplicationAddresses': [
            'string',
        ],
        'ReplicationPrivateAddresses': [
            'string',
        ]
    },
    'InstanceCreateTime': datetime(2015, 1, 1),
    'LatestRestorableTime': datetime(2015, 1, 1),
    'LicenseModel': 'string',
    'MasterUsername': 'string',
    'MultiAZ': True|False,
    'NextMonthAccountingType': 'string',
    'NiftyMasterPrivateAddress': 'string',
    'NiftyMultiAZType': 'string',
    'NiftyNetworkId': 'string',
    'NiftySlavePrivateAddress': 'string',
    'NiftyStorageType': 123,
    'OptionGroupMemberships': [
        {
            'OptionGroupName': 'string',
            'Status': 'string'
        },
    ],
    'PendingModifiedValues': {
        'AllocatedStorage': 123,
        'BackupRetentionPeriod': 123,
        'DBInstanceClass': 'string',
        'DBInstanceIdentifier': 'string',
        'EngineVersion': 'string',
        'MasterUserPassword': 'string',
        'MultiAZ': True|False,
        'Port': 123
    },
    'PreferredBackupWindow': 'string',
    'PreferredMaintenanceWindow': 'string',
    'PubliclyAccessible': True|False,
    'ReadReplicaDBInstanceIdentifiers': [
        'string',
    ],
    'ReadReplicaSourceDBInstanceIdentifier': 'string',
    'SecondaryAvailabilityZone': 'string',
    'StatusInfos': [
        {
            'Message': 'string',
            'Normal': True|False,
            'Status': 'string',
            'StatusType': 'string'
        },
    ],
    'VpcSecurityGroups': 'string'
},
'Marker': 'string',
'ResponseMetadata': {

```

(continues on next page)

(continued from previous page)

```

    'RequestId': 'string'
  }
}

```

**Response Structure**

- (dict) –
  - DBInstances (list) –
    - \* (dict) –
      - AccountingType (string) –
      - AllocatedStorage (integer) –
      - AutoMinorVersionUpgrade (boolean) –
      - AvailabilityZone (string) –
      - BackupRetentionPeriod (integer) –
      - BinlogRetentionPeriod (integer) –
      - CACertificateIdentifier (string) –
      - DBInstanceClass (string) –
      - DBInstanceIdentifier (string) –
      - DBInstanceStatus (string) –
      - DBName (string) –
      - DBParameterGroups (list) –
      - (dict) –
      - DBParameterGroupName (string) –
      - ParameterApplyStatus (string) –
      - DBSecurityGroups (list) –
      - (dict) –
      - DBSecurityGroupName (string) –
      - Status (string) –
      - Endpoint (dict) –
      - Address (string) –
      - NiftyPrivateAddress (string) –
      - Port (integer) –
      - Engine (string) –
      - EngineVersion (string) –
      - ExternalReplicationInfo (dict) –
      - ExternalMasterAddress (string) –
      - ExternalReplicationMessage (string) –
      - ExternalReplicationStatus (string) –
      - ReplicationAddresses (list) –
      - (string) –
      - ReplicationPrivateAddresses (list) –
      - (string) –
      - InstanceCreateTime (datetime) –
      - LatestRestorableTime (datetime) –
      - LicenseModel (string) –
      - MasterUsername (string) –
      - MultiAZ (boolean) –
      - NextMonthAccountingType (string) –
      - NiftyMasterPrivateAddress (string) –
      - NiftyMultiAZType (string) –
      - NiftyNetworkId (string) –
      - NiftySlavePrivateAddress (string) –
      - NiftyStorageType (integer) –
      - OptionGroupMemberships (list) –

- *(dict)* –
- **OptionGroupName** (*string*) –
- **Status** (*string*) –
- **PendingModifiedValues** (*dict*) –
- **AllocatedStorage** (*integer*) –
- **BackupRetentionPeriod** (*integer*) –
- **DBInstanceClass** (*string*) –
- **DBInstanceIdentifier** (*string*) –
- **EngineVersion** (*string*) –
- **MasterUserPassword** (*string*) –
- **MultiAZ** (*boolean*) –
- **Port** (*integer*) –
- **PreferredBackupWindow** (*string*) –
- **PreferredMaintenanceWindow** (*string*) –
- **PubliclyAccessible** (*boolean*) –
- **ReadReplicaDBInstanceIdentifiers** (*list*) –
- (*string*) –
- **ReadReplicaSourceDBInstanceIdentifier** (*string*) –
- **SecondaryAvailabilityZone** (*string*) –
- **StatusInfos** (*list*) –
- (*dict*) –
- **Message** (*string*) –
- **Normal** (*boolean*) –
- **Status** (*string*) –
- **StatusType** (*string*) –
- **VpcSecurityGroups** (*string*) –
- **Marker** (*string*) –
- **ResponseMetadata** (*dict*) –
  - \* **RequestId** (*string*) –

*rdb* / Client / describe\_db\_log\_files

## describe\_db\_log\_files

`rdb.Client.describe_db_log_files(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_db_log_files(
    DBInstanceIdentifier='string',
    FileLastWritten=123,
    FileSize=123,
    FilenameContains='string',
    Marker='string',
    MaxRecords=123
)
```

### Parameters

- **DBInstanceIdentifier** (*string*) – [REQUIRED]
- **FileLastWritten** (*integer*) –
- **FileSize** (*integer*) –
- **FilenameContains** (*string*) –
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'DescribeDBLogFiles': [
    {
      'LastWritten': 123,
      'LogFileName': 'string',
      'Size': 123
    },
  ],
  'Marker': 'string',
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

#### Response Structure

- (dict) –
  - **DescribeDBLogFiles** (list) –
    - \* (dict) –
      - **LastWritten** (integer) –
      - **LogFileName** (string) –
      - **Size** (integer) –
  - **Marker** (string) –
  - **ResponseMetadata** (dict) –
    - \* **RequestId** (string) –

*rdb* / Client / describe\_db\_parameter\_groups

### describe\_db\_parameter\_groups

`rdb.Client.describe_db_parameter_groups(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.describe_db_parameter_groups(
    DBParameterGroupName='string',
    Filter='string',
    FilterName='string',
    FilterValue='string',
    Filters=[
        'string',
    ],
    Marker='string',
    MaxRecords=123
)
```

#### Parameters

- **DBParameterGroupName** (string) –
- **Filter** (string) –
- **FilterName** (string) –
- **FilterValue** (string) –
- **Filters** (list) –



- (string) –
- **Marker** (string) –
- **MaxRecords** (integer) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
  'DBParameterGroups': [
    {
      'DBParameterGroupFamily': 'string',
      'DBParameterGroupName': 'string',
      'Description': 'string'
    },
  ],
  'Marker': 'string',
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

#### Response Structure

- (dict) –
  - **DBParameterGroups** (list) –
    - \* (dict) –
      - **DBParameterGroupFamily** (string) –
      - **DBParameterGroupName** (string) –
      - **Description** (string) –
  - **Marker** (string) –
  - **ResponseMetadata** (dict) –
    - \* **RequestId** (string) –

*rdb* / Client / describe\_db\_parameters

### describe\_db\_parameters

`rdb.Client.describe_db_parameters(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.describe_db_parameters(
    DBParameterGroupName='string',
    Marker='string',
    MaxRecords=123,
    Source='user'|'system'|'engine-default'
)
```

#### Parameters

- **DBParameterGroupName** (string) – [REQUIRED]
- **Marker** (string) –
- **MaxRecords** (integer) –
- **Source** (string) –

**Return type** dict

**Returns**

### Response Syntax

```
{
  'Marker': 'string',
  'Parameters': [
    {
      'AllowedValues': 'string',
      'ApplyMethod': 'string',
      'ApplyType': 'string',
      'DataType': 'string',
      'Description': 'string',
      'IsModifiable': True|False,
      'MinimumEngineVersion': 'string',
      'ParameterName': 'string',
      'ParameterValue': 'string',
      'Source': 'string'
    },
  ],
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

### Response Structure

- (dict) –
  - **Marker** (string) –
  - **Parameters** (list) –
    - \* (dict) –
      - **AllowedValues** (string) –
      - **ApplyMethod** (string) –
      - **ApplyType** (string) –
      - **DataType** (string) –
      - **Description** (string) –
      - **IsModifiable** (boolean) –
      - **MinimumEngineVersion** (string) –
      - **ParameterName** (string) –
      - **ParameterValue** (string) –
      - **Source** (string) –
  - **ResponseMetadata** (dict) –
    - \* **RequestId** (string) –

*rdb* / Client / describe\_db\_security\_groups

### describe\_db\_security\_groups

`rdb.Client.describe_db_security_groups(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_db_security_groups(
    DBSecurityGroupName='string',
    Filter='string',
    FilterName='string',
    FilterValue='string',
    Filters=[
```

(continues on next page)

(continued from previous page)

```

        'string',
    ],
    Marker='string',
    MaxRecords=123
)

```

**Parameters**

- **DBSecurityGroupName** (*string*) –
- **Filter** (*string*) –
- **FilterName** (*string*) –
- **FilterValue** (*string*) –
- **Filters** (*list*) –
  - (*string*) –
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –

**Return type** dict**Returns****Response Syntax**

```

{
    'DBSecurityGroups': [
        {
            'DBSecurityGroupDescription': 'string',
            'DBSecurityGroupName': 'string',
            'EC2SecurityGroups': [
                {
                    'EC2SecurityGroupName': 'string',
                    'EC2SecurityGroupOwnerId': 'string',
                    'Status': 'string'
                },
            ],
            'IPRanges': [
                {
                    'CIDRIP': 'string',
                    'Status': 'string'
                },
            ],
            'NiftyAvailabilityZone': 'string',
            'OwnerId': 'string'
        },
    ],
    'Marker': 'string',
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}

```

**Response Structure**

- (*dict*) –
  - **DBSecurityGroups** (*list*) –
    - \* (*dict*) –
      - **DBSecurityGroupDescription** (*string*) –
      - **DBSecurityGroupName** (*string*) –
      - **EC2SecurityGroups** (*list*) –
      - (*dict*) –

- **EC2SecurityGroupName** (*string*) –
- **EC2SecurityGroupOwnerId** (*string*) –
- **Status** (*string*) –
- **IPRanges** (*list*) –
- (*dict*) –
- **CIDRIP** (*string*) –
- **Status** (*string*) –
- **NiftyAvailabilityZone** (*string*) –
- **OwnerId** (*string*) –
- **Marker** (*string*) –
- **ResponseMetadata** (*dict*) –
  - \* **RequestId** (*string*) –

*rdb* / Client / describe\_db\_snapshots

## describe\_db\_snapshots

`rdb.Client.describe_db_snapshots(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_db_snapshots(
    DBInstanceIdentifier='string',
    DBSnapshotIdentifier='string',
    Filter='string',
    FilterName='string',
    FilterValue='string',
    Filters=[
        'string',
    ],
    Marker='string',
    MaxRecords=123,
    SnapshotType='automated'|'manual'
)
```

### Parameters

- **DBInstanceIdentifier** (*string*) –
- **DBSnapshotIdentifier** (*string*) –
- **Filter** (*string*) –
- **FilterName** (*string*) –
- **FilterValue** (*string*) –
- **Filters** (*list*) –
  - (*string*) –
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –
- **SnapshotType** (*string*) –

**Return type** dict

### Returns

#### Response Syntax

```
{
    'DBSnapshots': [
        {
            'AllocatedStorage': 123,
```

(continues on next page)

(continued from previous page)

```

        'AvailabilityZone': 'string',
        'DBInstanceIdentifier': 'string',
        'DBSnapshotIdentifier': 'string',
        'Engine': 'string',
        'EngineVersion': 'string',
        'InstanceCreateTime': datetime(2015, 1, 1),
        'LicenseModel': 'string',
        'MasterUsername': 'string',
        'OptionGroupName': 'string',
        'Port': 123,
        'SnapshotCreateTime': datetime(2015, 1, 1),
        'SnapshotType': 'string',
        'Status': 'string'
    },
],
'Marker': 'string',
'ResponseMetadata': {
    'RequestId': 'string'
}
}

```

**Response Structure**

- *(dict)* –
  - **DBSnapshots** (*list*) –
    - \* *(dict)* –
      - **AllocatedStorage** (*integer*) –
      - **AvailabilityZone** (*string*) –
      - **DBInstanceIdentifier** (*string*) –
      - **DBSnapshotIdentifier** (*string*) –
      - **Engine** (*string*) –
      - **EngineVersion** (*string*) –
      - **InstanceCreateTime** (*datetime*) –
      - **LicenseModel** (*string*) –
      - **MasterUsername** (*string*) –
      - **OptionGroupName** (*string*) –
      - **Port** (*integer*) –
      - **SnapshotCreateTime** (*datetime*) –
      - **SnapshotType** (*string*) –
      - **Status** (*string*) –
  - **Marker** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*rd*b / Client / describe\_engine\_default\_parameters**describe\_engine\_default\_parameters***rd*b.Client.describe\_engine\_default\_parameters (\*\*kwargs)See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```

response = client.describe_engine_default_parameters(
    DBParameterGroupFamily='string',

```

(continues on next page)

(continued from previous page)

```

Marker='string',
MaxRecords=123
)

```

**Parameters**

- **DBParameterGroupFamily** (*string*) – [REQUIRED]
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –

**Return type** dict**Returns****Response Syntax**

```

{
    'EngineDefaults': {
        'DBParameterGroupFamily': 'string',
        'Marker': 'string',
        'Parameters': [
            {
                'AllowedValues': 'string',
                'ApplyMethod': 'string',
                'ApplyType': 'string',
                'DataType': 'string',
                'Description': 'string',
                'IsModifiable': True|False,
                'MinimumEngineVersion': 'string',
                'ParameterName': 'string',
                'ParameterValue': 'string',
                'Source': 'string'
            },
            ...
        ],
    },
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}

```

**Response Structure**

- (*dict*) –
  - **EngineDefaults** (*dict*) –
    - \* **DBParameterGroupFamily** (*string*) –
    - \* **Marker** (*string*) –
    - \* **Parameters** (*list*) –
      - (*dict*) –
      - **AllowedValues** (*string*) –
      - **ApplyMethod** (*string*) –
      - **ApplyType** (*string*) –
      - **DataType** (*string*) –
      - **Description** (*string*) –
      - **IsModifiable** (*boolean*) –
      - **MinimumEngineVersion** (*string*) –
      - **ParameterName** (*string*) –
      - **ParameterValue** (*string*) –
      - **Source** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*rdb* / Client / describe\_event\_categories

## describe\_event\_categories

`rdb.Client.describe_event_categories(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_event_categories(
    SourceType='db-instance'|'db-parameter-group'|'db-security-group'|'db-snapshot
    ↪ ')
)
```

**Parameters** `SourceType` (*string*) –

**Return type** dict

**Returns**

### Response Syntax

```
{
    'EventCategoriesMapList': [
        {
            'EventCategories': [
                'string',
            ],
            'SourceType': 'string'
        },
    ],
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

### Response Structure

- (*dict*) –
  - **EventCategoriesMapList** (*list*) –
    - \* (*dict*) –
      - **EventCategories** (*list*) –
        - (*string*) –
      - **SourceType** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*rdb* / Client / describe\_event\_subscriptions

## describe\_event\_subscriptions

`rdb.Client.describe_event_subscriptions(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_event_subscriptions(
    Marker='string',
    MaxRecords=123,
```

(continues on next page)

(continued from previous page)

```

    NiftySortKey='SubscriptionName'|'SourceType'|'Enabled'|
    ↪'SubscriptionCreationTime',
    NiftyFilters={
        'ListOfRequestFilter': [
            {
                'FilterName': 'SubscriptionName'|'SourceType'|'Enabled'|
                ↪'NiftyDescription',
                'FilterValue': 'string'
            },
        ]
    },
    NiftySortDesc=123,
    SubscriptionName='string'
)

```

**Parameters**

- **Marker** (*string*) –
- **MaxRecords** (*integer*) –
- **NiftySortKey** (*string*) –
- **NiftyFilters** (*dict*) –
  - **ListOfRequestFilter** (*list*) –
    - \* (*dict*) –
      - **FilterName** (*string*) –
      - **FilterValue** (*string*) –
- **NiftySortDesc** (*integer*) –
- **SubscriptionName** (*string*) –

**Return type** dict**Returns****Response Syntax**

```

{
    'EventSubscriptionsList': [
        {
            'CustSubscriptionId': 'string',
            'Enabled': True|False,
            'EventCategoriesList': [
                'string',
            ],
            'NiftyDescription': 'string',
            'NiftyEmailAddressesList': [
                'string',
            ],
            'SourceIdsList': [
                'string',
            ],
            'SourceType': 'string',
            'Status': 'string',
            'SubscriptionCreationTime': 'string'
        },
    ],
    'Marker': 'string',
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}

```



**Response Structure**

- (*dict*) –
  - **EventSubscriptionsList** (*list*) –
    - \* (*dict*) –
      - **CustSubscriptionId** (*string*) –
      - **Enabled** (*boolean*) –
      - **EventCategoriesList** (*list*) –
      - (*string*) –
      - **NiftyDescription** (*string*) –
      - **NiftyEmailAddressesList** (*list*) –
      - (*string*) –
      - **SourceIdsList** (*list*) –
      - (*string*) –
      - **SourceType** (*string*) –
      - **Status** (*string*) –
      - **SubscriptionCreationTime** (*string*) –
  - **Marker** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*rdb* / Client / describe\_events

**describe\_events**

`rdb.Client.describe_events(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.describe_events(
    Duration=123,
    EndTime=datetime(2015, 1, 1),
    EventCategories=[
        'string',
    ],
    Marker='string',
    MaxRecords=123,
    SourceIdentifier='string',
    SourceType='db-instance'|'db-parameter-group'|'db-security-group'|'db-snapshot
    ↪',
    StartTime=datetime(2015, 1, 1)
)
```

**Parameters**

- **Duration** (*integer*) –
- **EndTime** (*datetime*) –
- **EventCategories** (*list*) –
  - (*string*) –
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –
- **SourceIdentifier** (*string*) –
- **SourceType** (*string*) –
- **StartTime** (*datetime*) –

**Return type** dict

**Returns**

### Response Syntax

```
{
  'Events': [
    {
      'Date': datetime(2015, 1, 1),
      'EventCategories': [
        'string',
      ],
      'Message': 'string',
      'SourceIdentifier': 'string',
      'SourceType': 'string'
    },
  ],
  'Marker': 'string',
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

### Response Structure

- (dict) –
  - **Events** (list) –
    - \* (dict) –
      - **Date** (datetime) –
      - **EventCategories** (list) –
      - (string) –
      - **Message** (string) –
      - **SourceIdentifier** (string) –
      - **SourceType** (string) –
  - **Marker** (string) –
  - **ResponseMetadata** (dict) –
    - \* **RequestId** (string) –

*rdb* / Client / describe\_orderable\_db\_instance\_options

### describe\_orderable\_db\_instance\_options

`rdb.Client.describe_orderable_db_instance_options(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_orderable_db_instance_options(
    DBInstanceClass='db.mini'|'db.e-small'|'db.small'|'db.e-small2'|'db.small2'|
    ↳ 'db.e-small4'|'db.small4'|'db.e-small8'|'db.small8'|'db.e-small16'|'db.small16'|
    ↳ 'db.e-medium'|'db.medium'|'db.e-medium4'|'db.medium4'|'db.e-medium8'|'db.medium8'
    ↳ '| 'db.e-medium16'|'db.medium16'|'db.e-medium24'|'db.medium24'|'db.e-large'|'db.'
    ↳ 'large'|'db.e-large8'|'db.large8'|'db.e-large16'|'db.large16'|'db.e-large24'|'db.'
    ↳ 'large24'|'db.e-large32'|'db.large32'|'db.e-extra-large8'|'db.extra-large8'|'db.'
    ↳ 'e-extra-large16'|'db.extra-large16'|'db.e-extra-large24'|'db.extra-large24'|'db.'
    ↳ 'e-extra-large32'|'db.extra-large32'|'db.e-extra-large48'|'db.extra-large48'|'db.'
    ↳ 'e-double-large16'|'db.double-large16'|'db.e-double-large24'|'db.double-large24'|
    ↳ 'db.e-double-large32'|'db.double-large32'|'db.e-double-large48'|'db.double-'
    ↳ 'large48'|'db.e-double-large64'|'db.double-large64'|'db.e-double-large96'|'db.'
    ↳ 'double-large96'|'db.triple-large32'|'db.triple-large48'|'db.triple-large64'|'db.'
    ↳ 'triple-large96'|'db.triple-large128'|'db.quad-large64'|'db.quad-large96'|'db.'
    ↳ 'quad-large128'|'db.septa-large128',
```

(continues on next page)

(continued from previous page)

```

Engine='MySQL'|'postgres',
EngineVersion='string',
LicenseModel='string',
Marker='string',
MaxRecords=123,
Vpc=True|False
)

```

**Parameters**

- **DBInstanceClass** (*string*) –
- **Engine** (*string*) –
- **EngineVersion** (*string*) –
- **LicenseModel** (*string*) –
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –
- **Vpc** (*boolean*) –

**Return type** dict**Returns****Response Syntax**

```

{
  'Marker': 'string',
  'OrderableDBInstanceOptions': [
    {
      'AvailabilityZones': [
        {
          'Name': 'string',
          'NiftyStorageTypes': [
            123,
          ],
          'ProvisionedIopsCapable': True|False
        },
      ],
      'DBInstanceClass': 'string',
      'Engine': 'string',
      'EngineVersion': 'string',
      'LicenseModel': 'string',
      'MultiAZCapable': True|False,
      'ReadReplicaCapable': True|False,
      'Vpc': True|False
    },
  ],
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}

```

**Response Structure**

- (*dict*) –
  - **Marker** (*string*) –
  - **OrderableDBInstanceOptions** (*list*) –
    - \* (*dict*) –
      - **AvailabilityZones** (*list*) –
      - (*dict*) –
      - **Name** (*string*) –

- **NiftyStorageTypes** (*list*) –
- (*integer*) –
- **ProvisionedIopsCapable** (*boolean*) –
- **DBInstanceClass** (*string*) –
- **Engine** (*string*) –
- **EngineVersion** (*string*) –
- **LicenseModel** (*string*) –
- **MultiAZCapable** (*boolean*) –
- **ReadReplicaCapable** (*boolean*) –
- **Vpc** (*boolean*) –
- **ResponseMetadata** (*dict*) –
  - \* **RequestId** (*string*) –

*rd*b / Client / download\_db\_log\_file\_portion

### download\_db\_log\_file\_portion

`rd`b.Client.download\_db\_log\_file\_portion(\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.download_db_log_file_portion(  
    DBInstanceIdentifier='string',  
    LogFileName='string',  
    Marker='string',  
    NumberOfLines=123  
)
```

#### Parameters

- **DBInstanceIdentifier** (*string*) – [REQUIRED]
- **LogFileName** (*string*) –
- **Marker** (*string*) –
- **NumberOfLines** (*integer*) –

**Return type** dict

#### Returns

#### Response Syntax

```
{  
    'AdditionalDataPending': True|False,  
    'LogFileData': 'string',  
    'Marker': 'string',  
    'ResponseMetadata': {  
        'RequestId': 'string'  
    }  
}
```

#### Response Structure

- (*dict*) –
  - **AdditionalDataPending** (*boolean*) –
  - **LogFileData** (*string*) –
  - **Marker** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*rd*b / Client / get\_paginator

## get\_paginator

`rdp.Client.get_paginator(operation_name)`

Create a paginator for an operation.

**Parameters** `operation_name` (*string*) – The operation name. This is the same name as the method name on the client. For example, if the method name is `create_foo`, and you'd normally invoke the operation as `client.create_foo(**kwargs)`, if the `create_foo` operation can be paginated, you can use the call `client.get_paginator("create_foo")`.

**Raises** `OperationNotPageableError` – Raised if the operation is not pageable. You can use the `client.can_paginate` method to check if an operation is pageable.

**Return type** `L{botocore.paginate.Paginator}`

**Returns** A paginator object.

*rdp* / Client / get\_waiter

## get\_waiter

`rdp.Client.get_waiter(waiter_name)`

Returns an object that can wait for some condition.

**Parameters** `waiter_name` (*str*) – The name of the waiter to get. See the waiters section of the service docs for a list of available waiters.

**Returns** The specified waiter object.

**Return type** `botocore.waiter.Waiter`

*rdp* / Client / modify\_db\_instance

## modify\_db\_instance

`rdp.Client.modify_db_instance(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```

response = client.modify_db_instance(
    AccountingType='1'|'2',
    AllocatedStorage=123,
    AllowMajorVersionUpgrade=True|False,
    ApplyImmediately=True|False,
    AutoMinorVersionUpgrade=True|False,
    BackupRetentionPeriod=123,
    BinlogRetentionPeriod=123,
    CACertificateIdentifier='string',
    CustomBinlogRetentionPeriod=True|False,
    DBInstanceClass='db.mini'|'db.e-small1'|'db.small1'|'db.e-small2'|'db.small2'|
    ↪ 'db.e-small4'|'db.small4'|'db.e-small8'|'db.small8'|'db.e-small16'|'db.small16'|
    ↪ 'db.e-medium'|'db.medium'|'db.e-medium4'|'db.medium4'|'db.e-medium8'|'db.medium8'
    ↪ '| 'db.e-medium16'|'db.medium16'|'db.e-medium24'|'db.medium24'|'db.e-large'|'db.
    ↪ large'|'db.e-large8'|'db.large8'|'db.e-large16'|'db.large16'|'db.e-large24'|'db.
    ↪ large24'|'db.e-large32'|'db.large32'|'db.e-extra-large8'|'db.extra-large8'|'db.
    ↪ e-extra-large16'|'db.extra-large16'|'db.e-extra-large24'|'db.extra-large24'|'db.
    ↪ e-extra-large32'|'db.extra-large32'|'db.e-extra-large48'|'db.extra-large48'|'db.
    ↪ e-double-large16'|'db.double-large16'|'db.e-double-large24'|'db.double-large24'|
    ↪ 'db.e-double-large32'|'db.double-large32'|'db.e-double-large48'|'db.double-
    ↪ large48'|'db.e-double-large64'|'db.double-large64'|'db.e-double-large96'|'db.
    ↪ double-large96'|'db.triple-large32'|'db.triple-large48'|'db.triple-large64'|'db.
    ↪ triple-large96'|'db.triple-large128'|'db.quad-large64'|'db.quad-large96'|'db.
    ↪ quad-large128'|'db.septa-large128',

```

(continues on next page)

(continued from previous page)

```

DBInstanceIdentifier='string',
DBParameterGroupName='string',
DBSecurityGroups=[
    'string',
],
EngineVersion='string',
Iops=123,
MasterUserPassword='string',
MultiAZ=True|False,
NewDBInstanceIdentifier='string',
NiftyMultiAZType=123,
NiftySlavePrivateAddress='string',
OptionGroupName='string',
PreferredBackupWindow='string',
PreferredMaintenanceWindow='string',
VpcSecurityGroupIds=[
    'string',
]
)

```

### Parameters

- **AccountingType** (*string*) –
- **AllocatedStorage** (*integer*) –
- **AllowMajorVersionUpgrade** (*boolean*) –
- **ApplyImmediately** (*boolean*) –
- **AutoMinorVersionUpgrade** (*boolean*) –
- **BackupRetentionPeriod** (*integer*) –
- **BinlogRetentionPeriod** (*integer*) –
- **CACertificateIdentifier** (*string*) –
- **CustomBinlogRetentionPeriod** (*boolean*) –
- **DBInstanceClass** (*string*) –
- **DBInstanceIdentifier** (*string*) – [REQUIRED]
- **DBParameterGroupName** (*string*) –
- **DBSecurityGroups** (*list*) –
  - (*string*) –
- **EngineVersion** (*string*) –
- **Iops** (*integer*) –
- **MasterUserPassword** (*string*) –
- **MultiAZ** (*boolean*) –
- **NewDBInstanceIdentifier** (*string*) –
- **NiftyMultiAZType** (*integer*) –
- **NiftySlavePrivateAddress** (*string*) –
- **OptionGroupName** (*string*) –
- **PreferredBackupWindow** (*string*) –
- **PreferredMaintenanceWindow** (*string*) –
- **VpcSecurityGroupIds** (*list*) –
  - (*string*) –

**Return type** dict

**Returns**

### Response Syntax

```

{
    'DBInstance': {

```

(continues on next page)

(continued from previous page)

```

'AccountingType': 'string',
'AllocatedStorage': 123,
'AutoMinorVersionUpgrade': True|False,
'AvailabilityZone': 'string',
'BackupRetentionPeriod': 123,
'BinlogRetentionPeriod': 123,
'CACertificateIdentifier': 'string',
'DBInstanceClass': 'string',
'DBInstanceIdentifier': 'string',
'DBInstanceStatus': 'string',
'DBName': 'string',
'DBParameterGroups': [
    {
        'DBParameterGroupName': 'string',
        'ParameterApplyStatus': 'string'
    },
],
'DBSecurityGroups': [
    {
        'DBSecurityGroupName': 'string',
        'Status': 'string'
    },
],
'Endpoint': {
    'Address': 'string',
    'NiftyPrivateAddress': 'string',
    'Port': 123
},
'Engine': 'string',
'EngineVersion': 'string',
'ExternalReplicationInfo': {
    'ExternalMasterAddress': 'string',
    'ExternalReplicationMessage': 'string',
    'ExternalReplicationStatus': 'string',
    'ReplicationAddresses': [
        'string',
    ],
    'ReplicationPrivateAddresses': [
        'string',
    ]
},
'InstanceCreateTime': datetime(2015, 1, 1),
'LatestRestorableTime': datetime(2015, 1, 1),
'LicenseModel': 'string',
'MasterUsername': 'string',
'MultiAZ': True|False,
'NextMonthAccountingType': 'string',
'NiftyMasterPrivateAddress': 'string',
'NiftyMultiAZType': 'string',
'NiftyNetworkId': 'string',
'NiftySlavePrivateAddress': 'string',
'NiftyStorageType': 123,
'OptionGroupMemberships': [
    {
        'OptionGroupName': 'string',
        'Status': 'string'
    },
],

```

(continues on next page)

(continued from previous page)

```

    ],
    'PendingModifiedValues': {
        'AllocatedStorage': 123,
        'BackupRetentionPeriod': 123,
        'DBInstanceClass': 'string',
        'DBInstanceIdentifier': 'string',
        'EngineVersion': 'string',
        'MasterUserPassword': 'string',
        'MultiAZ': True|False,
        'Port': 123
    },
    'PreferredBackupWindow': 'string',
    'PreferredMaintenanceWindow': 'string',
    'PubliclyAccessible': True|False,
    'ReadReplicaDBInstanceIdentifiers': [
        'string',
    ],
    'ReadReplicaSourceDBInstanceIdentifier': 'string',
    'SecondaryAvailabilityZone': 'string',
    'StatusInfos': [
        {
            'Message': 'string',
            'Normal': True|False,
            'Status': 'string',
            'StatusType': 'string'
        },
    ],
    'VpcSecurityGroups': 'string'
},
'ResponseMetadata': {
    'RequestId': 'string'
}
}

```

### Response Structure

- (dict) –
  - DBInstance (dict) –
    - \* **AccountingType** (string) –
    - \* **AllocatedStorage** (integer) –
    - \* **AutoMinorVersionUpgrade** (boolean) –
    - \* **AvailabilityZone** (string) –
    - \* **BackupRetentionPeriod** (integer) –
    - \* **BinlogRetentionPeriod** (integer) –
    - \* **CACertificateIdentifier** (string) –
    - \* **DBInstanceClass** (string) –
    - \* **DBInstanceIdentifier** (string) –
    - \* **DBInstanceStatus** (string) –
    - \* **DBName** (string) –
    - \* **DBParameterGroups** (list) –
      - (dict) –
      - **DBParameterGroupName** (string) –
      - **ParameterApplyStatus** (string) –
    - \* **DBSecurityGroups** (list) –
      - (dict) –
      - **DBSecurityGroupName** (string) –



- **Status** (*string*) –
- \* **Endpoint** (*dict*) –
  - **Address** (*string*) –
  - **NiftyPrivateAddress** (*string*) –
  - **Port** (*integer*) –
- \* **Engine** (*string*) –
- \* **EngineVersion** (*string*) –
- \* **ExternalReplicationInfo** (*dict*) –
  - **ExternalMasterAddress** (*string*) –
  - **ExternalReplicationMessage** (*string*) –
  - **ExternalReplicationStatus** (*string*) –
  - **ReplicationAddresses** (*list*) –
  - (*string*) –
  - **ReplicationPrivateAddresses** (*list*) –
  - (*string*) –
- \* **InstanceCreateTime** (*datetime*) –
- \* **LatestRestorableTime** (*datetime*) –
- \* **LicenseModel** (*string*) –
- \* **MasterUsername** (*string*) –
- \* **MultiAZ** (*boolean*) –
- \* **NextMonthAccountingType** (*string*) –
- \* **NiftyMasterPrivateAddress** (*string*) –
- \* **NiftyMultiAZType** (*string*) –
- \* **NiftyNetworkId** (*string*) –
- \* **NiftySlavePrivateAddress** (*string*) –
- \* **NiftyStorageType** (*integer*) –
- \* **OptionGroupMemberships** (*list*) –
  - (*dict*) –
  - **OptionGroupName** (*string*) –
  - **Status** (*string*) –
- \* **PendingModifiedValues** (*dict*) –
  - **AllocatedStorage** (*integer*) –
  - **BackupRetentionPeriod** (*integer*) –
  - **DBInstanceClass** (*string*) –
  - **DBInstanceIdentifier** (*string*) –
  - **EngineVersion** (*string*) –
  - **MasterUserPassword** (*string*) –
  - **MultiAZ** (*boolean*) –
  - **Port** (*integer*) –
- \* **PreferredBackupWindow** (*string*) –
- \* **PreferredMaintenanceWindow** (*string*) –
- \* **PubliclyAccessible** (*boolean*) –
- \* **ReadReplicaDBInstanceIdentifiers** (*list*) –
  - (*string*) –
- \* **ReadReplicaSourceDBInstanceIdentifier** (*string*) –
- \* **SecondaryAvailabilityZone** (*string*) –
- \* **StatusInfos** (*list*) –
  - (*dict*) –
  - **Message** (*string*) –
  - **Normal** (*boolean*) –
  - **Status** (*string*) –
  - **StatusType** (*string*) –
- \* **VpcSecurityGroups** (*string*) –
- **ResponseMetadata** (*dict*) –

\* **RequestId** (*string*) –

*rdb* / Client / modify\_db\_instance\_network

## modify\_db\_instance\_network

`rdb.Client.modify_db_instance_network(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.modify_db_instance_network(  
    DBInstanceIdentifier='string',  
    NiftyMasterPrivateAddress='string',  
    NiftyNetworkId='string',  
    NiftySlavePrivateAddress='string',  
    NiftyVirtualPrivateAddress='string'  
)
```

### Parameters

- **DBInstanceIdentifier** (*string*) – [REQUIRED]
- **NiftyMasterPrivateAddress** (*string*) –
- **NiftyNetworkId** (*string*) –
- **NiftySlavePrivateAddress** (*string*) –
- **NiftyVirtualPrivateAddress** (*string*) –

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'DBInstance': {  
        'AccountingType': 'string',  
        'AllocatedStorage': 123,  
        'AutoMinorVersionUpgrade': True|False,  
        'AvailabilityZone': 'string',  
        'BackupRetentionPeriod': 123,  
        'BinlogRetentionPeriod': 123,  
        'CACertificateIdentifier': 'string',  
        'DBInstanceClass': 'string',  
        'DBInstanceIdentifier': 'string',  
        'DBInstanceStatus': 'string',  
        'DBName': 'string',  
        'DBParameterGroups': [  
            {  
                'DBParameterGroupName': 'string',  
                'ParameterApplyStatus': 'string'  
            },  
        ],  
        'DBSecurityGroups': [  
            {  
                'DBSecurityGroupName': 'string',  
                'Status': 'string'  
            },  
        ],  
        'Endpoint': {  
            'Address': 'string',
```

(continues on next page)

(continued from previous page)

```

        'NiftyPrivateAddress': 'string',
        'Port': 123
    },
    'Engine': 'string',
    'EngineVersion': 'string',
    'ExternalReplicationInfo': {
        'ExternalMasterAddress': 'string',
        'ExternalReplicationMessage': 'string',
        'ExternalReplicationStatus': 'string',
        'ReplicationAddresses': [
            'string',
        ],
        'ReplicationPrivateAddresses': [
            'string',
        ]
    },
    'InstanceCreateTime': datetime(2015, 1, 1),
    'LatestRestorableTime': datetime(2015, 1, 1),
    'LicenseModel': 'string',
    'MasterUsername': 'string',
    'MultiAZ': True|False,
    'NextMonthAccountingType': 'string',
    'NiftyMasterPrivateAddress': 'string',
    'NiftyMultiAZType': 'string',
    'NiftyNetworkId': 'string',
    'NiftySlavePrivateAddress': 'string',
    'NiftyStorageType': 123,
    'OptionGroupMemberships': [
        {
            'OptionGroupName': 'string',
            'Status': 'string'
        },
    ],
    'PendingModifiedValues': {
        'AllocatedStorage': 123,
        'BackupRetentionPeriod': 123,
        'DBInstanceClass': 'string',
        'DBInstanceIdentifier': 'string',
        'EngineVersion': 'string',
        'MasterUserPassword': 'string',
        'MultiAZ': True|False,
        'Port': 123
    },
    'PreferredBackupWindow': 'string',
    'PreferredMaintenanceWindow': 'string',
    'PubliclyAccessible': True|False,
    'ReadReplicaDBInstanceIdentifiers': [
        'string',
    ],
    'ReadReplicaSourceDBInstanceIdentifier': 'string',
    'SecondaryAvailabilityZone': 'string',
    'StatusInfos': [
        {
            'Message': 'string',
            'Normal': True|False,
            'Status': 'string',
            'StatusType': 'string'
        },
    ],

```

(continues on next page)

(continued from previous page)

```
    },
    ],
    'VpcSecurityGroups': 'string'
  },
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

### Response Structure

- (dict) –
  - DBInstance (dict) –
    - \* **AccountingType** (string) –
    - \* **AllocatedStorage** (integer) –
    - \* **AutoMinorVersionUpgrade** (boolean) –
    - \* **AvailabilityZone** (string) –
    - \* **BackupRetentionPeriod** (integer) –
    - \* **BinlogRetentionPeriod** (integer) –
    - \* **CACertificateIdentifier** (string) –
    - \* **DBInstanceClass** (string) –
    - \* **DBInstanceIdentifier** (string) –
    - \* **DBInstanceStatus** (string) –
    - \* **DBName** (string) –
    - \* **DBParameterGroups** (list) –
      - (dict) –
      - **DBParameterGroupName** (string) –
      - **ParameterApplyStatus** (string) –
    - \* **DBSecurityGroups** (list) –
      - (dict) –
      - **DBSecurityGroupName** (string) –
      - **Status** (string) –
    - \* **Endpoint** (dict) –
      - **Address** (string) –
      - **NiftyPrivateAddress** (string) –
      - **Port** (integer) –
    - \* **Engine** (string) –
    - \* **EngineVersion** (string) –
    - \* **ExternalReplicationInfo** (dict) –
      - **ExternalMasterAddress** (string) –
      - **ExternalReplicationMessage** (string) –
      - **ExternalReplicationStatus** (string) –
      - **ReplicationAddresses** (list) –
        - (string) –
        - **ReplicationPrivateAddresses** (list) –
          - (string) –
    - \* **InstanceCreateTime** (datetime) –
    - \* **LatestRestorableTime** (datetime) –
    - \* **LicenseModel** (string) –
    - \* **MasterUsername** (string) –
    - \* **MultiAZ** (boolean) –
    - \* **NextMonthAccountingType** (string) –
    - \* **NiftyMasterPrivateAddress** (string) –
    - \* **NiftyMultiAZType** (string) –

- \* **NiftyNetworkId** (*string*) –
- \* **NiftySlavePrivateAddress** (*string*) –
- \* **NiftyStorageType** (*integer*) –
- \* **OptionGroupMemberships** (*list*) –
  - (*dict*) –
  - **OptionGroupName** (*string*) –
  - **Status** (*string*) –
- \* **PendingModifiedValues** (*dict*) –
  - **AllocatedStorage** (*integer*) –
  - **BackupRetentionPeriod** (*integer*) –
  - **DBInstanceClass** (*string*) –
  - **DBInstanceIdentifier** (*string*) –
  - **EngineVersion** (*string*) –
  - **MasterUserPassword** (*string*) –
  - **MultiAZ** (*boolean*) –
  - **Port** (*integer*) –
- \* **PreferredBackupWindow** (*string*) –
- \* **PreferredMaintenanceWindow** (*string*) –
- \* **PubliclyAccessible** (*boolean*) –
- \* **ReadReplicaDBInstanceIdentifiers** (*list*) –
  - (*string*) –
- \* **ReadReplicaSourceDBInstanceIdentifier** (*string*) –
- \* **SecondaryAvailabilityZone** (*string*) –
- \* **StatusInfos** (*list*) –
  - (*dict*) –
  - **Message** (*string*) –
  - **Normal** (*boolean*) –
  - **Status** (*string*) –
  - **StatusType** (*string*) –
- \* **VpcSecurityGroups** (*string*) –
- **ResponseMetadata** (*dict*) –
  - \* **RequestId** (*string*) –

*rdb* / Client / modify\_db\_parameter\_group

## modify\_db\_parameter\_group

`rdb.Client.modify_db_parameter_group(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.modify_db_parameter_group(
    DBParameterGroupName='string',
    Parameters=[
        {
            'ApplyMethod': 'immediate'|'pending-reboot',
            'ParameterName': 'string',
            'ParameterValue': 'string'
        },
    ]
)
```

### Parameters

- **DBParameterGroupName** (*string*) – [REQUIRED]

- **Parameters** (*list*) – [REQUIRED]
  - (*dict*) –
    - \* **ApplyMethod** (*string*) – [REQUIRED]
    - \* **ParameterName** (*string*) – [REQUIRED]
    - \* **ParameterValue** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'DBParameterGroupName': 'string',
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

#### Response Structure

- (*dict*) –
  - **DBParameterGroupName** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*rdb* / Client / `modify_event_subscription`

### `modify_event_subscription`

`rdb.Client.modify_event_subscription(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.modify_event_subscription(
    Enabled=True|False,
    EventCategories=[
        'string',
    ],
    NiftyDescription='string',
    NiftyEmailAddresses=[
        'string',
    ],
    NiftySourceIds=[
        'string',
    ],
    SourceType='string',
    SubscriptionName='string'
)
```

#### Parameters

- **Enabled** (*boolean*) –
- **EventCategories** (*list*) –
  - (*string*) –
- **NiftyDescription** (*string*) –
- **NiftyEmailAddresses** (*list*) –
  - (*string*) –
- **NiftySourceIds** (*list*) –
  - (*string*) –

- **SourceType** (*string*) –
- **SubscriptionName** (*string*) – [REQUIRED]

Return type dict

Returns

#### Response Syntax

```
{
  'EventSubscription': {
    'CustSubscriptionId': 'string',
    'Enabled': True|False,
    'EventCategoriesList': [
      'string',
    ],
    'NiftyDescription': 'string',
    'NiftyEmailAddressesList': [
      'string',
    ],
    'SourceIdsList': [
      'string',
    ],
    'SourceType': 'string',
    'Status': 'string',
    'SubscriptionCreationTime': 'string'
  },
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

#### Response Structure

- (*dict*) –
  - **EventSubscription** (*dict*) –
    - \* **CustSubscriptionId** (*string*) –
    - \* **Enabled** (*boolean*) –
    - \* **EventCategoriesList** (*list*) –
      - (*string*) –
    - \* **NiftyDescription** (*string*) –
    - \* **NiftyEmailAddressesList** (*list*) –
      - (*string*) –
    - \* **SourceIdsList** (*list*) –
      - (*string*) –
    - \* **SourceType** (*string*) –
    - \* **Status** (*string*) –
    - \* **SubscriptionCreationTime** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*rd*b / Client / nifty\_get\_metric\_statistics

#### nifty\_get\_metric\_statistics

`rd`b.Client.nifty\_get\_metric\_statistics(\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.nifty_get_metric_statistics(  
    Dimensions=[  
        {  
            'Name': 'string',  
            'Value': 'string'  
        },  
    ],  
    EndTime=datetime(2015, 1, 1),  
    MetricName='BinLogDiskUsage'|'CPUUtilization'|'DatabaseConnections'|  
    ↳'DiskQueueDepth'|'FreeableMemory'|'FreeStorageSpace'|'ReplicaLag'|'SwapUsage'|  
    ↳'ReadIOPS'|'WriteIOPS'|'ReadThroughput'|'WriteThroughput',  
    StartTime=datetime(2015, 1, 1)  
)
```

#### Parameters

- **Dimensions** (*list*) – [REQUIRED]
  - (*dict*) –
    - \* **Name** (*string*) – [REQUIRED]
    - \* **Value** (*string*) – [REQUIRED]
- **EndTime** (*datetime*) –
- **MetricName** (*string*) – [REQUIRED]
- **StartTime** (*datetime*) –

Return type `dict`

#### Returns

#### Response Syntax

```
{  
    'Datapoints': [  
        {  
            'NiftyTargetName': 'string',  
            'SampleCount': 123,  
            'Sum': 123.0,  
            'Timestamp': datetime(2015, 1, 1)  
        },  
    ],  
    'Label': 'string',  
    'ResponseMetadata': {  
        'RequestId': 'string'  
    }  
}
```

#### Response Structure

- (*dict*) –
  - **Datapoints** (*list*) –
    - \* (*dict*) –
      - **NiftyTargetName** (*string*) –
      - **SampleCount** (*integer*) –
      - **Sum** (*float*) –
      - **Timestamp** (*datetime*) –
  - **Label** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*rd*b / Client / reboot\_db\_instance



## reboot\_db\_instance

`rdb.Client.reboot_db_instance(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.reboot_db_instance(
    DBInstanceIdentifier='string',
    ForceFailover=True|False,
    NiftyRebootType='0'|'1'|'2'|'3'
)
```

### Parameters

- **DBInstanceIdentifier** (*string*) – [REQUIRED]
- **ForceFailover** (*boolean*) –
- **NiftyRebootType** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```
{
  'DBInstance': {
    'AccountingType': 'string',
    'AllocatedStorage': 123,
    'AutoMinorVersionUpgrade': True|False,
    'AvailabilityZone': 'string',
    'BackupRetentionPeriod': 123,
    'BinlogRetentionPeriod': 123,
    'CACertificateIdentifier': 'string',
    'DBInstanceClass': 'string',
    'DBInstanceIdentifier': 'string',
    'DBInstanceStatus': 'string',
    'DBName': 'string',
    'DBParameterGroups': [
      {
        'DBParameterGroupName': 'string',
        'ParameterApplyStatus': 'string'
      },
    ],
    'DBSecurityGroups': [
      {
        'DBSecurityGroupName': 'string',
        'Status': 'string'
      },
    ],
    'Endpoint': {
      'Address': 'string',
      'NiftyPrivateAddress': 'string',
      'Port': 123
    },
    'Engine': 'string',
    'EngineVersion': 'string',
    'ExternalReplicationInfo': {
      'ExternalMasterAddress': 'string',
      'ExternalReplicationMessage': 'string',
      'ExternalReplicationStatus': 'string',
```

(continues on next page)

(continued from previous page)

```

        'ReplicationAddresses': [
            'string',
        ],
        'ReplicationPrivateAddresses': [
            'string',
        ]
    },
    'InstanceCreateTime': datetime(2015, 1, 1),
    'LatestRestorableTime': datetime(2015, 1, 1),
    'LicenseModel': 'string',
    'MasterUsername': 'string',
    'MultiAZ': True|False,
    'NextMonthAccountingType': 'string',
    'NiftyMasterPrivateAddress': 'string',
    'NiftyMultiAZType': 'string',
    'NiftyNetworkId': 'string',
    'NiftySlavePrivateAddress': 'string',
    'NiftyStorageType': 123,
    'OptionGroupMemberships': [
        {
            'OptionGroupName': 'string',
            'Status': 'string'
        },
    ],
    'PendingModifiedValues': {
        'AllocatedStorage': 123,
        'BackupRetentionPeriod': 123,
        'DBInstanceClass': 'string',
        'DBInstanceIdentifier': 'string',
        'EngineVersion': 'string',
        'MasterUserPassword': 'string',
        'MultiAZ': True|False,
        'Port': 123
    },
    'PreferredBackupWindow': 'string',
    'PreferredMaintenanceWindow': 'string',
    'PubliclyAccessible': True|False,
    'ReadReplicaDBInstanceIdentifiers': [
        'string',
    ],
    'ReadReplicaSourceDBInstanceIdentifier': 'string',
    'SecondaryAvailabilityZone': 'string',
    'StatusInfos': [
        {
            'Message': 'string',
            'Normal': True|False,
            'Status': 'string',
            'StatusType': 'string'
        },
    ],
    'VpcSecurityGroups': 'string'
},
'ResponseMetadata': {
    'RequestId': 'string'
}
}

```

**Response Structure**

- *(dict)* –
  - **DBInstance** *(dict)* –
    - \* **AccountingType** *(string)* –
    - \* **AllocatedStorage** *(integer)* –
    - \* **AutoMinorVersionUpgrade** *(boolean)* –
    - \* **AvailabilityZone** *(string)* –
    - \* **BackupRetentionPeriod** *(integer)* –
    - \* **BinlogRetentionPeriod** *(integer)* –
    - \* **CACertificateIdentifier** *(string)* –
    - \* **DBInstanceClass** *(string)* –
    - \* **DBInstanceIdentifier** *(string)* –
    - \* **DBInstanceStatus** *(string)* –
    - \* **DBName** *(string)* –
    - \* **DBParameterGroups** *(list)* –
      - *(dict)* –
      - **DBParameterGroupName** *(string)* –
      - **ParameterApplyStatus** *(string)* –
    - \* **DBSecurityGroups** *(list)* –
      - *(dict)* –
      - **DBSecurityGroupName** *(string)* –
      - **Status** *(string)* –
    - \* **Endpoint** *(dict)* –
      - **Address** *(string)* –
      - **NiftyPrivateAddress** *(string)* –
      - **Port** *(integer)* –
    - \* **Engine** *(string)* –
    - \* **EngineVersion** *(string)* –
    - \* **ExternalReplicationInfo** *(dict)* –
      - **ExternalMasterAddress** *(string)* –
      - **ExternalReplicationMessage** *(string)* –
      - **ExternalReplicationStatus** *(string)* –
      - **ReplicationAddresses** *(list)* –
      - *(string)* –
      - **ReplicationPrivateAddresses** *(list)* –
      - *(string)* –
    - \* **InstanceCreateTime** *(datetime)* –
    - \* **LatestRestorableTime** *(datetime)* –
    - \* **LicenseModel** *(string)* –
    - \* **MasterUsername** *(string)* –
    - \* **MultiAZ** *(boolean)* –
    - \* **NextMonthAccountingType** *(string)* –
    - \* **NiftyMasterPrivateAddress** *(string)* –
    - \* **NiftyMultiAZType** *(string)* –
    - \* **NiftyNetworkId** *(string)* –
    - \* **NiftySlavePrivateAddress** *(string)* –
    - \* **NiftyStorageType** *(integer)* –
    - \* **OptionGroupMemberships** *(list)* –
      - *(dict)* –
      - **OptionGroupName** *(string)* –
      - **Status** *(string)* –
    - \* **PendingModifiedValues** *(dict)* –
      - **AllocatedStorage** *(integer)* –
      - **BackupRetentionPeriod** *(integer)* –

- **DBInstanceClass** (*string*) –
- **DBInstanceIdentifier** (*string*) –
- **EngineVersion** (*string*) –
- **MasterUserPassword** (*string*) –
- **MultiAZ** (*boolean*) –
- **Port** (*integer*) –
- \* **PreferredBackupWindow** (*string*) –
- \* **PreferredMaintenanceWindow** (*string*) –
- \* **PubliclyAccessible** (*boolean*) –
- \* **ReadReplicaDBInstanceIdentifiers** (*list*) –
  - (*string*) –
- \* **ReadReplicaSourceDBInstanceIdentifier** (*string*) –
- \* **SecondaryAvailabilityZone** (*string*) –
- \* **StatusInfos** (*list*) –
  - (*dict*) –
  - **Message** (*string*) –
  - **Normal** (*boolean*) –
  - **Status** (*string*) –
  - **StatusType** (*string*) –
- \* **VpcSecurityGroups** (*string*) –
- **ResponseMetadata** (*dict*) –
  - \* **RequestId** (*string*) –

*rdb* / Client / `remove_source_identifier_from_subscription`

### `remove_source_identifier_from_subscription`

`rdb.Client.remove_source_identifier_from_subscription(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.remove_source_identifier_from_subscription(  
    SourceIdentifier='string',  
    SubscriptionName='string'  
)
```

#### Parameters

- **SourceIdentifier** (*string*) – [REQUIRED]
- **SubscriptionName** (*string*) – [REQUIRED]

**Return type** dict

#### Returns

#### Response Syntax

```
{  
    'EventSubscription': {  
        'CustSubscriptionId': 'string',  
        'Enabled': True|False,  
        'EventCategoriesList': [  
            'string',  
        ],  
        'NiftyDescription': 'string',  
        'NiftyEmailAddressesList': [  
            'string',  
        ],  
    },  
}
```

(continues on next page)

(continued from previous page)

```

        'SourceIdsList': [
            'string',
        ],
        'SourceType': 'string',
        'Status': 'string',
        'SubscriptionCreationTime': 'string'
    },
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}

```

**Response Structure**

- (dict) –
  - **EventSubscription** (dict) –
    - \* **CustSubscriptionId** (string) –
    - \* **Enabled** (boolean) –
    - \* **EventCategoriesList** (list) –
      - (string) –
    - \* **NiftyDescription** (string) –
    - \* **NiftyEmailAddressesList** (list) –
      - (string) –
    - \* **SourceIdsList** (list) –
      - (string) –
    - \* **SourceType** (string) –
    - \* **Status** (string) –
    - \* **SubscriptionCreationTime** (string) –
  - **ResponseMetadata** (dict) –
    - \* **RequestId** (string) –

*rdb* / Client / reset\_db\_parameter\_group**reset\_db\_parameter\_group**`rdb.Client.reset_db_parameter_group(**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```

response = client.reset_db_parameter_group(
    DBParameterGroupName='string',
    Parameters=[
        {
            'ApplyMethod': 'immediate'|'pending-reboot',
            'ParameterName': 'string'
        },
    ],
    ResetAllParameters=True|False
)

```

**Parameters**

- **DBParameterGroupName** (string) – [REQUIRED]
- **Parameters** (list) –
  - (dict) –
    - \* **ApplyMethod** (string) –

- \* **ParameterName** (*string*) –
- **ResetAllParameters** (*boolean*) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'DBParameterGroupName': 'string',
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

#### Response Structure

- (*dict*) –
  - **DBParameterGroupName** (*string*) –
  - **ResponseMetadata** (*dict*) –
    - \* **RequestId** (*string*) –

*rdb* / Client / reset\_external\_master

### reset\_external\_master

`rdb.Client.reset_external_master(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.reset_external_master(
    DBInstanceIdentifier='string'
)
```

**Parameters** **DBInstanceIdentifier** (*string*) – **[REQUIRED]**

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'DBInstance': {
        'AccountingType': 'string',
        'AllocatedStorage': 123,
        'AutoMinorVersionUpgrade': True|False,
        'AvailabilityZone': 'string',
        'BackupRetentionPeriod': 123,
        'BinlogRetentionPeriod': 123,
        'CACertificateIdentifier': 'string',
        'DBInstanceClass': 'string',
        'DBInstanceIdentifier': 'string',
        'DBInstanceStatus': 'string',
        'DBName': 'string',
        'DBParameterGroups': [
            {
                'DBParameterGroupName': 'string',
                'ParameterApplyStatus': 'string'
            },
        ],
    },
}
```

(continues on next page)

(continued from previous page)

```

],
'DBSecurityGroups': [
    {
        'DBSecurityGroupName': 'string',
        'Status': 'string'
    },
],
'Endpoint': {
    'Address': 'string',
    'NiftyPrivateAddress': 'string',
    'Port': 123
},
'Engine': 'string',
'EngineVersion': 'string',
'ExternalReplicationInfo': {
    'ExternalMasterAddress': 'string',
    'ExternalReplicationMessage': 'string',
    'ExternalReplicationStatus': 'string',
    'ReplicationAddresses': [
        'string',
    ],
    'ReplicationPrivateAddresses': [
        'string',
    ]
},
'InstanceCreateTime': datetime(2015, 1, 1),
'LatestRestorableTime': datetime(2015, 1, 1),
'LicenseModel': 'string',
'MasterUsername': 'string',
'MultiAZ': True|False,
'NextMonthAccountingType': 'string',
'NiftyMasterPrivateAddress': 'string',
'NiftyMultiAZType': 'string',
'NiftyNetworkId': 'string',
'NiftySlavePrivateAddress': 'string',
'NiftyStorageType': 123,
'OptionGroupMemberships': [
    {
        'OptionGroupName': 'string',
        'Status': 'string'
    },
],
'PendingModifiedValues': {
    'AllocatedStorage': 123,
    'BackupRetentionPeriod': 123,
    'DBInstanceClass': 'string',
    'DBInstanceIdentifier': 'string',
    'EngineVersion': 'string',
    'MasterUserPassword': 'string',
    'MultiAZ': True|False,
    'Port': 123
},
'PreferredBackupWindow': 'string',
'PreferredMaintenanceWindow': 'string',
'PubliclyAccessible': True|False,
'ReadReplicaDBInstanceIdentifiers': [
    'string',

```

(continues on next page)

(continued from previous page)

```

    ],
    'ReadReplicaSourceDBInstanceIdentifier': 'string',
    'SecondaryAvailabilityZone': 'string',
    'StatusInfos': [
        {
            'Message': 'string',
            'Normal': True|False,
            'Status': 'string',
            'StatusType': 'string'
        },
    ],
    'VpcSecurityGroups': 'string'
},
'ResponseMetadata': {
    'RequestId': 'string'
}
}

```

**Response Structure**

- (dict) –
  - **DBInstance** (dict) –
    - \* **AccountingType** (string) –
    - \* **AllocatedStorage** (integer) –
    - \* **AutoMinorVersionUpgrade** (boolean) –
    - \* **AvailabilityZone** (string) –
    - \* **BackupRetentionPeriod** (integer) –
    - \* **BinlogRetentionPeriod** (integer) –
    - \* **CACertificateIdentifier** (string) –
    - \* **DBInstanceClass** (string) –
    - \* **DBInstanceIdentifier** (string) –
    - \* **DBInstanceStatus** (string) –
    - \* **DBName** (string) –
    - \* **DBParameterGroups** (list) –
      - (dict) –
      - **DBParameterGroupName** (string) –
      - **ParameterApplyStatus** (string) –
    - \* **DBSecurityGroups** (list) –
      - (dict) –
      - **DBSecurityGroupName** (string) –
      - **Status** (string) –
    - \* **Endpoint** (dict) –
      - **Address** (string) –
      - **NiftyPrivateAddress** (string) –
      - **Port** (integer) –
    - \* **Engine** (string) –
    - \* **EngineVersion** (string) –
    - \* **ExternalReplicationInfo** (dict) –
      - **ExternalMasterAddress** (string) –
      - **ExternalReplicationMessage** (string) –
      - **ExternalReplicationStatus** (string) –
      - **ReplicationAddresses** (list) –
        - (string) –
      - **ReplicationPrivateAddresses** (list) –
        - (string) –



- \* **InstanceCreateTime** (*datetime*) –
- \* **LatestRestorableTime** (*datetime*) –
- \* **LicenseModel** (*string*) –
- \* **MasterUsername** (*string*) –
- \* **MultiAZ** (*boolean*) –
- \* **NextMonthAccountingType** (*string*) –
- \* **NiftyMasterPrivateAddress** (*string*) –
- \* **NiftyMultiAZType** (*string*) –
- \* **NiftyNetworkId** (*string*) –
- \* **NiftySlavePrivateAddress** (*string*) –
- \* **NiftyStorageType** (*integer*) –
- \* **OptionGroupMemberships** (*list*) –
  - (*dict*) –
  - **OptionGroupName** (*string*) –
  - **Status** (*string*) –
- \* **PendingModifiedValues** (*dict*) –
  - **AllocatedStorage** (*integer*) –
  - **BackupRetentionPeriod** (*integer*) –
  - **DBInstanceClass** (*string*) –
  - **DBInstanceIdentifier** (*string*) –
  - **EngineVersion** (*string*) –
  - **MasterUserPassword** (*string*) –
  - **MultiAZ** (*boolean*) –
  - **Port** (*integer*) –
- \* **PreferredBackupWindow** (*string*) –
- \* **PreferredMaintenanceWindow** (*string*) –
- \* **PubliclyAccessible** (*boolean*) –
- \* **ReadReplicaDBInstanceIdentifiers** (*list*) –
  - (*string*) –
- \* **ReadReplicaSourceDBInstanceIdentifier** (*string*) –
- \* **SecondaryAvailabilityZone** (*string*) –
- \* **StatusInfos** (*list*) –
  - (*dict*) –
  - **Message** (*string*) –
  - **Normal** (*boolean*) –
  - **Status** (*string*) –
  - **StatusType** (*string*) –
- \* **VpcSecurityGroups** (*string*) –
- **ResponseMetadata** (*dict*) –
  - \* **RequestId** (*string*) –

*rdb* / Client / `restore_db_instance_from_db_snapshot`

## restore\_db\_instance\_from\_db\_snapshot

`rdb.Client.restore_db_instance_from_db_snapshot(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.restore_db_instance_from_db_snapshot(
    AccountingType='1'|'2',
    AutoMinorVersionUpgrade=True|False,
    AvailabilityZone='string',
```

(continues on next page)

(continued from previous page)

```

    DBInstanceClass='db.mini'|'db.e-small'|'db.small'|'db.e-small2'|'db.small2'|
    ↳ 'db.e-small4'|'db.small4'|'db.e-small8'|'db.small8'|'db.e-small16'|'db.small16'|
    ↳ 'db.e-medium'|'db.medium'|'db.e-medium4'|'db.medium4'|'db.e-medium8'|'db.medium8
    ↳ '| 'db.e-medium16'|'db.medium16'|'db.e-medium24'|'db.medium24'|'db.e-large'|'db.
    ↳ large'|'db.e-large8'|'db.large8'|'db.e-large16'|'db.large16'|'db.e-large24'|'db.
    ↳ large24'|'db.e-large32'|'db.large32'|'db.e-extra-large8'|'db.extra-large8'|'db.
    ↳ e-extra-large16'|'db.extra-large16'|'db.e-extra-large24'|'db.extra-large24'|'db.
    ↳ e-extra-large32'|'db.extra-large32'|'db.e-extra-large48'|'db.extra-large48'|'db.
    ↳ e-double-large16'|'db.double-large16'|'db.e-double-large24'|'db.double-large24'|
    ↳ 'db.e-double-large32'|'db.double-large32'|'db.e-double-large48'|'db.double-
    ↳ large48'|'db.e-double-large64'|'db.double-large64'|'db.e-double-large96'|'db.
    ↳ double-large96'|'db.triple-large32'|'db.triple-large48'|'db.triple-large64'|'db.
    ↳ triple-large96'|'db.triple-large128'|'db.quad-large64'|'db.quad-large96'|'db.
    ↳ quad-large128'|'db.septa-large128',
    DBInstanceIdentifier='string',
    DBName='string',
    DBSnapshotIdentifier='string',
    DBSubnetGroupName='string',
    Engine='string',
    Iops=123,
    LicenseModel='string',
    MultiAZ=True|False,
    NiftyDBParameterGroupName='string',
    NiftyDBSecurityGroups=[
        'string',
    ],
    NiftyMasterPrivateAddress='string',
    NiftyMultiAZType=123,
    NiftyNetworkId='string',
    NiftySlavePrivateAddress='string',
    NiftyStorageType=123,
    NiftyVirtualPrivateAddress='string',
    OptionGroupName='string',
    Port=123,
    PubliclyAccessible=True|False
)

```

### Parameters

- **AccountingType** (*string*) –
- **AutoMinorVersionUpgrade** (*boolean*) –
- **AvailabilityZone** (*string*) –
- **DBInstanceClass** (*string*) – [REQUIRED]
- **DBInstanceIdentifier** (*string*) – [REQUIRED]
- **DBName** (*string*) –
- **DBSnapshotIdentifier** (*string*) – [REQUIRED]
- **DBSubnetGroupName** (*string*) –
- **Engine** (*string*) –
- **Iops** (*integer*) –
- **LicenseModel** (*string*) –
- **MultiAZ** (*boolean*) –
- **NiftyDBParameterGroupName** (*string*) –
- **NiftyDBSecurityGroups** (*list*) –
  - (*string*) –
- **NiftyMasterPrivateAddress** (*string*) –
- **NiftyMultiAZType** (*integer*) –
- **NiftyNetworkId** (*string*) –

- **NiftySlavePrivateAddress** (*string*) –
- **NiftyStorageType** (*integer*) –
- **NiftyVirtualPrivateAddress** (*string*) –
- **OptionGroupName** (*string*) –
- **Port** (*integer*) –
- **PubliclyAccessible** (*boolean*) –

**Return type** dict

**Returns**

### Response Syntax

```
{
  'DBInstance': {
    'AccountingType': 'string',
    'AllocatedStorage': 123,
    'AutoMinorVersionUpgrade': True|False,
    'AvailabilityZone': 'string',
    'BackupRetentionPeriod': 123,
    'BinlogRetentionPeriod': 123,
    'CACertificateIdentifier': 'string',
    'DBInstanceClass': 'string',
    'DBInstanceIdentifier': 'string',
    'DBInstanceStatus': 'string',
    'DBName': 'string',
    'DBParameterGroups': [
      {
        'DBParameterGroupName': 'string',
        'ParameterApplyStatus': 'string'
      },
    ],
    'DBSecurityGroups': [
      {
        'DBSecurityGroupName': 'string',
        'Status': 'string'
      },
    ],
    'Endpoint': {
      'Address': 'string',
      'NiftyPrivateAddress': 'string',
      'Port': 123
    },
    'Engine': 'string',
    'EngineVersion': 'string',
    'ExternalReplicationInfo': {
      'ExternalMasterAddress': 'string',
      'ExternalReplicationMessage': 'string',
      'ExternalReplicationStatus': 'string',
      'ReplicationAddresses': [
        'string',
      ],
      'ReplicationPrivateAddresses': [
        'string',
      ]
    },
    'InstanceCreateTime': datetime(2015, 1, 1),
    'LatestRestorableTime': datetime(2015, 1, 1),
    'LicenseModel': 'string',
    'MasterUsername': 'string',
  },
}
```

(continues on next page)

(continued from previous page)

```

'MultiAZ': True|False,
'NextMonthAccountingType': 'string',
'NiftyMasterPrivateAddress': 'string',
'NiftyMultiAZType': 'string',
'NiftyNetworkId': 'string',
'NiftySlavePrivateAddress': 'string',
'NiftyStorageType': 123,
'OptionGroupMemberships': [
    {
        'OptionGroupName': 'string',
        'Status': 'string'
    },
],
'PendingModifiedValues': {
    'AllocatedStorage': 123,
    'BackupRetentionPeriod': 123,
    'DBInstanceClass': 'string',
    'DBInstanceIdentifier': 'string',
    'EngineVersion': 'string',
    'MasterUserPassword': 'string',
    'MultiAZ': True|False,
    'Port': 123
},
'PreferredBackupWindow': 'string',
'PreferredMaintenanceWindow': 'string',
'PubliclyAccessible': True|False,
'ReadReplicaDBInstanceIdentifiers': [
    'string',
],
'ReadReplicaSourceDBInstanceIdentifier': 'string',
'SecondaryAvailabilityZone': 'string',
'StatusInfos': [
    {
        'Message': 'string',
        'Normal': True|False,
        'Status': 'string',
        'StatusType': 'string'
    },
],
'VpcSecurityGroups': 'string'
},
'ResponseMetadata': {
    'RequestId': 'string'
}
}

```

**Response Structure**

- (dict) –
  - DBInstance (dict) –
    - \* AccountingType (string) –
    - \* AllocatedStorage (integer) –
    - \* AutoMinorVersionUpgrade (boolean) –
    - \* AvailabilityZone (string) –
    - \* BackupRetentionPeriod (integer) –
    - \* BinlogRetentionPeriod (integer) –
    - \* CACertificateIdentifier (string) –

- \* **DBInstanceClass** (*string*) –
- \* **DBInstanceIdentifier** (*string*) –
- \* **DBInstanceStatus** (*string*) –
- \* **DBName** (*string*) –
- \* **DBParameterGroups** (*list*) –
  - (*dict*) –
  - **DBParameterGroupName** (*string*) –
  - **ParameterApplyStatus** (*string*) –
- \* **DBSecurityGroups** (*list*) –
  - (*dict*) –
  - **DBSecurityGroupName** (*string*) –
  - **Status** (*string*) –
- \* **Endpoint** (*dict*) –
  - **Address** (*string*) –
  - **NiftyPrivateAddress** (*string*) –
  - **Port** (*integer*) –
- \* **Engine** (*string*) –
- \* **EngineVersion** (*string*) –
- \* **ExternalReplicationInfo** (*dict*) –
  - **ExternalMasterAddress** (*string*) –
  - **ExternalReplicationMessage** (*string*) –
  - **ExternalReplicationStatus** (*string*) –
  - **ReplicationAddresses** (*list*) –
  - (*string*) –
  - **ReplicationPrivateAddresses** (*list*) –
  - (*string*) –
- \* **InstanceCreateTime** (*datetime*) –
- \* **LatestRestorableTime** (*datetime*) –
- \* **LicenseModel** (*string*) –
- \* **MasterUsername** (*string*) –
- \* **MultiAZ** (*boolean*) –
- \* **NextMonthAccountingType** (*string*) –
- \* **NiftyMasterPrivateAddress** (*string*) –
- \* **NiftyMultiAZType** (*string*) –
- \* **NiftyNetworkId** (*string*) –
- \* **NiftySlavePrivateAddress** (*string*) –
- \* **NiftyStorageType** (*integer*) –
- \* **OptionGroupMemberships** (*list*) –
  - (*dict*) –
  - **OptionGroupName** (*string*) –
  - **Status** (*string*) –
- \* **PendingModifiedValues** (*dict*) –
  - **AllocatedStorage** (*integer*) –
  - **BackupRetentionPeriod** (*integer*) –
  - **DBInstanceClass** (*string*) –
  - **DBInstanceIdentifier** (*string*) –
  - **EngineVersion** (*string*) –
  - **MasterUserPassword** (*string*) –
  - **MultiAZ** (*boolean*) –
  - **Port** (*integer*) –
- \* **PreferredBackupWindow** (*string*) –
- \* **PreferredMaintenanceWindow** (*string*) –
- \* **PubliclyAccessible** (*boolean*) –
- \* **ReadReplicaDBInstanceIdentifiers** (*list*) –

- (string) –
- \* **ReadReplicaSourceDBInstanceIdentifier** (string) –
- \* **SecondaryAvailabilityZone** (string) –
- \* **StatusInfos** (list) –
  - (dict) –
  - **Message** (string) –
  - **Normal** (boolean) –
  - **Status** (string) –
  - **StatusType** (string) –
- \* **VpcSecurityGroups** (string) –
- **ResponseMetadata** (dict) –
  - \* **RequestId** (string) –

*rdb* / Client / `restore_db_instance_to_point_in_time`

## restore\_db\_instance\_to\_point\_in\_time

`rdb.Client.restore_db_instance_to_point_in_time(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.restore_db_instance_to_point_in_time(
    AccountingType='1'|'2',
    AutoMinorVersionUpgrade=True|False,
    AvailabilityZone='string',
    DBInstanceClass='db.mini'|'db.e-small1'|'db.small1'|'db.e-small2'|'db.small2'|
    ↪ 'db.e-small4'|'db.small4'|'db.e-small8'|'db.small8'|'db.e-small16'|'db.small16'|
    ↪ 'db.e-medium'|'db.medium'|'db.e-medium4'|'db.medium4'|'db.e-medium8'|'db.medium8'
    ↪ '| 'db.e-medium16'|'db.medium16'|'db.e-medium24'|'db.medium24'|'db.e-large'|'db.
    ↪ large'|'db.e-large8'|'db.large8'|'db.e-large16'|'db.large16'|'db.e-large24'|'db.
    ↪ large24'|'db.e-large32'|'db.large32'|'db.e-extra-large8'|'db.extra-large8'|'db.
    ↪ e-extra-large16'|'db.extra-large16'|'db.e-extra-large24'|'db.extra-large24'|'db.
    ↪ e-extra-large32'|'db.extra-large32'|'db.e-extra-large48'|'db.extra-large48'|'db.
    ↪ e-double-large16'|'db.double-large16'|'db.e-double-large24'|'db.double-large24'|
    ↪ 'db.e-double-large32'|'db.double-large32'|'db.e-double-large48'|'db.double-
    ↪ large48'|'db.e-double-large64'|'db.double-large64'|'db.e-double-large96'|'db.
    ↪ double-large96'|'db.triple-large32'|'db.triple-large48'|'db.triple-large64'|'db.
    ↪ triple-large96'|'db.triple-large128'|'db.quad-large64'|'db.quad-large96'|'db.
    ↪ quad-large128'|'db.septa-large128',
    DBName='string',
    DBSubnetGroupName='string',
    Engine='string',
    Iops=123,
    LicenseModel='string',
    MultiAZ=True|False,
    NiftyDBParameterGroupName='string',
    NiftyDBSecurityGroups=[
        'string',
    ],
    NiftyMasterPrivateAddress='string',
    NiftyMultiAZType=123,
    NiftyNetworkId='string',
    NiftySlavePrivateAddress='string',
    NiftyStorageType=123,
    NiftyVirtualPrivateAddress='string',
    OptionGroupName='string',
```

(continues on next page)

(continued from previous page)

```

Port=123,
PubliclyAccessible=True|False,
RestoreTime=datetime(2015, 1, 1),
SourceDBInstanceIdentifier='string',
TargetDBInstanceIdentifier='string',
UseLatestRestorableTime=True|False
)

```

### Parameters

- **AccountingType** (*string*) –
- **AutoMinorVersionUpgrade** (*boolean*) –
- **AvailabilityZone** (*string*) –
- **DBInstanceClass** (*string*) –
- **DBName** (*string*) –
- **DBSubnetGroupName** (*string*) –
- **Engine** (*string*) –
- **Iops** (*integer*) –
- **LicenseModel** (*string*) –
- **MultiAZ** (*boolean*) –
- **NiftyDBParameterGroupName** (*string*) –
- **NiftyDBSecurityGroups** (*list*) –  
– (*string*) –
- **NiftyMasterPrivateAddress** (*string*) –
- **NiftyMultiAZType** (*integer*) –
- **NiftyNetworkId** (*string*) –
- **NiftySlavePrivateAddress** (*string*) –
- **NiftyStorageType** (*integer*) –
- **NiftyVirtualPrivateAddress** (*string*) –
- **OptionGroupName** (*string*) –
- **Port** (*integer*) –
- **PubliclyAccessible** (*boolean*) –
- **RestoreTime** (*datetime*) –
- **SourceDBInstanceIdentifier** (*string*) – [REQUIRED]
- **TargetDBInstanceIdentifier** (*string*) – [REQUIRED]
- **UseLatestRestorableTime** (*boolean*) –

**Return type** dict

**Returns**

### Response Syntax

```

{
  'DBInstance': {
    'AccountingType': 'string',
    'AllocatedStorage': 123,
    'AutoMinorVersionUpgrade': True|False,
    'AvailabilityZone': 'string',
    'BackupRetentionPeriod': 123,
    'BinlogRetentionPeriod': 123,
    'CACertificateIdentifier': 'string',
    'DBInstanceClass': 'string',
    'DBInstanceIdentifier': 'string',
    'DBInstanceStatus': 'string',
    'DBName': 'string',
    'DBParameterGroups': [

```

(continues on next page)

(continued from previous page)

```

        {
            'DBParameterGroupName': 'string',
            'ParameterApplyStatus': 'string'
        },
    ],
    'DBSecurityGroups': [
        {
            'DBSecurityGroupName': 'string',
            'Status': 'string'
        },
    ],
    'Endpoint': {
        'Address': 'string',
        'NiftyPrivateAddress': 'string',
        'Port': 123
    },
    'Engine': 'string',
    'EngineVersion': 'string',
    'ExternalReplicationInfo': {
        'ExternalMasterAddress': 'string',
        'ExternalReplicationMessage': 'string',
        'ExternalReplicationStatus': 'string',
        'ReplicationAddresses': [
            'string',
        ],
        'ReplicationPrivateAddresses': [
            'string',
        ]
    },
    'InstanceCreateTime': datetime(2015, 1, 1),
    'LatestRestorableTime': datetime(2015, 1, 1),
    'LicenseModel': 'string',
    'MasterUsername': 'string',
    'MultiAZ': True|False,
    'NextMonthAccountingType': 'string',
    'NiftyMasterPrivateAddress': 'string',
    'NiftyMultiAZType': 'string',
    'NiftyNetworkId': 'string',
    'NiftySlavePrivateAddress': 'string',
    'NiftyStorageType': 123,
    'OptionGroupMemberships': [
        {
            'OptionGroupName': 'string',
            'Status': 'string'
        },
    ],
    'PendingModifiedValues': {
        'AllocatedStorage': 123,
        'BackupRetentionPeriod': 123,
        'DBInstanceClass': 'string',
        'DBInstanceIdentifier': 'string',
        'EngineVersion': 'string',
        'MasterUserPassword': 'string',
        'MultiAZ': True|False,
        'Port': 123
    },
    'PreferredBackupWindow': 'string',

```

(continues on next page)



(continued from previous page)

```

    'PreferredMaintenanceWindow': 'string',
    'PubliclyAccessible': True|False,
    'ReadReplicaDBInstanceIdentifiers': [
        'string',
    ],
    'ReadReplicaSourceDBInstanceIdentifier': 'string',
    'SecondaryAvailabilityZone': 'string',
    'StatusInfos': [
        {
            'Message': 'string',
            'Normal': True|False,
            'Status': 'string',
            'StatusType': 'string'
        },
    ],
    'VpcSecurityGroups': 'string'
},
'ResponseMetadata': {
    'RequestId': 'string'
}
}

```

## Response Structure

- (dict) –
  - DBInstance (dict) –
    - \* **AccountingType** (string) –
    - \* **AllocatedStorage** (integer) –
    - \* **AutoMinorVersionUpgrade** (boolean) –
    - \* **AvailabilityZone** (string) –
    - \* **BackupRetentionPeriod** (integer) –
    - \* **BinlogRetentionPeriod** (integer) –
    - \* **CACertificateIdentifier** (string) –
    - \* **DBInstanceClass** (string) –
    - \* **DBInstanceIdentifier** (string) –
    - \* **DBInstanceStatus** (string) –
    - \* **DBName** (string) –
    - \* **DBParameterGroups** (list) –
      - (dict) –
      - **DBParameterGroupName** (string) –
      - **ParameterApplyStatus** (string) –
    - \* **DBSecurityGroups** (list) –
      - (dict) –
      - **DBSecurityGroupName** (string) –
      - **Status** (string) –
    - \* **Endpoint** (dict) –
      - **Address** (string) –
      - **NiftyPrivateAddress** (string) –
      - **Port** (integer) –
    - \* **Engine** (string) –
    - \* **EngineVersion** (string) –
    - \* **ExternalReplicationInfo** (dict) –
      - **ExternalMasterAddress** (string) –
      - **ExternalReplicationMessage** (string) –
      - **ExternalReplicationStatus** (string) –

- **ReplicationAddresses** (*list*) –
- (*string*) –
- **ReplicationPrivateAddresses** (*list*) –
- (*string*) –
- \* **InstanceCreateTime** (*datetime*) –
- \* **LatestRestorableTime** (*datetime*) –
- \* **LicenseModel** (*string*) –
- \* **MasterUsername** (*string*) –
- \* **MultiAZ** (*boolean*) –
- \* **NextMonthAccountingType** (*string*) –
- \* **NiftyMasterPrivateAddress** (*string*) –
- \* **NiftyMultiAZType** (*string*) –
- \* **NiftyNetworkId** (*string*) –
- \* **NiftySlavePrivateAddress** (*string*) –
- \* **NiftyStorageType** (*integer*) –
- \* **OptionGroupMemberships** (*list*) –
  - (*dict*) –
  - **OptionGroupName** (*string*) –
  - **Status** (*string*) –
- \* **PendingModifiedValues** (*dict*) –
  - **AllocatedStorage** (*integer*) –
  - **BackupRetentionPeriod** (*integer*) –
  - **DBInstanceClass** (*string*) –
  - **DBInstanceIdentifier** (*string*) –
  - **EngineVersion** (*string*) –
  - **MasterUserPassword** (*string*) –
  - **MultiAZ** (*boolean*) –
  - **Port** (*integer*) –
- \* **PreferredBackupWindow** (*string*) –
- \* **PreferredMaintenanceWindow** (*string*) –
- \* **PubliclyAccessible** (*boolean*) –
- \* **ReadReplicaDBInstanceIdentifiers** (*list*) –
  - (*string*) –
- \* **ReadReplicaSourceDBInstanceIdentifier** (*string*) –
- \* **SecondaryAvailabilityZone** (*string*) –
- \* **StatusInfos** (*list*) –
  - (*dict*) –
  - **Message** (*string*) –
  - **Normal** (*boolean*) –
  - **Status** (*string*) –
  - **StatusType** (*string*) –
- \* **VpcSecurityGroups** (*string*) –
- **ResponseMetadata** (*dict*) –
  - \* **RequestId** (*string*) –

*rdb* / Client / revoke\_db\_security\_group\_ingress

### revoke\_db\_security\_group\_ingress

`rdb.Client.revoke_db_security_group_ingress(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.revoke_db_security_group_ingress(
    CIDRIP='string',
    DBSecurityGroupName='string',
    EC2SecurityGroupId='string',
    EC2SecurityGroupName='string',
    EC2SecurityGroupOwnerId='string'
)
```

#### Parameters

- **CIDRIP** (*string*) –
- **DBSecurityGroupName** (*string*) – [REQUIRED]
- **EC2SecurityGroupId** (*string*) –
- **EC2SecurityGroupName** (*string*) –
- **EC2SecurityGroupOwnerId** (*string*) –

**Return type** dict

#### Returns

#### Response Syntax

```
{
  'DBSecurityGroup': {
    'DBSecurityGroupDescription': 'string',
    'DBSecurityGroupName': 'string',
    'EC2SecurityGroups': [
      {
        'EC2SecurityGroupName': 'string',
        'EC2SecurityGroupOwnerId': 'string',
        'Status': 'string'
      },
    ],
    'IPRanges': [
      {
        'CIDRIP': 'string',
        'Status': 'string'
      },
    ],
    'NiftyAvailabilityZone': 'string',
    'OwnerId': 'string'
  },
  'ResponseMetadata': {
    'RequestId': 'string'
  }
}
```

#### Response Structure

- (*dict*) –
  - **DBSecurityGroup** (*dict*) –
    - \* **DBSecurityGroupDescription** (*string*) –
    - \* **DBSecurityGroupName** (*string*) –
    - \* **EC2SecurityGroups** (*list*) –
      - (*dict*) –
      - **EC2SecurityGroupName** (*string*) –
      - **EC2SecurityGroupOwnerId** (*string*) –
      - **Status** (*string*) –
    - \* **IPRanges** (*list*) –
      - (*dict*) –
      - **CIDRIP** (*string*) –

- **Status** (*string*) –
- \* **NiftyAvailabilityZone** (*string*) –
- \* **OwnerId** (*string*) –
- **ResponseMetadata** (*dict*) –
- \* **RequestId** (*string*) –

*rdb* / Client / `set_external_master`

## `set_external_master`

`rdb.Client.set_external_master` (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.set_external_master(  
    BinlogFileName='string',  
    BinlogPosition=123,  
    DBInstanceIdentifier='string',  
    MasterHost='string',  
    MasterPort=123,  
    ReplicationUserName='string',  
    ReplicationUserPassword='string'  
)
```

### Parameters

- **BinlogFileName** (*string*) – [REQUIRED]
- **BinlogPosition** (*integer*) – [REQUIRED]
- **DBInstanceIdentifier** (*string*) – [REQUIRED]
- **MasterHost** (*string*) – [REQUIRED]
- **MasterPort** (*integer*) –
- **ReplicationUserName** (*string*) – [REQUIRED]
- **ReplicationUserPassword** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'DBInstance': {  
        'AccountingType': 'string',  
        'AllocatedStorage': 123,  
        'AutoMinorVersionUpgrade': True|False,  
        'AvailabilityZone': 'string',  
        'BackupRetentionPeriod': 123,  
        'BinlogRetentionPeriod': 123,  
        'CACertificateIdentifier': 'string',  
        'DBInstanceClass': 'string',  
        'DBInstanceIdentifier': 'string',  
        'DBInstanceStatus': 'string',  
        'DBName': 'string',  
        'DBParameterGroups': [  
            {  
                'DBParameterGroupName': 'string',  
                'ParameterApplyStatus': 'string'  
            },  
        ],  
    },  
}
```

(continues on next page)

(continued from previous page)

```

'DBSecurityGroups': [
    {
        'DBSecurityGroupName': 'string',
        'Status': 'string'
    },
],
'Endpoint': {
    'Address': 'string',
    'NiftyPrivateAddress': 'string',
    'Port': 123
},
'Engine': 'string',
'EngineVersion': 'string',
'ExternalReplicationInfo': {
    'ExternalMasterAddress': 'string',
    'ExternalReplicationMessage': 'string',
    'ExternalReplicationStatus': 'string',
    'ReplicationAddresses': [
        'string',
    ],
    'ReplicationPrivateAddresses': [
        'string',
    ]
},
'InstanceCreateTime': datetime(2015, 1, 1),
'LatestRestorableTime': datetime(2015, 1, 1),
'LicenseModel': 'string',
'MasterUsername': 'string',
'MultiAZ': True|False,
'NextMonthAccountingType': 'string',
'NiftyMasterPrivateAddress': 'string',
'NiftyMultiAZType': 'string',
'NiftyNetworkId': 'string',
'NiftySlavePrivateAddress': 'string',
'NiftyStorageType': 123,
'OptionGroupMemberships': [
    {
        'OptionGroupName': 'string',
        'Status': 'string'
    },
],
'PendingModifiedValues': {
    'AllocatedStorage': 123,
    'BackupRetentionPeriod': 123,
    'DBInstanceClass': 'string',
    'DBInstanceIdentifier': 'string',
    'EngineVersion': 'string',
    'MasterUserPassword': 'string',
    'MultiAZ': True|False,
    'Port': 123
},
'PreferredBackupWindow': 'string',
'PreferredMaintenanceWindow': 'string',
'PubliclyAccessible': True|False,
'ReadReplicaDBInstanceIdentifiers': [
    'string',
],

```

(continues on next page)

(continued from previous page)

```

        'ReadReplicaSourceDBInstanceIdentifier': 'string',
        'SecondaryAvailabilityZone': 'string',
        'StatusInfos': [
            {
                'Message': 'string',
                'Normal': True|False,
                'Status': 'string',
                'StatusType': 'string'
            },
        ],
        'VpcSecurityGroups': 'string'
    },
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}

```

**Response Structure**

- *(dict)* –
  - **DBInstance** (*dict*) –
    - \* **AccountingType** (*string*) –
    - \* **AllocatedStorage** (*integer*) –
    - \* **AutoMinorVersionUpgrade** (*boolean*) –
    - \* **AvailabilityZone** (*string*) –
    - \* **BackupRetentionPeriod** (*integer*) –
    - \* **BinlogRetentionPeriod** (*integer*) –
    - \* **CACertificateIdentifier** (*string*) –
    - \* **DBInstanceClass** (*string*) –
    - \* **DBInstanceIdentifier** (*string*) –
    - \* **DBInstanceStatus** (*string*) –
    - \* **DBName** (*string*) –
    - \* **DBParameterGroups** (*list*) –
      - *(dict)* –
      - **DBParameterGroupName** (*string*) –
      - **ParameterApplyStatus** (*string*) –
    - \* **DBSecurityGroups** (*list*) –
      - *(dict)* –
      - **DBSecurityGroupName** (*string*) –
      - **Status** (*string*) –
    - \* **Endpoint** (*dict*) –
      - **Address** (*string*) –
      - **NiftyPrivateAddress** (*string*) –
      - **Port** (*integer*) –
    - \* **Engine** (*string*) –
    - \* **EngineVersion** (*string*) –
    - \* **ExternalReplicationInfo** (*dict*) –
      - **ExternalMasterAddress** (*string*) –
      - **ExternalReplicationMessage** (*string*) –
      - **ExternalReplicationStatus** (*string*) –
      - **ReplicationAddresses** (*list*) –
        - (*string*) –
      - **ReplicationPrivateAddresses** (*list*) –
        - (*string*) –
    - \* **InstanceCreateTime** (*datetime*) –

- \* **LatestRestorableTime** (*datetime*) –
- \* **LicenseModel** (*string*) –
- \* **MasterUsername** (*string*) –
- \* **MultiAZ** (*boolean*) –
- \* **NextMonthAccountingType** (*string*) –
- \* **NiftyMasterPrivateAddress** (*string*) –
- \* **NiftyMultiAZType** (*string*) –
- \* **NiftyNetworkId** (*string*) –
- \* **NiftySlavePrivateAddress** (*string*) –
- \* **NiftyStorageType** (*integer*) –
- \* **OptionGroupMemberships** (*list*) –
  - (*dict*) –
  - **OptionGroupName** (*string*) –
  - **Status** (*string*) –
- \* **PendingModifiedValues** (*dict*) –
  - **AllocatedStorage** (*integer*) –
  - **BackupRetentionPeriod** (*integer*) –
  - **DBInstanceClass** (*string*) –
  - **DBInstanceIdentifier** (*string*) –
  - **EngineVersion** (*string*) –
  - **MasterUserPassword** (*string*) –
  - **MultiAZ** (*boolean*) –
  - **Port** (*integer*) –
- \* **PreferredBackupWindow** (*string*) –
- \* **PreferredMaintenanceWindow** (*string*) –
- \* **PubliclyAccessible** (*boolean*) –
- \* **ReadReplicaDBInstanceIdentifiers** (*list*) –
  - (*string*) –
- \* **ReadReplicaSourceDBInstanceIdentifier** (*string*) –
- \* **SecondaryAvailabilityZone** (*string*) –
- \* **StatusInfos** (*list*) –
  - (*dict*) –
  - **Message** (*string*) –
  - **Normal** (*boolean*) –
  - **Status** (*string*) –
  - **StatusType** (*string*) –
- \* **VpcSecurityGroups** (*string*) –
- **ResponseMetadata** (*dict*) –
  - \* **RequestId** (*string*) –

*rdb* / Client / start\_replication

## start\_replication

`rdb.Client.start_replication(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.start_replication(
    DBInstanceIdentifier='string'
)
```

**Parameters** `DBInstanceIdentifier` (*string*) – [REQUIRED]

**Return type** dict

## Returns

### Response Syntax

```
{
  'DBInstance': {
    'AccountingType': 'string',
    'AllocatedStorage': 123,
    'AutoMinorVersionUpgrade': True|False,
    'AvailabilityZone': 'string',
    'BackupRetentionPeriod': 123,
    'BinlogRetentionPeriod': 123,
    'CACertificateIdentifier': 'string',
    'DBInstanceClass': 'string',
    'DBInstanceIdentifier': 'string',
    'DBInstanceStatus': 'string',
    'DBName': 'string',
    'DBParameterGroups': [
      {
        'DBParameterGroupName': 'string',
        'ParameterApplyStatus': 'string'
      },
    ],
    'DBSecurityGroups': [
      {
        'DBSecurityGroupName': 'string',
        'Status': 'string'
      },
    ],
    'Endpoint': {
      'Address': 'string',
      'NiftyPrivateAddress': 'string',
      'Port': 123
    },
    'Engine': 'string',
    'EngineVersion': 'string',
    'ExternalReplicationInfo': {
      'ExternalMasterAddress': 'string',
      'ExternalReplicationMessage': 'string',
      'ExternalReplicationStatus': 'string',
      'ReplicationAddresses': [
        'string',
      ],
      'ReplicationPrivateAddresses': [
        'string',
      ]
    },
    'InstanceCreateTime': datetime(2015, 1, 1),
    'LatestRestorableTime': datetime(2015, 1, 1),
    'LicenseModel': 'string',
    'MasterUsername': 'string',
    'MultiAZ': True|False,
    'NextMonthAccountingType': 'string',
    'NiftyMasterPrivateAddress': 'string',
    'NiftyMultiAZType': 'string',
    'NiftyNetworkId': 'string',
    'NiftySlavePrivateAddress': 'string',
    'NiftyStorageType': 123,
```

(continues on next page)



(continued from previous page)

```

'OptionGroupMemberships': [
    {
        'OptionGroupName': 'string',
        'Status': 'string'
    },
],
'PendingModifiedValues': {
    'AllocatedStorage': 123,
    'BackupRetentionPeriod': 123,
    'DBInstanceClass': 'string',
    'DBInstanceIdentifier': 'string',
    'EngineVersion': 'string',
    'MasterUserPassword': 'string',
    'MultiAZ': True|False,
    'Port': 123
},
'PreferredBackupWindow': 'string',
'PreferredMaintenanceWindow': 'string',
'PubliclyAccessible': True|False,
'ReadReplicaDBInstanceIdentifiers': [
    'string',
],
'ReadReplicaSourceDBInstanceIdentifier': 'string',
'SecondaryAvailabilityZone': 'string',
'StatusInfos': [
    {
        'Message': 'string',
        'Normal': True|False,
        'Status': 'string',
        'StatusType': 'string'
    },
],
'VpcSecurityGroups': 'string'
},
'ResponseMetadata': {
    'RequestId': 'string'
}
}

```

### Response Structure

- (dict) –
  - DBInstance (dict) –
    - \* **AccountingType** (string) –
    - \* **AllocatedStorage** (integer) –
    - \* **AutoMinorVersionUpgrade** (boolean) –
    - \* **AvailabilityZone** (string) –
    - \* **BackupRetentionPeriod** (integer) –
    - \* **BinlogRetentionPeriod** (integer) –
    - \* **CACertificateIdentifier** (string) –
    - \* **DBInstanceClass** (string) –
    - \* **DBInstanceIdentifier** (string) –
    - \* **DBInstanceStatus** (string) –
    - \* **DBName** (string) –
    - \* **DBParameterGroups** (list) –
      - (dict) –
      - **DBParameterGroupName** (string) –

- **ParameterApplyStatus** (*string*) –
- \* **DBSecurityGroups** (*list*) –
  - (*dict*) –
  - **DBSecurityGroupName** (*string*) –
  - **Status** (*string*) –
- \* **Endpoint** (*dict*) –
  - **Address** (*string*) –
  - **NiftyPrivateAddress** (*string*) –
  - **Port** (*integer*) –
- \* **Engine** (*string*) –
- \* **EngineVersion** (*string*) –
- \* **ExternalReplicationInfo** (*dict*) –
  - **ExternalMasterAddress** (*string*) –
  - **ExternalReplicationMessage** (*string*) –
  - **ExternalReplicationStatus** (*string*) –
  - **ReplicationAddresses** (*list*) –
  - (*string*) –
  - **ReplicationPrivateAddresses** (*list*) –
  - (*string*) –
- \* **InstanceCreateTime** (*datetime*) –
- \* **LatestRestorableTime** (*datetime*) –
- \* **LicenseModel** (*string*) –
- \* **MasterUsername** (*string*) –
- \* **MultiAZ** (*boolean*) –
- \* **NextMonthAccountingType** (*string*) –
- \* **NiftyMasterPrivateAddress** (*string*) –
- \* **NiftyMultiAZType** (*string*) –
- \* **NiftyNetworkId** (*string*) –
- \* **NiftySlavePrivateAddress** (*string*) –
- \* **NiftyStorageType** (*integer*) –
- \* **OptionGroupMemberships** (*list*) –
  - (*dict*) –
  - **OptionGroupName** (*string*) –
  - **Status** (*string*) –
- \* **PendingModifiedValues** (*dict*) –
  - **AllocatedStorage** (*integer*) –
  - **BackupRetentionPeriod** (*integer*) –
  - **DBInstanceClass** (*string*) –
  - **DBInstanceIdentifier** (*string*) –
  - **EngineVersion** (*string*) –
  - **MasterUserPassword** (*string*) –
  - **MultiAZ** (*boolean*) –
  - **Port** (*integer*) –
- \* **PreferredBackupWindow** (*string*) –
- \* **PreferredMaintenanceWindow** (*string*) –
- \* **PubliclyAccessible** (*boolean*) –
- \* **ReadReplicaDBInstanceIdentifiers** (*list*) –
  - (*string*) –
- \* **ReadReplicaSourceDBInstanceIdentifier** (*string*) –
- \* **SecondaryAvailabilityZone** (*string*) –
- \* **StatusInfos** (*list*) –
  - (*dict*) –
  - **Message** (*string*) –
  - **Normal** (*boolean*) –

- **Status** (*string*) –
- **StatusType** (*string*) –
- \* **VpcSecurityGroups** (*string*) –
- **ResponseMetadata** (*dict*) –
- \* **RequestId** (*string*) –

*rdb* / Client / stop\_replication

## stop\_replication

`rdb.Client.stop_replication(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.stop_replication(
    DBInstanceIdentifier='string'
)
```

**Parameters** `DBInstanceIdentifier` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
    'DBInstance': {
        'AccountingType': 'string',
        'AllocatedStorage': 123,
        'AutoMinorVersionUpgrade': True|False,
        'AvailabilityZone': 'string',
        'BackupRetentionPeriod': 123,
        'BinlogRetentionPeriod': 123,
        'CACertificateIdentifier': 'string',
        'DBInstanceClass': 'string',
        'DBInstanceIdentifier': 'string',
        'DBInstanceStatus': 'string',
        'DBName': 'string',
        'DBParameterGroups': [
            {
                'DBParameterGroupName': 'string',
                'ParameterApplyStatus': 'string'
            },
        ],
        'DBSecurityGroups': [
            {
                'DBSecurityGroupName': 'string',
                'Status': 'string'
            },
        ],
        'Endpoint': {
            'Address': 'string',
            'NiftyPrivateAddress': 'string',
            'Port': 123
        },
        'Engine': 'string',
        'EngineVersion': 'string',
```

(continues on next page)

(continued from previous page)

```

'ExternalReplicationInfo': {
    'ExternalMasterAddress': 'string',
    'ExternalReplicationMessage': 'string',
    'ExternalReplicationStatus': 'string',
    'ReplicationAddresses': [
        'string',
    ],
    'ReplicationPrivateAddresses': [
        'string',
    ]
},
'InstanceCreateTime': datetime(2015, 1, 1),
'LatestRestorableTime': datetime(2015, 1, 1),
'LicenseModel': 'string',
'MasterUsername': 'string',
'MultiAZ': True|False,
'NextMonthAccountingType': 'string',
'NiftyMasterPrivateAddress': 'string',
'NiftyMultiAZType': 'string',
'NiftyNetworkId': 'string',
'NiftySlavePrivateAddress': 'string',
'NiftyStorageType': 123,
'OptionGroupMemberships': [
    {
        'OptionGroupName': 'string',
        'Status': 'string'
    },
],
'PendingModifiedValues': {
    'AllocatedStorage': 123,
    'BackupRetentionPeriod': 123,
    'DBInstanceClass': 'string',
    'DBInstanceIdentifier': 'string',
    'EngineVersion': 'string',
    'MasterUserPassword': 'string',
    'MultiAZ': True|False,
    'Port': 123
},
'PreferredBackupWindow': 'string',
'PreferredMaintenanceWindow': 'string',
'PubliclyAccessible': True|False,
'ReadReplicaDBInstanceIdentifiers': [
    'string',
],
'ReadReplicaSourceDBInstanceIdentifier': 'string',
'SecondaryAvailabilityZone': 'string',
'StatusInfos': [
    {
        'Message': 'string',
        'Normal': True|False,
        'Status': 'string',
        'StatusType': 'string'
    },
],
'VpcSecurityGroups': 'string'
},
'ResponseMetadata': {

```

(continues on next page)

(continued from previous page)

```

    'RequestId': 'string'
  }
}

```

**Response Structure**

- *(dict)* –
  - **DBInstance** *(dict)* –
    - \* **AccountingType** *(string)* –
    - \* **AllocatedStorage** *(integer)* –
    - \* **AutoMinorVersionUpgrade** *(boolean)* –
    - \* **AvailabilityZone** *(string)* –
    - \* **BackupRetentionPeriod** *(integer)* –
    - \* **BinlogRetentionPeriod** *(integer)* –
    - \* **CACertificateIdentifier** *(string)* –
    - \* **DBInstanceClass** *(string)* –
    - \* **DBInstanceIdentifier** *(string)* –
    - \* **DBInstanceStatus** *(string)* –
    - \* **DBName** *(string)* –
    - \* **DBParameterGroups** *(list)* –
      - *(dict)* –
      - **DBParameterGroupName** *(string)* –
      - **ParameterApplyStatus** *(string)* –
    - \* **DBSecurityGroups** *(list)* –
      - *(dict)* –
      - **DBSecurityGroupName** *(string)* –
      - **Status** *(string)* –
    - \* **Endpoint** *(dict)* –
      - **Address** *(string)* –
      - **NiftyPrivateAddress** *(string)* –
      - **Port** *(integer)* –
    - \* **Engine** *(string)* –
    - \* **EngineVersion** *(string)* –
    - \* **ExternalReplicationInfo** *(dict)* –
      - **ExternalMasterAddress** *(string)* –
      - **ExternalReplicationMessage** *(string)* –
      - **ExternalReplicationStatus** *(string)* –
      - **ReplicationAddresses** *(list)* –
        - *(string)* –
      - **ReplicationPrivateAddresses** *(list)* –
        - *(string)* –
    - \* **InstanceCreateTime** *(datetime)* –
    - \* **LatestRestorableTime** *(datetime)* –
    - \* **LicenseModel** *(string)* –
    - \* **MasterUsername** *(string)* –
    - \* **MultiAZ** *(boolean)* –
    - \* **NextMonthAccountingType** *(string)* –
    - \* **NiftyMasterPrivateAddress** *(string)* –
    - \* **NiftyMultiAZType** *(string)* –
    - \* **NiftyNetworkId** *(string)* –
    - \* **NiftySlavePrivateAddress** *(string)* –
    - \* **NiftyStorageType** *(integer)* –
    - \* **OptionGroupMemberships** *(list)* –
      - *(dict)* –

- **OptionGroupName** (*string*) –
- **Status** (*string*) –
- \* **PendingModifiedValues** (*dict*) –
  - **AllocatedStorage** (*integer*) –
  - **BackupRetentionPeriod** (*integer*) –
  - **DBInstanceClass** (*string*) –
  - **DBInstanceIdentifier** (*string*) –
  - **EngineVersion** (*string*) –
  - **MasterUserPassword** (*string*) –
  - **MultiAZ** (*boolean*) –
  - **Port** (*integer*) –
- \* **PreferredBackupWindow** (*string*) –
- \* **PreferredMaintenanceWindow** (*string*) –
- \* **PubliclyAccessible** (*boolean*) –
- \* **ReadReplicaDBInstanceIdentifiers** (*list*) –
  - (*string*) –
- \* **ReadReplicaSourceDBInstanceIdentifier** (*string*) –
- \* **SecondaryAvailabilityZone** (*string*) –
- \* **StatusInfos** (*list*) –
  - (*dict*) –
  - **Message** (*string*) –
  - **Normal** (*boolean*) –
  - **Status** (*string*) –
  - **StatusType** (*string*) –
- \* **VpcSecurityGroups** (*string*) –
- **ResponseMetadata** (*dict*) –
  - \* **RequestId** (*string*) –

*rd*b / Client / upgrade\_db\_engine\_version

## upgrade\_db\_engine\_version

`rd`b.Client.**upgrade\_db\_engine\_version** (*\*\*kwargs*)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.upgrade_db_engine_version(  
    AllowMajorVersionUpgrade=True|False,  
    DBInstanceIdentifier='string',  
    DBParameterGroupName='string',  
    EngineVersion='string',  
    PreUpgradeDBSnapshotIdentifier='string',  
    SkipPreUpgradeSnapshot=True|False  
)
```

### Parameters

- **AllowMajorVersionUpgrade** (*boolean*) –
- **DBInstanceIdentifier** (*string*) – [REQUIRED]
- **DBParameterGroupName** (*string*) –
- **EngineVersion** (*string*) – [REQUIRED]
- **PreUpgradeDBSnapshotIdentifier** (*string*) –
- **SkipPreUpgradeSnapshot** (*boolean*) –

**Return type** dict

**Returns**

## Response Syntax

```
{
  'DBInstance': {
    'AccountingType': 'string',
    'AllocatedStorage': 123,
    'AutoMinorVersionUpgrade': True|False,
    'AvailabilityZone': 'string',
    'BackupRetentionPeriod': 123,
    'BinlogRetentionPeriod': 123,
    'CACertificateIdentifier': 'string',
    'DBInstanceClass': 'string',
    'DBInstanceIdentifier': 'string',
    'DBInstanceStatus': 'string',
    'DBName': 'string',
    'DBParameterGroups': [
      {
        'DBParameterGroupName': 'string',
        'ParameterApplyStatus': 'string'
      },
    ],
    'DBSecurityGroups': [
      {
        'DBSecurityGroupName': 'string',
        'Status': 'string'
      },
    ],
    'Endpoint': {
      'Address': 'string',
      'NiftyPrivateAddress': 'string',
      'Port': 123
    },
    'Engine': 'string',
    'EngineVersion': 'string',
    'ExternalReplicationInfo': {
      'ExternalMasterAddress': 'string',
      'ExternalReplicationMessage': 'string',
      'ExternalReplicationStatus': 'string',
      'ReplicationAddresses': [
        'string',
      ],
      'ReplicationPrivateAddresses': [
        'string',
      ]
    },
    'InstanceCreateTime': datetime(2015, 1, 1),
    'LatestRestorableTime': datetime(2015, 1, 1),
    'LicenseModel': 'string',
    'MasterUsername': 'string',
    'MultiAZ': True|False,
    'NextMonthAccountingType': 'string',
    'NiftyMasterPrivateAddress': 'string',
    'NiftyMultiAZType': 'string',
    'NiftyNetworkId': 'string',
    'NiftySlavePrivateAddress': 'string',
    'NiftyStorageType': 123,
    'OptionGroupMemberships': [
      {

```

(continues on next page)

(continued from previous page)

```

        'OptionGroupName': 'string',
        'Status': 'string'
    },
],
'PendingModifiedValues': {
    'AllocatedStorage': 123,
    'BackupRetentionPeriod': 123,
    'DBInstanceClass': 'string',
    'DBInstanceIdentifier': 'string',
    'EngineVersion': 'string',
    'MasterUserPassword': 'string',
    'MultiAZ': True|False,
    'Port': 123
},
'PreferredBackupWindow': 'string',
'PreferredMaintenanceWindow': 'string',
'PubliclyAccessible': True|False,
'ReadReplicaDBInstanceIdentifiers': [
    'string',
],
'ReadReplicaSourceDBInstanceIdentifier': 'string',
'SecondaryAvailabilityZone': 'string',
'StatusInfos': [
    {
        'Message': 'string',
        'Normal': True|False,
        'Status': 'string',
        'StatusType': 'string'
    },
],
'VpcSecurityGroups': 'string'
},
'Marker': 'string',
'ResponseMetadata': {
    'RequestId': 'string'
}
}

```

### Response Structure

- (dict) –
  - DBInstance (dict) –
    - \* **AccountingType** (string) –
    - \* **AllocatedStorage** (integer) –
    - \* **AutoMinorVersionUpgrade** (boolean) –
    - \* **AvailabilityZone** (string) –
    - \* **BackupRetentionPeriod** (integer) –
    - \* **BinlogRetentionPeriod** (integer) –
    - \* **CACertificateIdentifier** (string) –
    - \* **DBInstanceClass** (string) –
    - \* **DBInstanceIdentifier** (string) –
    - \* **DBInstanceStatus** (string) –
    - \* **DBName** (string) –
    - \* **DBParameterGroups** (list) –
      - (dict) –
      - **DBParameterGroupName** (string) –
      - **ParameterApplyStatus** (string) –



- \* **DBSecurityGroups** (*list*) –
  - (*dict*) –
  - **DBSecurityGroupName** (*string*) –
  - **Status** (*string*) –
- \* **Endpoint** (*dict*) –
  - **Address** (*string*) –
  - **NiftyPrivateAddress** (*string*) –
  - **Port** (*integer*) –
- \* **Engine** (*string*) –
- \* **EngineVersion** (*string*) –
- \* **ExternalReplicationInfo** (*dict*) –
  - **ExternalMasterAddress** (*string*) –
  - **ExternalReplicationMessage** (*string*) –
  - **ExternalReplicationStatus** (*string*) –
  - **ReplicationAddresses** (*list*) –
  - (*string*) –
  - **ReplicationPrivateAddresses** (*list*) –
  - (*string*) –
- \* **InstanceCreateTime** (*datetime*) –
- \* **LatestRestorableTime** (*datetime*) –
- \* **LicenseModel** (*string*) –
- \* **MasterUsername** (*string*) –
- \* **MultiAZ** (*boolean*) –
- \* **NextMonthAccountingType** (*string*) –
- \* **NiftyMasterPrivateAddress** (*string*) –
- \* **NiftyMultiAZType** (*string*) –
- \* **NiftyNetworkId** (*string*) –
- \* **NiftySlavePrivateAddress** (*string*) –
- \* **NiftyStorageType** (*integer*) –
- \* **OptionGroupMemberships** (*list*) –
  - (*dict*) –
  - **OptionGroupName** (*string*) –
  - **Status** (*string*) –
- \* **PendingModifiedValues** (*dict*) –
  - **AllocatedStorage** (*integer*) –
  - **BackupRetentionPeriod** (*integer*) –
  - **DBInstanceClass** (*string*) –
  - **DBInstanceIdentifier** (*string*) –
  - **EngineVersion** (*string*) –
  - **MasterUserPassword** (*string*) –
  - **MultiAZ** (*boolean*) –
  - **Port** (*integer*) –
- \* **PreferredBackupWindow** (*string*) –
- \* **PreferredMaintenanceWindow** (*string*) –
- \* **PubliclyAccessible** (*boolean*) –
- \* **ReadReplicaDBInstanceIdentifiers** (*list*) –
  - (*string*) –
- \* **ReadReplicaSourceDBInstanceIdentifier** (*string*) –
- \* **SecondaryAvailabilityZone** (*string*) –
- \* **StatusInfos** (*list*) –
  - (*dict*) –
  - **Message** (*string*) –
  - **Normal** (*boolean*) –
  - **Status** (*string*) –

- **StatusType** (*string*) –
- \* **VpcSecurityGroups** (*string*) –
- **Marker** (*string*) –
- **ResponseMetadata** (*dict*) –
- \* **RequestId** (*string*) –

## 1.6.2 Client Exceptions

Client exceptions are available on a client instance via the `exceptions` property. For more detailed instructions and examples on the exact usage of client exceptions, see the error handling [user guide](#).

This client has no modeled exception classes.

## 1.6.3 Waiters

Waiters are available on a client instance via the `get_waiter` method. For more detailed instructions and examples on the usage or waiters, see the waiters [user guide](#).

The available waiters are:

*rdb* / Waiter / DBInstanceAvailable

### DBInstanceAvailable

**class** `rdb.Waiter.DBInstanceAvailable`

```
waiter = client.get_waiter('db_instance_available')
```

**wait** (*\*\*kwargs*)

Polls `rdb.Client.describe_db_instances()` every 40 seconds until a successful state is reached. An error is returned after 80 failed checks.

See also: [AWS API Documentation](#)

#### Request Syntax

```
waiter.wait(  
    DBInstanceIdentifier='string',  
    Filter='string',  
    FilterName='string',  
    FilterValue='string',  
    Filters=[  
        'string',  
    ],  
    Marker='string',  
    MaxRecords=123,  
    WaiterConfig={  
        'Delay': 123,  
        'MaxAttempts': 123  
    }  
)
```

#### Parameters

- **DBInstanceIdentifier** (*string*) –
- **Filter** (*string*) –

- **FilterName** (*string*) –
- **FilterValue** (*string*) –
- **Filters** (*list*) –
  - (*string*) –
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 40
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 80

**Returns** None

*rdb* / Waiter / DBInstanceDeleted

## DBInstanceDeleted

**class** `rdb.Waiter.DBInstanceDeleted`

```
waiter = client.get_waiter('db_instance_deleted')
```

**wait** (*\*\*kwargs*)

Polls `rdb.Client.describe_db_instances()` every 40 seconds until a successful state is reached. An error is returned after 80 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    DBInstanceIdentifier='string',
    Filter='string',
    FilterName='string',
    FilterValue='string',
    Filters=[
        'string',
    ],
    Marker='string',
    MaxRecords=123,
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **DBInstanceIdentifier** (*string*) –
- **Filter** (*string*) –
- **FilterName** (*string*) –
- **FilterValue** (*string*) –
- **Filters** (*list*) –
  - (*string*) –
- **Marker** (*string*) –

- **MaxRecords** (*integer*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 40
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 80

**Returns** None

*rdb* / Waiter / DBInstanceExists

## DBInstanceExists

**class** *rdb.Waiter.DBInstanceExists*

```
waiter = client.get_waiter('db_instance_exists')
```

**wait** (*\*\*kwargs*)

Polls *rdb.Client.describe\_db\_instances()* every 40 seconds until a successful state is reached. An error is returned after 80 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(  
    DBInstanceIdentifier='string',  
    Filter='string',  
    FilterName='string',  
    FilterValue='string',  
    Filters=[  
        'string',  
    ],  
    Marker='string',  
    MaxRecords=123,  
    WaiterConfig={  
        'Delay': 123,  
        'MaxAttempts': 123  
    }  
)
```

### Parameters

- **DBInstanceIdentifier** (*string*) –
- **Filter** (*string*) –
- **FilterName** (*string*) –
- **FilterValue** (*string*) –
- **Filters** (*list*) –
  - (*string*) –
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 40

– **MaxAttempts** (*integer*) –

The maximum number of attempts to be made. Default: 80

**Returns** None

*rdb* / Waiter / DBInstanceFailed

## DBInstanceFailed

**class** `rdb.Waiter.DBInstanceFailed`

```
waiter = client.get_waiter('db_instance_failed')
```

**wait** (*\*\*kwargs*)

Polls `rdb.Client.describe_db_instances()` every 40 seconds until a successful state is reached. An error is returned after 80 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    DBInstanceIdentifier='string',
    Filter='string',
    FilterName='string',
    FilterValue='string',
    Filters=[
        'string',
    ],
    Marker='string',
    MaxRecords=123,
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **DBInstanceIdentifier** (*string*) –
- **Filter** (*string*) –
- **FilterName** (*string*) –
- **FilterValue** (*string*) –
- **Filters** (*list*) –
  - (*string*) –
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 40
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 80

**Returns** None

*rdb* / Waiter / DBInstanceStorageFull

## DBInstanceStorageFull

**class** rdb.Waiter.DBInstanceStorageFull

```
waiter = client.get_waiter('db_instance_storage_full')
```

**wait** (\*\*kwargs)

Polls `rdb.Client.describe_db_instances()` every 40 seconds until a successful state is reached. An error is returned after 80 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    DBInstanceIdentifier='string',
    Filter='string',
    FilterName='string',
    FilterValue='string',
    Filters=[
        'string',
    ],
    Marker='string',
    MaxRecords=123,
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **DBInstanceIdentifier** (*string*) –
- **Filter** (*string*) –
- **FilterName** (*string*) –
- **FilterValue** (*string*) –
- **Filters** (*list*) –
  - (*string*) –
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 40
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 80

**Returns** None

*rdb* / Waiter / DBSecurityGroupDeleted

## DBSecurityGroupDeleted

**class** rdb.Waiter.DBSecurityGroupDeleted

```
waiter = client.get_waiter('db_security_group_deleted')
```

**wait** (*\*\*kwargs*)

Polls `rdb.Client.describe_db_security_groups()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    DBSecurityGroupName='string',
    Filter='string',
    FilterName='string',
    FilterValue='string',
    Filters=[
        'string',
    ],
    Marker='string',
    MaxRecords=123,
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **DBSecurityGroupName** (*string*) –
- **Filter** (*string*) –
- **FilterName** (*string*) –
- **FilterValue** (*string*) –
- **Filters** (*list*) –
  - (*string*) –
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None

*rdb* / Waiter / DBSecurityGroupEC2SecurityGroupsAuthFailed

## DBSecurityGroupEC2SecurityGroupsAuthFailed

**class** `rdb.Waiter.DBSecurityGroupEC2SecurityGroupsAuthFailed`

```
waiter = client.get_waiter('db_security_group_ec2_security_groups_auth_failed')
```

**wait** (*\*\*kwargs*)

Polls `rdb.Client.describe_db_security_groups()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(  
    DBSecurityGroupName='string',  
    Filter='string',  
    FilterName='string',  
    FilterValue='string',  
    Filters=[  
        'string',  
    ],  
    Marker='string',  
    MaxRecords=123,  
    WaiterConfig={  
        'Delay': 123,  
        'MaxAttempts': 123  
    }  
)
```

### Parameters

- **DBSecurityGroupName** (*string*) –
- **Filter** (*string*) –
- **FilterName** (*string*) –
- **FilterValue** (*string*) –
- **Filters** (*list*) –
  - (*string*) –
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –  
The maximum number of attempts to be made. Default: 40

**Returns** None

*rdb* / Waiter / DBSecurityGroupEC2SecurityGroupsAuthorized

### DBSecurityGroupEC2SecurityGroupsAuthorized

**class** `rdb.Waiter.DBSecurityGroupEC2SecurityGroupsAuthorized`

```
waiter = client.get_waiter('db_security_group_ec2_security_groups_authorized')
```

**wait** (*\*\*kwargs*)

Polls `rdb.Client.describe_db_security_groups()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax



```

waiter.wait(
    DBSecurityGroupName='string',
    Filter='string',
    FilterName='string',
    FilterValue='string',
    Filters=[
        'string',
    ],
    Marker='string',
    MaxRecords=123,
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)

```

### Parameters

- **DBSecurityGroupName** (*string*) –
- **Filter** (*string*) –
- **FilterName** (*string*) –
- **FilterValue** (*string*) –
- **Filters** (*list*) –
  - (*string*) –
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
    - The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
    - The maximum number of attempts to be made. Default: 40

**Returns** None

*rdb* / Waiter / DBSecurityGroupEC2SecurityGroupsEmptied

## DBSecurityGroupEC2SecurityGroupsEmptied

**class** *rdb*.Waiter.DBSecurityGroupEC2SecurityGroupsEmptied

```
waiter = client.get_waiter('db_security_group_ec2_security_groups_emptied')
```

**wait** (*\*\*kwargs*)

Polls *rdb.Client.describe\_db\_security\_groups()* every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```

waiter.wait(
    DBSecurityGroupName='string',
    Filter='string',
    FilterName='string',

```

(continues on next page)

(continued from previous page)

```

FilterValue='string',
Filters=[
    'string',
],
Marker='string',
MaxRecords=123,
WaiterConfig={
    'Delay': 123,
    'MaxAttempts': 123
}
)

```

**Parameters**

- **DBSecurityGroupName** (*string*) –
- **Filter** (*string*) –
- **FilterName** (*string*) –
- **FilterValue** (*string*) –
- **Filters** (*list*) –
  - (*string*) –
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None*rdb* / Waiter / DBSecurityGroupEC2SecurityGroupsRevokeFailed**DBSecurityGroupEC2SecurityGroupsRevokeFailed****class** *rdb*.Waiter.DBSecurityGroupEC2SecurityGroupsRevokeFailed

```
waiter = client.get_waiter('db_security_group_ec2_security_groups_revoke_failed')
```

**wait** (*\*\*kwargs*)

Polls *rdb.Client.describe\_db\_security\_groups()* every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```

waiter.wait(
    DBSecurityGroupName='string',
    Filter='string',
    FilterName='string',
    FilterValue='string',
    Filters=[
        'string',
    ],
)

```

(continues on next page)

(continued from previous page)

```

Marker='string',
MaxRecords=123,
WaiterConfig={
    'Delay': 123,
    'MaxAttempts': 123
}
)

```

**Parameters**

- **DBSecurityGroupName** (*string*) –
- **Filter** (*string*) –
- **FilterName** (*string*) –
- **FilterValue** (*string*) –
- **Filters** (*list*) –
  - (*string*) –
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None*rdb* / Waiter / DBSecurityGroupExists**DBSecurityGroupExists****class** *rdb*.Waiter.DBSecurityGroupExists

```
waiter = client.get_waiter('db_security_group_exists')
```

**wait** (*\*\*kwargs*)

Polls *rdb.Client.describe\_db\_security\_groups()* every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```

waiter.wait(
    DBSecurityGroupName='string',
    Filter='string',
    FilterName='string',
    FilterValue='string',
    Filters=[
        'string',
    ],
    Marker='string',
    MaxRecords=123,
    WaiterConfig={
        'Delay': 123,

```

(continues on next page)

(continued from previous page)

```

        'MaxAttempts': 123
    }
)

```

**Parameters**

- **DBSecurityGroupName** (*string*) –
- **Filter** (*string*) –
- **FilterName** (*string*) –
- **FilterValue** (*string*) –
- **Filters** (*list*) –
  - (*string*) –
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None*rdb* / Waiter / DBSecurityGroupIPRangesAuthFailed**DBSecurityGroupIPRangesAuthFailed****class** *rdb*.Waiter.DBSecurityGroupIPRangesAuthFailed

```
waiter = client.get_waiter('db_security_group_ip_ranges_auth_failed')
```

**wait** (*\*\*kwargs*)

Polls *rdb.Client.describe\_db\_security\_groups()* every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

**Request Syntax**

```

waiter.wait(
    DBSecurityGroupName='string',
    Filter='string',
    FilterName='string',
    FilterValue='string',
    Filters=[
        'string',
    ],
    Marker='string',
    MaxRecords=123,
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)

```

**Parameters**

- **DBSecurityGroupName** (*string*) –
- **Filter** (*string*) –
- **FilterName** (*string*) –
- **FilterValue** (*string*) –
- **Filters** (*list*) –
  - (*string*) –
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
    - The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
    - The maximum number of attempts to be made. Default: 40

**Returns** None

*rdb* / Waiter / DBSecurityGroupIPRangesAuthorized

## DBSecurityGroupIPRangesAuthorized

**class** `rdb.Waiter.DBSecurityGroupIPRangesAuthorized`

```
waiter = client.get_waiter('db_security_group_ip_ranges_authorized')
```

**wait** (*\*\*kwargs*)

Polls `rdb.Client.describe_db_security_groups()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    DBSecurityGroupName='string',
    Filter='string',
    FilterName='string',
    FilterValue='string',
    Filters=[
        'string',
    ],
    Marker='string',
    MaxRecords=123,
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **DBSecurityGroupName** (*string*) –
- **Filter** (*string*) –
- **FilterName** (*string*) –
- **FilterValue** (*string*) –
- **Filters** (*list*) –

- (string) –
- **Marker** (string) –
- **MaxRecords** (integer) –
- **WaiterConfig** (dict) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (integer) –  
The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (integer) –  
The maximum number of attempts to be made. Default: 40

**Returns** None

*rdb* / Waiter / DBSecurityGroupIPRangesEmptied

### DBSecurityGroupIPRangesEmptied

**class** `rdb.Waiter.DBSecurityGroupIPRangesEmptied`

```
waiter = client.get_waiter('db_security_group_ip_ranges_emptied')
```

**wait** (\*\*kwargs)

Polls `rdb.Client.describe_db_security_groups()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

#### Request Syntax

```
waiter.wait(  
    DBSecurityGroupName='string',  
    Filter='string',  
    FilterName='string',  
    FilterValue='string',  
    Filters=[  
        'string',  
    ],  
    Marker='string',  
    MaxRecords=123,  
    WaiterConfig={  
        'Delay': 123,  
        'MaxAttempts': 123  
    }  
)
```

#### Parameters

- **DBSecurityGroupName** (string) –
- **Filter** (string) –
- **FilterName** (string) –
- **FilterValue** (string) –
- **Filters** (list) –
  - (string) –
- **Marker** (string) –
- **MaxRecords** (integer) –
- **WaiterConfig** (dict) – A dictionary that provides parameters to control waiting behavior.

– **Delay** (*integer*) –

The amount of time in seconds to wait between attempts. Default: 20

– **MaxAttempts** (*integer*) –

The maximum number of attempts to be made. Default: 40

**Returns** None

*rdb* / Waiter / DBSecurityGroupIPRangesRevokeFailed

## DBSecurityGroupIPRangesRevokeFailed

**class** `rdb.Waiter.DBSecurityGroupIPRangesRevokeFailed`

```
waiter = client.get_waiter('db_security_group_ip_ranges_revoke_failed')
```

**wait** (*\*\*kwargs*)

Polls `rdb.Client.describe_db_security_groups()` every 20 seconds until a successful state is reached. An error is returned after 40 failed checks.

See also: [AWS API Documentation](#)

### Request Syntax

```
waiter.wait(
    DBSecurityGroupName='string',
    Filter='string',
    FilterName='string',
    FilterValue='string',
    Filters=[
        'string',
    ],
    Marker='string',
    MaxRecords=123,
    WaiterConfig={
        'Delay': 123,
        'MaxAttempts': 123
    }
)
```

### Parameters

- **DBSecurityGroupName** (*string*) –
- **Filter** (*string*) –
- **FilterName** (*string*) –
- **FilterValue** (*string*) –
- **Filters** (*list*) –
  - (*string*) –
- **Marker** (*string*) –
- **MaxRecords** (*integer*) –
- **WaiterConfig** (*dict*) – A dictionary that provides parameters to control waiting behavior.
  - **Delay** (*integer*) –
 

The amount of time in seconds to wait between attempts. Default: 20
  - **MaxAttempts** (*integer*) –
 

The maximum number of attempts to be made. Default: 40

**Returns** None

## 1.7 script

### 1.7.1 Client

**class** `script.Client`

A low-level client representing NIFCLOUD Script

```
client = session.create_client('script')
```

These are the available methods:

*script* / Client / `can_paginate`

#### `can_paginate`

`script.Client.can_paginate(operation_name)`

Check if an operation can be paginated.

**Parameters** `operation_name` (*string*) – The operation name. This is the same name as the method name on the client. For example, if the method name is `create_foo`, and you'd normally invoke the operation as `client.create_foo(**kwargs)`, if the `create_foo` operation can be paginated, you can use the call `client.get_paginator("create_foo")`.

**Returns** True if the operation can be paginated, False otherwise.

*script* / Client / `close`

#### `close`

`script.Client.close()`

Closes underlying endpoint connections.

*script* / Client / `execute_script`

#### `execute_script`

`script.Client.execute_script(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

##### Request Syntax

```
response = client.execute_script(  
    Body='string',  
    Header='string',  
    Method='string',  
    Query='string',  
    ScriptIdentifier='string'  
)
```

##### Parameters

- **Body** (*string*) –
- **Header** (*string*) –
- **Method** (*string*) – [REQUIRED]
- **Query** (*string*) –
- **ScriptIdentifier** (*string*) – [REQUIRED]



**Return type** dict

**Returns**

#### Response Syntax

```
{
  'Result': {
    'RequestBody': 'string',
    'RequestHeader': 'string',
    'RequestQuery': 'string',
    'ResponseData': 'string',
    'ResponseHeader': 'string',
    'ResponseStatus': 123,
    'ScriptIdentifier': 'string'
  }
}
```

#### Response Structure

- (dict) –
  - **Result** (dict) –
    - \* **RequestBody** (string) –
    - \* **RequestHeader** (string) –
    - \* **RequestQuery** (string) –
    - \* **ResponseData** (string) –
    - \* **ResponseHeader** (string) –
    - \* **ResponseStatus** (integer) –
    - \* **ScriptIdentifier** (string) –

*script* / Client / get\_paginator

### get\_paginator

`script.Client.get_paginator(operation_name)`

Create a paginator for an operation.

**Parameters** **operation\_name** (string) – The operation name. This is the same name as the method name on the client. For example, if the method name is `create_foo`, and you'd normally invoke the operation as `client.create_foo(**kwargs)`, if the `create_foo` operation can be paginated, you can use the call `client.get_paginator("create_foo").`

**Raises** **OperationNotPageableError** – Raised if the operation is not pageable. You can use the `client.can_paginate` method to check if an operation is pageable.

**Return type** L{botocore.paginate.Paginator}

**Returns** A paginator object.

*script* / Client / get\_waiter

### get\_waiter

`script.Client.get_waiter(waiter_name)`

Returns an object that can wait for some condition.

**Parameters** **waiter\_name** (str) – The name of the waiter to get. See the waiters section of the service docs for a list of available waiters.

**Returns** The specified waiter object.

**Return type** botocore.waiter.Waiter

## 1.7.2 Client Exceptions

Client exceptions are available on a client instance via the `exceptions` property. For more detailed instructions and examples on the exact usage of client exceptions, see the error handling [user guide](#).

This client has no modeled exception classes.

## 1.8 serviceactivity

### 1.8.1 Client

**class** `serviceactivity.Client`

A low-level client representing NIFCLOUD Service Activity (service-activity)

```
client = session.create_client('service-activity')
```

These are the available methods:

[serviceactivity](#) / Client / `can_paginate`

#### `can_paginate`

`serviceactivity.Client.can_paginate(operation_name)`

Check if an operation can be paginated.

**Parameters** `operation_name` (*string*) – The operation name. This is the same name as the method name on the client. For example, if the method name is `create_foo`, and you'd normally invoke the operation as `client.create_foo(**kwargs)`, if the `create_foo` operation can be paginated, you can use the call `client.get_paginator("create_foo")`.

**Returns** True if the operation can be paginated, False otherwise.

[serviceactivity](#) / Client / `close`

#### `close`

`serviceactivity.Client.close()`

Closes underlying endpoint connections.

[serviceactivity](#) / Client / `describe_event_attributes`

#### `describe_event_attributes`

`serviceactivity.Client.describe_event_attributes(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

##### Request Syntax

```
response = client.describe_event_attributes(
    Location='string',
    Mode='user'|'all',
    YearMonth='string'
)
```

**Parameters**

- **Location** (*string*) –
- **Mode** (*string*) –
- **YearMonth** (*string*) – [REQUIRED]

**Return type** dict

**Returns****Response Syntax**

```
{
  'Data': {
    'Event': [
      {
        'AffectedServices': [
          {
            'EndAt': 'string',
            'Influence': 'string',
            'Location': 'string',
            'Menu': 'string',
            'Number': 123,
            'Resources': [
              {
                'DiskName': 'string',
                'ResourceName': 'string',
                'ResourceType': 'string'
              }
            ],
            'Service': 'string',
            'StartAt': 'string',
            'Status': 'string'
          }
        ],
        'EndAt': 'string',
        'EventHistories': [
          {
            'Date': 'string',
            'Message': 'string'
          }
        ],
        'EventID': 'string',
        'EventStatus': 'string',
        'StartAt': 'string'
      }
    ],
    'Mode': 'string',
    'TargetDate': 'string'
  },
  'Datetime': 'string',
  'RequestID': 'string'
}
```

**Response Structure**

- (*dict*) –
  - **Data** (*dict*) –
    - \* **Event** (*list*) –
      - (*dict*) –
      - **AffectedServices** (*list*) –
      - (*dict*) –

- **EndAt** (*string*) –
- **Influence** (*string*) –
- **Location** (*string*) –
- **Menu** (*string*) –
- **Number** (*integer*) –
- **Resources** (*list*) –
- (*dict*) –
- **DiskName** (*string*) –
- **ResourceName** (*string*) –
- **ResourceType** (*string*) –
- **Service** (*string*) –
- **StartAt** (*string*) –
- **Status** (*string*) –
- **EndAt** (*string*) –
- **EventHistories** (*list*) –
- (*dict*) –
- **Date** (*string*) –
- **Message** (*string*) –
- **EventID** (*string*) –
- **EventStatus** (*string*) –
- **StartAt** (*string*) –
- \* **Mode** (*string*) –
- \* **TargetDate** (*string*) –
- **Datetime** (*string*) –
- **RequestID** (*string*) –

*serviceactivity* / Client / describe\_event\_calendar

## describe\_event\_calendar

`serviceactivity.Client.describe_event_calendar(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.describe_event_calendar(
    Mode='user'|'all',
    YearMonth='string'
)
```

### Parameters

- **Mode** (*string*) –
- **YearMonth** (*string*) – [REQUIRED]

**Return type** dict

### Returns

### Response Syntax

```
{
    'Data': {
        'Calendar': [
            {
                'CancelMaintenance': 'string',
                'CompletedMaintenance': 'string',
                'Day': 'string',
```

(continues on next page)

(continued from previous page)

```

        'DayOfWeek': 'string',
        'Information': 'string',
        'Maintenance': 'string',
        'NoTroubleImpact': 'string',
        'RecoveredTrouble': 'string',
        'Trouble': 'string'
    },
],
'Mode': 'string',
'TargetDate': 'string'
},
'Datetime': 'string',
'RequestID': 'string'
}

```

**Response Structure**

- (dict) –
  - **Data** (dict) –
    - \* **Calendar** (list) –
      - (dict) –
      - **CancelMaintenance** (string) –
      - **CompletedMaintenance** (string) –
      - **Day** (string) –
      - **DayOfWeek** (string) –
      - **Information** (string) –
      - **Maintenance** (string) –
      - **NoTroubleImpact** (string) –
      - **RecoveredTrouble** (string) –
      - **Trouble** (string) –
    - \* **Mode** (string) –
    - \* **TargetDate** (string) –
  - **Datetime** (string) –
  - **RequestID** (string) –

*serviceactivity* / Client / describe\_service\_statuses

**describe\_service\_statuses**

`serviceactivity.Client.describe_service_statuses (**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```

response = client.describe_service_statuses(
    Mode='user' | 'all'
)

```

**Parameters** **Mode** (string) –

**Return type** dict

**Returns**

**Response Syntax**

```

{
    'Data': {

```

(continues on next page)

(continued from previous page)

```

        'ServiceMenu': [
            {
                'MaintenanceStatus': 'string',
                'Name': 'string',
                'NormalStatus': 'string',
                'Services': [
                    {
                        'Name': 'string',
                        'Statuses': [
                            {
                                'Location': 'string',
                                'MaintenanceStatus': 'string',
                                'NormalStatus': 'string',
                                'TroubleStatus': 'string'
                            }
                        ],
                    }
                ],
            },
            'TroubleStatus': 'string'
        ],
        'Datetime': 'string',
        'RequestID': 'string'
    }

```

**Response Structure**

- (dict) –
  - **Data** (dict) –
    - \* **ServiceMenu** (list) –
      - (dict) –
      - **MaintenanceStatus** (string) –
      - **Name** (string) –
      - **NormalStatus** (string) –
      - **Services** (list) –
        - (dict) –
        - **Name** (string) –
        - **Statuses** (list) –
          - (dict) –
          - **Location** (string) –
          - **MaintenanceStatus** (string) –
          - **NormalStatus** (string) –
          - **TroubleStatus** (string) –
          - **TroubleStatus** (string) –
    - **Datetime** (string) –
    - **RequestID** (string) –

*serviceactivity* / Client / get\_paginator

**get\_paginator**

`serviceactivity.Client.get_paginator(operation_name)`

Create a paginator for an operation.

**Parameters** `operation_name` (string) – The operation name. This is the same name as the method name on the client. For example, if the method name is `create_foo`,

and you'd normally invoke the operation as `client.create_foo(**kwargs)`, if the `create_foo` operation can be paginated, you can use the call `client.get_paginator("create_foo")`.

**Raises** `OperationNotPageableError` – Raised if the operation is not pageable. You can use the `client.can_paginate` method to check if an operation is pageable.

**Return type** `L{botocore.paginate.Paginator}`

**Returns** A paginator object.

*serviceactivity* / Client / `get_waiter`

## get\_waiter

`serviceactivity.Client.get_waiter(waiter_name)`

Returns an object that can wait for some condition.

**Parameters** `waiter_name` (*str*) – The name of the waiter to get. See the waiters section of the service docs for a list of available waiters.

**Returns** The specified waiter object.

**Return type** `botocore.waiter.Waiter`

## 1.8.2 Client Exceptions

Client exceptions are available on a client instance via the `exceptions` property. For more detailed instructions and examples on the exact usage of client exceptions, see the error handling [user guide](#).

This client has no modeled exception classes.

## 1.9 storage

### 1.9.1 Client

**class** `storage.Client`

A low-level client representing NIFCLOUD Object Storage Service

```
client = session.create_client('storage')
```

These are the available methods:

*storage* / Client / `abort_multipart_upload`

## abort\_multipart\_upload

`storage.Client.abort_multipart_upload(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.abort_multipart_upload(
    Bucket='string',
    Object='string',
    UploadId='string'
)
```

### Parameters

- **Bucket** (*string*) – [REQUIRED]
- **Object** (*string*) – [REQUIRED]
- **UploadId** (*string*) – [REQUIRED]

**Returns** None

*storage* / Client / can\_paginate

## can\_paginate

`storage.Client.can_paginate(operation_name)`

Check if an operation can be paginated.

**Parameters** **operation\_name** (*string*) – The operation name. This is the same name as the method name on the client. For example, if the method name is `create_foo`, and you'd normally invoke the operation as `client.create_foo(**kwargs)`, if the `create_foo` operation can be paginated, you can use the call `client.get_paginator("create_foo")`.

**Returns** True if the operation can be paginated, False otherwise.

*storage* / Client / close

## close

`storage.Client.close()`

Closes underlying endpoint connections.

*storage* / Client / complete\_multipart\_upload

## complete\_multipart\_upload

`storage.Client.complete_multipart_upload(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.complete_multipart_upload(
    Bucket='string',
    CompleteMultipartUpload={
        'ListOfRequestPart': [
            {
                'ETag': 'string',
                'PartNumber': 123
            },
        ],
    },
    Object='string',
    UploadId='string'
)
```

### Parameters

- **Bucket** (*string*) – [REQUIRED]
- **CompleteMultipartUpload** (*dict*) – [REQUIRED]
  - **ListOfRequestPart** (*list*) – [REQUIRED]
    - \* (*dict*) –
      - **ETag** (*string*) – [REQUIRED]
      - **PartNumber** (*integer*) – [REQUIRED]



- **Object** (*string*) – [REQUIRED]
- **UploadId** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'Bucket': 'string',
    'ContentType': 'string',
    'ETag': 'string',
    'Key': 'string',
    'Location': 'string',
    'XAmzExpiration': 'string',
    'XAmzServerSideEncryption': 'string',
    'XAmzServerSideEncryptionAwsKmsKeyId': 'string',
    'XAmzServerSideEncryptionCustomerAlgorithm': 'string',
    'XAmzVersionId': 'string'
}
```

#### Response Structure

- (*dict*) –
  - **Bucket** (*string*) –
  - **ContentType** (*string*) –
  - **ETag** (*string*) –
  - **Key** (*string*) –
  - **Location** (*string*) –
  - **XAmzExpiration** (*string*) –
  - **XAmzServerSideEncryption** (*string*) –
  - **XAmzServerSideEncryptionAwsKmsKeyId** (*string*) –
  - **XAmzServerSideEncryptionCustomerAlgorithm** (*string*) –
  - **XAmzVersionId** (*string*) –

*storage* / Client / delete\_bucket

### delete\_bucket

`storage.Client.delete_bucket(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.delete_bucket(
    Bucket='string'
)
```

**Parameters** **Bucket** (*string*) – [REQUIRED]

**Returns** None

*storage* / Client / delete\_bucket\_cors

### delete\_bucket\_cors

`storage.Client.delete_bucket_cors(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.delete_bucket_cors(  
    Bucket='string'  
)
```

**Parameters** **Bucket** (*string*) – [REQUIRED]

**Returns** None

*storage* / Client / delete\_bucket\_lifecycle

## delete\_bucket\_lifecycle

`storage.Client.delete_bucket_lifecycle(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_bucket_lifecycle(  
    Bucket='string'  
)
```

**Parameters** **Bucket** (*string*) – [REQUIRED]

**Returns** None

*storage* / Client / delete\_bucket\_policy

## delete\_bucket\_policy

`storage.Client.delete_bucket_policy(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_bucket_policy(  
    Bucket='string'  
)
```

**Parameters** **Bucket** (*string*) – [REQUIRED]

**Returns** None

*storage* / Client / delete\_bucket\_tagging

## delete\_bucket\_tagging

`storage.Client.delete_bucket_tagging(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_bucket_tagging(  
    Bucket='string'  
)
```

**Parameters** **Bucket** (*string*) – [REQUIRED]

**Returns** None

*storage* / Client / delete\_multiple\_objects

## delete\_multiple\_objects

`storage.Client.delete_multiple_objects(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```

response = client.delete_multiple_objects(
    Bucket='string',
    ContentMd5='string',
    Delete={
        'ListOfRequestObject': [
            {
                'Key': 'string',
                'Quiet': True|False,
                'VersionId': 'string'
            },
        ]
    }
)

```

### Parameters

- **Bucket** (*string*) – [REQUIRED]
- **ContentMd5** (*string*) – [REQUIRED]
- **Delete** (*dict*) – [REQUIRED]
  - **ListOfRequestObject** (*list*) – [REQUIRED]
    - \* (*dict*) –
      - **Key** (*string*) – [REQUIRED]
      - **Quiet** (*boolean*) –
      - **VersionId** (*string*) –

**Return type** dict

### Returns

### Response Syntax

```

{
    'ContentType': 'string',
    'Deleted': [
        {
            'Key': 'string',
            'VersionId': 'string'
        },
    ]
}

```

### Response Structure

- (*dict*) –
  - **ContentType** (*string*) –
  - **Deleted** (*list*) –
    - \* (*dict*) –
      - **Key** (*string*) –
      - **VersionId** (*string*) –

*storage* / Client / delete\_object

## delete\_object

`storage.Client.delete_object(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_object(  
    Bucket='string',  
    Object='string',  
    VersionId='string'  
)
```

#### Parameters

- **Bucket** (*string*) – [REQUIRED]
- **Object** (*string*) – [REQUIRED]
- **VersionId** (*string*) –

**Return type** dict

#### Returns

#### Response Syntax

```
{  
    'XAmzVersionId': 'string'  
}
```

#### Response Structure

- (*dict*) –
  - **XAmzVersionId** (*string*) –

*storage* / Client / delete\_object\_tagging

## delete\_object\_tagging

`storage.Client.delete_object_tagging(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.delete_object_tagging(  
    Bucket='string',  
    Object='string',  
    VersionId='string'  
)
```

#### Parameters

- **Bucket** (*string*) – [REQUIRED]
- **Object** (*string*) – [REQUIRED]
- **VersionId** (*string*) –

**Return type** dict

#### Returns

#### Response Syntax

```
{  
    'XAmzVersionId': 'string'  
}
```

**Response Structure**

- (*dict*) –
  - **XAmzVersionId** (*string*) –

*storage* / Client / get\_bucket

**get\_bucket**

`storage.Client.get_bucket(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.get_bucket(
    Bucket='string',
    Delimiter='string',
    EncodingType='string',
    Marker='string',
    MaxKeys='string',
    Prefix='string'
)
```

**Parameters**

- **Bucket** (*string*) – **[REQUIRED]**
- **Delimiter** (*string*) –
- **EncodingType** (*string*) –
- **Marker** (*string*) –
- **MaxKeys** (*string*) –
- **Prefix** (*string*) –

**Return type** dict

**Returns****Response Syntax**

```
{
  'CommonPrefixes': 'string',
  'ContentType': 'string',
  'Contents': [
    {
      'DisplayName': 'string',
      'ETag': 'string',
      'ID': 'string',
      'Key': 'string',
      'LastModified': datetime(2015, 1, 1),
      'Owner': {
        'DisplayName': 'string',
        'ID': 'string'
      },
      'Size': 'string',
      'StorageClass': 'string'
    },
  ],
  'Delimiter': 'string',
  'EncodingType': 'string',
  'IsTruncated': True|False,
  'Marker': 'string',
  'MaxKeys': 'string',
```

(continues on next page)

(continued from previous page)

```
'Name': 'string',
'NextMarker': 'string',
'Prefix': 'string'
}
```

**Response Structure**

- (dict) –
  - **CommonPrefixes** (string) –
  - **ContentType** (string) –
  - **Contents** (list) –
    - \* (dict) –
      - **DisplayName** (string) –
      - **ETag** (string) –
      - **ID** (string) –
      - **Key** (string) –
      - **LastModified** (datetime) –
      - **Owner** (dict) –
      - **DisplayName** (string) –
      - **ID** (string) –
      - **Size** (string) –
      - **StorageClass** (string) –
  - **Delimiter** (string) –
  - **EncodingType** (string) –
  - **IsTruncated** (boolean) –
  - **Marker** (string) –
  - **MaxKeys** (string) –
  - **Name** (string) –
  - **NextMarker** (string) –
  - **Prefix** (string) –

*storage* / Client / get\_bucket\_acl

**get\_bucket\_acl**

`storage.Client.get_bucket_acl(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.get_bucket_acl(
    Bucket='string'
)
```

**Parameters** **Bucket** (string) – [REQUIRED]

**Return type** dict

**Returns**

**Response Syntax**

```
{
  'AccessControlList': [
    {
      'Grantee': {
        'DisplayName': 'string',
```

(continues on next page)

(continued from previous page)

```

        'ID': 'string'
    },
    'Permission': 'string'
},
],
'ContentType': 'string',
'Owner': {
    'DisplayName': 'string',
    'ID': 'string'
}
}

```

**Response Structure**

- *(dict)* –
  - **AccessControlList** (*list*) –
    - \* *(dict)* –
      - **Grantee** (*dict*) –
      - **DisplayName** (*string*) –
      - **ID** (*string*) –
      - **Permission** (*string*) –
  - **ContentType** (*string*) –
  - **Owner** (*dict*) –
    - \* **DisplayName** (*string*) –
    - \* **ID** (*string*) –

*storage* / Client / `get_bucket_consistency`**get\_bucket\_consistency**`storage.Client.get_bucket_consistency(**kwargs)`See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```

response = client.get_bucket_consistency(
    Bucket='string'
)

```

**Parameters** **Bucket** (*string*) – [REQUIRED]**Return type** dict**Returns****Response Syntax**

```

{
    'Consistency': 'string',
    'ContentType': 'string'
}

```

**Response Structure**

- *(dict)* –
  - **Consistency** (*string*) –
  - **ContentType** (*string*) –

*storage* / Client / `get_bucket_cors`

## get\_bucket\_cors

`storage.Client.get_bucket_cors(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.get_bucket_cors(  
    Bucket='string'  
)
```

**Parameters** `Bucket` (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'CORSRule': [  
        {  
            'AllowedHeader': 'string',  
            'AllowedMethod': [  
                'string',  
            ],  
            'AllowedOrigin': 'string',  
            'ExposeHeader': 'string',  
            'MaxAgeSeconds': 123  
        },  
    ],  
    'ContentType': 'string'  
}
```

### Response Structure

- (*dict*) –
  - **CORSRule** (*list*) –
    - \* (*dict*) –
      - **AllowedHeader** (*string*) –
      - **AllowedMethod** (*list*) –
      - (*string*) –
      - **AllowedOrigin** (*string*) –
      - **ExposeHeader** (*string*) –
      - **MaxAgeSeconds** (*integer*) –
  - **ContentType** (*string*) –

*storage* / Client / `get_bucket_lifecycle_configuration`

## get\_bucket\_lifecycle\_configuration

`storage.Client.get_bucket_lifecycle_configuration(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.get_bucket_lifecycle_configuration(  
    Bucket='string'  
)
```

**Parameters** `Bucket` (*string*) – [REQUIRED]



**Return type** dict

**Returns**

### Response Syntax

```
{
  'Rule': {
    'Expiration': {
      'Date': datetime(2015, 1, 1),
      'Days': 123
    },
    'Filter': {
      'And': {
        'Tag': {
          'Key': 'string',
          'Value': 'string'
        }
      }
    },
    'ID': 'string',
    'NoncurrentVersionExpiration': {
      'NoncurrentDays': 123
    },
    'Prefix': 'string',
    'Status': 'string'
  }
}
```

### Response Structure

- (dict) –
  - **Rule** (dict) –
    - \* **Expiration** (dict) –
      - **Date** (datetime) –
      - **Days** (integer) –
    - \* **Filter** (dict) –
      - **And** (dict) –
      - **Tag** (dict) –
        - **Key** (string) –
        - **Value** (string) –
    - \* **ID** (string) –
    - \* **NoncurrentVersionExpiration** (dict) –
      - **NoncurrentDays** (integer) –
    - \* **Prefix** (string) –
    - \* **Status** (string) –

*storage* / Client / `get_bucket_object_versions`

### `get_bucket_object_versions`

`storage.Client.get_bucket_object_versions(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.get_bucket_object_versions(
    Bucket='string',
```

(continues on next page)

(continued from previous page)

```

Delimiter='string',
EncodingType='string',
KeyMarker='string',
MaxKeys='string',
Prefix='string',
VersionIdMarker='string'
)

```

**Parameters**

- **Bucket** (*string*) – **[REQUIRED]**
- **Delimiter** (*string*) –
- **EncodingType** (*string*) –
- **KeyMarker** (*string*) –
- **MaxKeys** (*string*) –
- **Prefix** (*string*) –
- **VersionIdMarker** (*string*) –

**Return type** dict**Returns****Response Syntax**

```

{
  'ContentType': 'string',
  'IsTruncated': True|False,
  'KeyMarker': 'string',
  'MaxKeys': 'string',
  'Name': 'string',
  'Prefix': 'string',
  'Version': [
    {
      'DisplayName': 'string',
      'ETag': 'string',
      'ID': 'string',
      'IsLatest': True|False,
      'Key': 'string',
      'LastModified': datetime(2015, 1, 1),
      'Owner': {
        'DisplayName': 'string',
        'ID': 'string'
      },
      'Size': 'string',
      'StorageClass': 'string',
      'VersionId': 'string'
    },
    ...
  ],
  'VersionIdMarker': 'string'
}

```

**Response Structure**

- (*dict*) –
  - **ContentType** (*string*) –
  - **IsTruncated** (*boolean*) –
  - **KeyMarker** (*string*) –
  - **MaxKeys** (*string*) –
  - **Name** (*string*) –
  - **Prefix** (*string*) –

- **Version** (*list*) –
  - \* (*dict*) –
    - **DisplayName** (*string*) –
    - **ETag** (*string*) –
    - **ID** (*string*) –
    - **IsLatest** (*boolean*) –
    - **Key** (*string*) –
    - **LastModified** (*datetime*) –
    - **Owner** (*dict*) –
    - **DisplayName** (*string*) –
    - **ID** (*string*) –
    - **Size** (*string*) –
    - **StorageClass** (*string*) –
    - **VersionId** (*string*) –
- **VersionIdMarker** (*string*) –

*storage* / Client / `get_bucket_policy`

## `get_bucket_policy`

`storage.Client.get_bucket_policy(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.get_bucket_policy(
    Bucket='string'
)
```

**Parameters** **Bucket** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{
    'ContentType': 'string',
    'Policy': 'string'
}
```

### Response Structure

- (*dict*) –
  - **ContentType** (*string*) –
  - **Policy** (*string*) –

*storage* / Client / `get_bucket_tagging`

## `get_bucket_tagging`

`storage.Client.get_bucket_tagging(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.get_bucket_tagging(  
    Bucket='string'  
)
```

**Parameters** **Bucket** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

#### Response Syntax

```
{  
    'ContentType': 'string',  
    'TagSet': [  
        {  
            'Key': 'string',  
            'Value': 'string'  
        },  
    ]  
}
```

#### Response Structure

- (*dict*) –
  - **ContentType** (*string*) –
  - **TagSet** (*list*) –
    - \* (*dict*) –
      - **Key** (*string*) –
      - **Value** (*string*) –

*storage* / Client / `get_bucket_version2`

### `get_bucket_version2`

`storage.Client.get_bucket_version2(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.get_bucket_version2(  
    Bucket='string',  
    ContinuationToken='string',  
    Delimiter='string',  
    EncodingType='string',  
    FetchOwner='string',  
    ListType='string',  
    Marker='string',  
    MaxKeys='string',  
    Prefix='string',  
    StartAfter='string'  
)
```

#### Parameters

- **Bucket** (*string*) – [REQUIRED]
- **ContinuationToken** (*string*) –
- **Delimiter** (*string*) –
- **EncodingType** (*string*) –
- **FetchOwner** (*string*) –
- **ListType** (*string*) – [REQUIRED]

- **Marker** (*string*) –
- **MaxKeys** (*string*) –
- **Prefix** (*string*) –
- **StartAfter** (*string*) –

Return type dict

Returns

### Response Syntax

```
{
  'CommonPrefixes': 'string',
  'ContentType': 'string',
  'Contents': [
    {
      'DisplayName': 'string',
      'ETag': 'string',
      'ID': 'string',
      'Key': 'string',
      'LastModified': datetime(2015, 1, 1),
      'Owner': {
        'DisplayName': 'string',
        'ID': 'string'
      },
      'Size': 'string',
      'StorageClass': 'string'
    },
    ...
  ],
  'ContinuationToken': 'string',
  'Delimiter': 'string',
  'EncodingType': 'string',
  'IsTruncated': True|False,
  'KeyCount': 'string',
  'MaxKeys': 'string',
  'Name': 'string',
  'NextContinuationToken': 'string',
  'Prefix': 'string',
  'StartAfter': 'string'
}
```

### Response Structure

- (*dict*) –
  - **CommonPrefixes** (*string*) –
  - **ContentType** (*string*) –
  - **Contents** (*list*) –
    - \* (*dict*) –
      - **DisplayName** (*string*) –
      - **ETag** (*string*) –
      - **ID** (*string*) –
      - **Key** (*string*) –
      - **LastModified** (*datetime*) –
      - **Owner** (*dict*) –
        - **DisplayName** (*string*) –
        - **ID** (*string*) –
      - **Size** (*string*) –
      - **StorageClass** (*string*) –
  - **ContinuationToken** (*string*) –
  - **Delimiter** (*string*) –

- **EncodingType** (*string*) –
- **IsTruncated** (*boolean*) –
- **KeyCount** (*string*) –
- **MaxKeys** (*string*) –
- **Name** (*string*) –
- **NextContinuationToken** (*string*) –
- **Prefix** (*string*) –
- **StartAfter** (*string*) –

*storage* / Client / `get_bucket_versioning`

## `get_bucket_versioning`

`storage.Client.get_bucket_versioning` (*\*\*kwargs*)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.get_bucket_versioning(  
    Bucket='string'  
)
```

**Parameters** **Bucket** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'ContentType': 'string',  
    'Status': 'string'  
}
```

### Response Structure

- (*dict*) –
  - **ContentType** (*string*) –
  - **Status** (*string*) –

*storage* / Client / `get_object`

## `get_object`

`storage.Client.get_object` (*\*\*kwargs*)

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.get_object(  
    Bucket='string',  
    Object='string',  
    PartNumber='string',  
    Range='string',  
    ResponseContentDisposition='string',  
    ResponseContentEncoding='string',  
    ResponseContentType='string',  
    VersionId='string',
```

(continues on next page)

(continued from previous page)

```

XAmzServerSideEncryptionCustomerAlgorithm='AES256',
XAmzServerSideEncryptionCustomerKey='string',
XAmzServerSideEncryptionCustomerKeyMd5='string'
)

```

**Parameters**

- **Bucket** (*string*) – [REQUIRED]
- **Object** (*string*) – [REQUIRED]
- **PartNumber** (*string*) –
- **Range** (*string*) –
- **ResponseContentDisposition** (*string*) –
- **ResponseContentEncoding** (*string*) –
- **ResponseContentType** (*string*) –
- **VersionId** (*string*) –
- **XAmzServerSideEncryptionCustomerAlgorithm** (*string*) –
- **XAmzServerSideEncryptionCustomerKey** (*string*) –
- **XAmzServerSideEncryptionCustomerKeyMd5** (*string*) –

**Return type** dict**Returns****Response Syntax**

```

{
    'AcceptRanges': 'string',
    'Body': StreamingBody(),
    'ContentRange': 'string',
    'ContentType': 'string',
    'ETag': 'string',
    'LastModified': 'string',
    'XAmzExpiration': 'string',
    'XAmzMpPartsCount': 'string',
    'XAmzServerSideEncryption': 'string'
}

```

**Response Structure**

- (*dict*) –
  - **AcceptRanges** (*string*) –
  - **Body** (*StreamingBody*) –
  - **ContentRange** (*string*) –
  - **ContentType** (*string*) –
  - **ETag** (*string*) –
  - **LastModified** (*string*) –
  - **XAmzExpiration** (*string*) –
  - **XAmzMpPartsCount** (*string*) –
  - **XAmzServerSideEncryption** (*string*) –

*storage* / *Client* / *get\_object\_acl***get\_object\_acl***storage*.*Client*.**get\_object\_acl** (*\*\*kwargs*)See also: [NIFCLOUD API Documentation](#)**Request Syntax**

```
response = client.get_object_acl(  
    Bucket='string',  
    Object='string',  
    VersionId='string'  
)
```

**Parameters**

- **Bucket** (*string*) – [REQUIRED]
- **Object** (*string*) – [REQUIRED]
- **VersionId** (*string*) –

**Return type** dict**Returns****Response Syntax**

```
{  
    'AccessControlList': {  
        'Grant': {  
            'Grantee': {  
                'DisplayName': 'string',  
                'ID': 'string'  
            },  
            'Permission': 'string'  
        },  
    },  
    'ContentType': 'string',  
    'Owner': {  
        'DisplayName': 'string',  
        'ID': 'string'  
    },  
}
```

**Response Structure**

- (*dict*) –
  - **AccessControlList** (*dict*) –
    - \* **Grant** (*dict*) –
      - **Grantee** (*dict*) –
      - **DisplayName** (*string*) –
      - **ID** (*string*) –
      - **Permission** (*string*) –
  - **ContentType** (*string*) –
  - **Owner** (*dict*) –
    - \* **DisplayName** (*string*) –
    - \* **ID** (*string*) –

*storage* / Client / `get_object_tagging`

**get\_object\_tagging**

`storage.Client.get_object_tagging(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.get_object_tagging(  
    Bucket='string',
```

(continues on next page)



(continued from previous page)

```
Object='string',
VersionId='string'
)
```

**Parameters**

- **Bucket** (*string*) – [REQUIRED]
- **Object** (*string*) – [REQUIRED]
- **VersionId** (*string*) –

**Return type** dict**Returns****Response Syntax**

```
{
  'ContentType': 'string',
  'TagSet': {
    'Tag': {
      'Key': 'string',
      'Value': 'string'
    }
  },
  'XAmzVersionId': 'string'
}
```

**Response Structure**

- (*dict*) –
  - **ContentType** (*string*) –
  - **TagSet** (*dict*) –
    - \* **Tag** (*dict*) –
      - **Key** (*string*) –
      - **Value** (*string*) –
  - **XAmzVersionId** (*string*) –

*storage* / Client / `get_paginator`**get\_paginator**`storage.Client.get_paginator(operation_name)`

Create a paginator for an operation.

**Parameters** **operation\_name** (*string*) – The operation name. This is the same name as the method name on the client. For example, if the method name is `create_foo`, and you'd normally invoke the operation as `client.create_foo(**kwargs)`, if the `create_foo` operation can be paginated, you can use the call `client.get_paginator("create_foo").`

**Raises** **OperationNotPageableError** – Raised if the operation is not pageable. You can use the `client.can_paginate` method to check if an operation is pageable.

**Return type** L{botocore.paginate.Paginator}**Returns** A paginator object.*storage* / Client / `get_service`

## get\_service

`storage.Client.get_service()`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.get_service()
```

**Return type** dict

**Returns**

### Response Syntax

```
{
  'Buckets': [
    {
      'CreationDate': datetime(2015, 1, 1),
      'Name': 'string'
    },
  ],
  'Owner': {
    'DisplayName': 'string',
    'ID': 'string'
  }
}
```

### Response Structure

- (dict) –
  - **Buckets** (list) –
    - \* (dict) –
      - **CreationDate** (datetime) –
      - **Name** (string) –
  - **Owner** (dict) –
    - \* **DisplayName** (string) –
    - \* **ID** (string) –

*storage* / Client / get\_waiter

## get\_waiter

`storage.Client.get_waiter(waiter_name)`

Returns an object that can wait for some condition.

**Parameters** **waiter\_name** (*str*) – The name of the waiter to get. See the waiters section of the service docs for a list of available waiters.

**Returns** The specified waiter object.

**Return type** `botocore.waiter.Waiter`

*storage* / Client / head\_bucket

## head\_bucket

`storage.Client.head_bucket(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.head_bucket(
    Bucket='string'
)
```

**Parameters** **Bucket** (*string*) – [REQUIRED]

**Returns** None

*storage* / Client / head\_object

## head\_object

`storage.Client.head_object(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.head_object(
    Bucket='string',
    ConsistencyControl='string',
    Object='string',
    PartNumber='string',
    VersionId='string',
    XAmzServerSideEncryptionCustomerAlgorithm='AES256',
    XAmzServerSideEncryptionCustomerKey='string',
    XAmzServerSideEncryptionCustomerKeyMd5='string'
)
```

### Parameters

- **Bucket** (*string*) – [REQUIRED]
- **ConsistencyControl** (*string*) –
- **Object** (*string*) – [REQUIRED]
- **PartNumber** (*string*) –
- **VersionId** (*string*) –
- **XAmzServerSideEncryptionCustomerAlgorithm** (*string*) –
- **XAmzServerSideEncryptionCustomerKey** (*string*) –
- **XAmzServerSideEncryptionCustomerKeyMd5** (*string*) –

**Return type** dict

**Returns**

### Response Syntax

```
{
    'AcceptRanges': 'string',
    'ContentType': 'string',
    'ETag': 'string',
    'LastModified': 'string',
    'XAmzExpiration': 'string',
    'XAmzMpPartsCount': 'string',
    'XAmzServerSideEncryption': 'string',
    'XAmzVersionId': 'string'
}
```

### Response Structure

- (*dict*) –
  - **AcceptRanges** (*string*) –
  - **ContentType** (*string*) –

- **ETag** (*string*) -
- **LastModified** (*string*) -
- **XAmzExpiration** (*string*) -
- **XAmzMpPartsCount** (*string*) -
- **XAmzServerSideEncryption** (*string*) -
- **XAmzVersionId** (*string*) -

*storage* / Client / `initiate_multipart_upload`

## `initiate_multipart_upload`

`storage.Client.initiate_multipart_upload(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.initiate_multipart_upload(
    Bucket='string',
    ContentDisposition='string',
    ContentEncoding='string',
    ContentType='string',
    Object='string',
    XAmzMeta='string',
    XAmzServerSideEncryption='string',
    XAmzServerSideEncryptionCustomerAlgorithm='AES256',
    XAmzServerSideEncryptionCustomerKey='string',
    XAmzServerSideEncryptionCustomerKeyMd5='string',
    XAmzStorageClass='STANDARD'|'REDUCED_REDUNDANCY',
    XAmzTagging='COPY'|'REPLACE'
)
```

### Parameters

- **Bucket** (*string*) - [REQUIRED]
- **ContentDisposition** (*string*) -
- **ContentEncoding** (*string*) -
- **ContentType** (*string*) -
- **Object** (*string*) - [REQUIRED]
- **XAmzMeta** (*string*) -
- **XAmzServerSideEncryption** (*string*) -
- **XAmzServerSideEncryptionCustomerAlgorithm** (*string*) -
- **XAmzServerSideEncryptionCustomerKey** (*string*) -
- **XAmzServerSideEncryptionCustomerKeyMd5** (*string*) -
- **XAmzStorageClass** (*string*) -
- **XAmzTagging** (*string*) -

**Return type** dict

**Returns**

### Response Syntax

```
{
    'Bucket': 'string',
    'ContentType': 'string',
    'Key': 'string',
    'UploadId': 'string'
}
```

### Response Structure

- (dict) –
  - **Bucket** (string) –
  - **ContentType** (string) –
  - **Key** (string) –
  - **UploadId** (string) –

*storage* / Client / list\_multipart\_uploads

## list\_multipart\_uploads

`storage.Client.list_multipart_uploads(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.list_multipart_uploads(
    Bucket='string',
    EncodingType='string',
    KeyMarker='string',
    MaxUploads=123,
    Prefix='string',
    UploadIdMarker='string'
)
```

### Parameters

- **Bucket** (string) – [REQUIRED]
- **EncodingType** (string) –
- **KeyMarker** (string) –
- **MaxUploads** (integer) –
- **Prefix** (string) –
- **UploadIdMarker** (string) –

**Return type** dict

### Returns

### Response Syntax

```
{
  'Bucket': 'string',
  'ContentType': 'string',
  'IsTruncated': True|False,
  'KeyMarker': 'string',
  'MaxUploads': 123,
  'NextKeyMarker': 'string',
  'NextUploadIdMarker': 'string',
  'Upload': [
    {
      'DisplayName': 'string',
      'ID': 'string',
      'Initiated': datetime(2015, 1, 1),
      'Initiator': {
        'DisplayName': 'string',
        'ID': 'string'
      },
      'Key': 'string',
      'Owner': {
        'DisplayName': 'string',
        'ID': 'string'
      }
    }
  ]
}
```

(continues on next page)

(continued from previous page)

```
        },
        'StorageClass': 'string',
        'UploadId': 'string'
    },
    ],
    'UploadIdMarker': 'string'
}
```

### Response Structure

- (dict) –
  - **Bucket** (*string*) –
  - **ContentType** (*string*) –
  - **IsTruncated** (*boolean*) –
  - **KeyMarker** (*string*) –
  - **MaxUploads** (*integer*) –
  - **NextKeyMarker** (*string*) –
  - **NextUploadIdMarker** (*string*) –
  - **Upload** (*list*) –
    - \* (dict) –
      - **DisplayName** (*string*) –
      - **ID** (*string*) –
      - **Initiated** (*datetime*) –
      - **Initiator** (*dict*) –
      - **DisplayName** (*string*) –
      - **ID** (*string*) –
      - **Key** (*string*) –
      - **Owner** (*dict*) –
      - **DisplayName** (*string*) –
      - **ID** (*string*) –
      - **StorageClass** (*string*) –
      - **UploadId** (*string*) –
  - **UploadIdMarker** (*string*) –

*storage* / Client / list\_parts

## list\_parts

`storage.Client.list_parts(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.list_parts(
    Bucket='string',
    EncodingType='string',
    MaxParts='string',
    Object='string',
    PartNumberMarker='string',
    UploadId='string'
)
```

### Parameters

- **Bucket** (*string*) – [REQUIRED]
- **EncodingType** (*string*) –

- **MaxParts** (*string*) –
- **Object** (*string*) – [REQUIRED]
- **PartNumberMarker** (*string*) –
- **UploadId** (*string*) – [REQUIRED]

Return type dict

Returns

### Response Syntax

```
{
  'Bucket': 'string',
  'ContentType': 'string',
  'EncodingType': 'string',
  'Initiator': {
    'DisplayName': 'string',
    'ID': 'string'
  },
  'IsTruncated': True|False,
  'Key': 'string',
  'MaxParts': 123,
  'NextPartNumberMarker': 123,
  'Owner': {
    'DisplayName': 'string',
    'ID': 'string'
  },
  'Part': [
    {
      'ETag': 'string',
      'LastModified': datetime(2015, 1, 1),
      'PartNumber': 123,
      'Size': 123
    },
  ],
  'PartNumberMarker': 123,
  'StorageClass': 'string',
  'UploadId': 'string'
}
```

### Response Structure

- (*dict*) –
  - **Bucket** (*string*) –
  - **ContentType** (*string*) –
  - **EncodingType** (*string*) –
  - **Initiator** (*dict*) –
    - \* **DisplayName** (*string*) –
    - \* **ID** (*string*) –
  - **IsTruncated** (*boolean*) –
  - **Key** (*string*) –
  - **MaxParts** (*integer*) –
  - **NextPartNumberMarker** (*integer*) –
  - **Owner** (*dict*) –
    - \* **DisplayName** (*string*) –
    - \* **ID** (*string*) –
  - **Part** (*list*) –
    - \* (*dict*) –
      - **ETag** (*string*) –
      - **LastModified** (*datetime*) –

- **PartNumber** (*integer*) –
- **Size** (*integer*) –
- **PartNumberMarker** (*integer*) –
- **StorageClass** (*string*) –
- **UploadId** (*string*) –

*storage* / Client / put\_bucket

## put\_bucket

`storage.Client.put_bucket(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.put_bucket(  
    Bucket='string'  
)
```

**Parameters** **Bucket** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax

```
{  
    'Location': 'string'  
}
```

### Response Structure

- (*dict*) –
  - **Location** (*string*) –

*storage* / Client / put\_bucket\_consistency

## put\_bucket\_consistency

`storage.Client.put_bucket_consistency(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.put_bucket_consistency(  
    Bucket='string',  
    XNtapSgConsistency='read-after-new-write'|'available'  
)
```

### Parameters

- **Bucket** (*string*) – [REQUIRED]
- **XNtapSgConsistency** (*string*) – [REQUIRED]

**Return type** dict

**Returns**

### Response Syntax



```
{
    'ComtentType': 'string'
}
```

**Response Structure**

- (*dict*) –
  - **ComtentType** (*string*) –

*storage* / Client / put\_bucket\_cors

**put\_bucket\_cors**

`storage.Client.put_bucket_cors(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.put_bucket_cors(
    Bucket='string',
    CORSConfiguration={
        'ListOfRequestCORSRule': [
            {
                'AllowedHeader': 'string',
                'AllowedOrigin': 'string',
                'ExposeHeader': 'string',
                'ID': 'string',
                'ListOfRequestAllowedMethod': [
                    'string',
                ],
                'MaxAgeSeconds': 123
            },
        ],
    },
)
```

**Parameters**

- **Bucket** (*string*) – [REQUIRED]
- **CORSConfiguration** (*dict*) – [REQUIRED]
  - **ListOfRequestCORSRule** (*list*) – [REQUIRED]
    - \* (*dict*) –
      - **AllowedHeader** (*string*) –
      - **AllowedOrigin** (*string*) – [REQUIRED]
      - **ExposeHeader** (*string*) –
      - **ID** (*string*) –
      - **ListOfRequestAllowedMethod** (*list*) – [REQUIRED]
      - (*string*) –
      - **MaxAgeSeconds** (*integer*) –

**Returns** None

*storage* / Client / put\_bucket\_lifecycle\_configuration

**put\_bucket\_lifecycle\_configuration**

`storage.Client.put_bucket_lifecycle_configuration(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

## Request Syntax

```
response = client.put_bucket_lifecycle_configuration(
    Bucket='string',
    LifecycleConfiguration={
        'ListOfRequestRule': [
            {
                'ID': 'string',
                'Prefix': 'string',
                'RequestExpiration': {
                    'Date': datetime(2015, 1, 1),
                    'Days': 123
                },
                'RequestFilter': {
                    'RequestAnd': {
                        'RequestTag': {
                            'Key': 'string',
                            'Value': 'string'
                        }
                    }
                },
                'RequestNoncurrentVersionExpiration': {
                    'NoncurrentDays': 123
                },
                'Status': 'Enabled'|'Disabled'
            },
        ]
    }
)
```

### Parameters

- **Bucket** (*string*) – [REQUIRED]
- **LifecycleConfiguration** (*dict*) – [REQUIRED]
  - **ListOfRequestRule** (*list*) – [REQUIRED]
    - \* (*dict*) –
      - **ID** (*string*) – [REQUIRED]
      - **Prefix** (*string*) –
      - **RequestExpiration** (*dict*) –
        - **Date** (*datetime*) –
        - **Days** (*integer*) –
      - **RequestFilter** (*dict*) –
        - **RequestAnd** (*dict*) –
          - **RequestTag** (*dict*) –
            - **Key** (*string*) –
            - **Value** (*string*) –
        - **RequestNoncurrentVersionExpiration** (*dict*) –
          - **NoncurrentDays** (*integer*) –
        - **Status** (*string*) –

**Returns** None

*storage* / Client / put\_bucket\_policy

## put\_bucket\_policy

`storage.Client.put_bucket_policy(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.put_bucket_policy(
    Bucket='string',
    ContentMd5='string',
    Policy='string'
)
```

#### Parameters

- **Bucket** (*string*) – [REQUIRED]
- **ContentMd5** (*string*) –
- **Policy** (*string*) –

**Returns** None

[storage](#) / Client / put\_bucket\_tagging

### put\_bucket\_tagging

`storage.Client.put_bucket_tagging(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.put_bucket_tagging(
    Bucket='string',
    ContentMd5='string',
    Tagging={
        'ListOfRequestTagSet': [
            {
                'RequestTag': {
                    'Key': 'string',
                    'Value': 'string'
                }
            },
        ]
    }
)
```

#### Parameters

- **Bucket** (*string*) – [REQUIRED]
- **ContentMd5** (*string*) –
- **Tagging** (*dict*) –
  - **ListOfRequestTagSet** (*list*) –
    - \* (*dict*) –
      - **RequestTag** (*dict*) –
        - **Key** (*string*) –
        - **Value** (*string*) –

**Returns** None

[storage](#) / Client / put\_bucket\_versioning

### put\_bucket\_versioning

`storage.Client.put_bucket_versioning(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.put_bucket_versioning(
    Bucket='string',
    ContentMd5='string',
    VersioningConfiguration={
        'Status': 'Suspended' | 'Enabled'
    }
)
```

**Parameters**

- **Bucket** (*string*) – [REQUIRED]
- **ContentMd5** (*string*) –
- **VersioningConfiguration** (*dict*) –
  - **Status** (*string*) –

**Returns** None

*storage* / Client / put\_object

**put\_object**

`storage.Client.put_object` (\*\*kwargs)

See also: [NIFCLOUD API Documentation](#)

**Request Syntax**

```
response = client.put_object(
    Body=b'bytes' | file,
    Bucket='string',
    ContentDisposition='string',
    ContentEncoding='aws-chunked',
    ContentMd5='string',
    ContentType='string',
    Object='string',
    XAmzMeta='string',
    XAmzServerSideEncryption='AES256',
    XAmzServerSideEncryptionCustomerAlgorithm='AES256',
    XAmzServerSideEncryptionCustomerKey='string',
    XAmzServerSideEncryptionCustomerKeyMd5='string',
    XAmzStorageClass='STANDARD' | 'REDUCED_REDUNDANCY',
    XAmzTagging='string'
)
```

**Parameters**

- **Body** (*bytes or seekable file-like object*) –
- **Bucket** (*string*) – [REQUIRED]
- **ContentDisposition** (*string*) –
- **ContentEncoding** (*string*) –
- **ContentMd5** (*string*) –
- **ContentType** (*string*) –
- **Object** (*string*) – [REQUIRED]
- **XAmzMeta** (*string*) –
- **XAmzServerSideEncryption** (*string*) –
- **XAmzServerSideEncryptionCustomerAlgorithm** (*string*) –
- **XAmzServerSideEncryptionCustomerKey** (*string*) –
- **XAmzServerSideEncryptionCustomerKeyMd5** (*string*) –
- **XAmzStorageClass** (*string*) –
- **XAmzTagging** (*string*) –

**Return type** dict

## Returns

### Response Syntax

```
{
    'ETag': 'string',
    'XAmzVersionId': 'string'
}
```

### Response Structure

- (*dict*) –
  - **ETag** (*string*) –
  - **XAmzVersionId** (*string*) –

*storage* / Client / put\_object\_copy

## put\_object\_copy

`storage.Client.put_object_copy(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.put_object_copy(
    Bucket='string',
    Object='string',
    XAmzCopySource='string',
    XAmzCopySourceIfMatch='string',
    XAmzCopySourceIfModifiedSince='string',
    XAmzCopySourceIfNoneMatch='string',
    XAmzCopySourceIfUnmodifiedSince='string',
    XAmzCopySourceServerSideEncryptionCustomerAlgorithm='AES256',
    XAmzCopySourceServerSideEncryptionCustomerKey='string',
    XAmzCopySourceServerSideEncryptionCustomerKeyMd5='string',
    XAmzMetadataDirective='COPY'|'REPLACE',
    XAmzServerSideEncryption='string',
    XAmzServerSideEncryptionCustomerAlgorithm='AES256',
    XAmzServerSideEncryptionCustomerKey='string',
    XAmzServerSideEncryptionCustomerKeyMd5='string',
    XAmzStorageClass='STANDARD'|'REDUCED_REDUNDANCY',
    XAmzTagging='string',
    XAmzTaggingDirective='COPY'|'REPLACE'
)
```

### Parameters

- **Bucket** (*string*) – [REQUIRED]
- **Object** (*string*) – [REQUIRED]
- **XAmzCopySource** (*string*) – [REQUIRED]
- **XAmzCopySourceIfMatch** (*string*) –
- **XAmzCopySourceIfModifiedSince** (*string*) –
- **XAmzCopySourceIfNoneMatch** (*string*) –
- **XAmzCopySourceIfUnmodifiedSince** (*string*) –
- **XAmzCopySourceServerSideEncryptionCustomerAlgorithm** (*string*) –
- **XAmzCopySourceServerSideEncryptionCustomerKey** (*string*) –
- **XAmzCopySourceServerSideEncryptionCustomerKeyMd5** (*string*) –

- **XAmzMetadataDirective** (*string*) –
- **XAmzServerSideEncryption** (*string*) –
- **XAmzServerSideEncryptionCustomerAlgorithm** (*string*) –
- **XAmzServerSideEncryptionCustomerKey** (*string*) –
- **XAmzServerSideEncryptionCustomerKeyMd5** (*string*) –
- **XAmzStorageClass** (*string*) –
- **XAmzTagging** (*string*) –
- **XAmzTaggingDirective** (*string*) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'ContentType': 'string',
    'ETag': 'string',
    'LastModified': datetime(2015, 1, 1)
}
```

#### Response Structure

- (*dict*) –
  - **ContentType** (*string*) –
  - **ETag** (*string*) –
  - **LastModified** (*datetime*) –

*storage* / Client / put\_object\_tagging

### put\_object\_tagging

`storage.Client.put_object_tagging(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.put_object_tagging(
    Bucket='string',
    ContentMd5='string',
    Object='string',
    Tagging={
        'RequestTagSet': {
            'RequestTag': {
                'Key': 'string',
                'Value': 'string'
            }
        }
    },
    VersionId='string'
)
```

#### Parameters

- **Bucket** (*string*) – [REQUIRED]
- **ContentMd5** (*string*) –
- **Object** (*string*) – [REQUIRED]
- **Tagging** (*dict*) –
  - **RequestTagSet** (*dict*) –
    - \* **RequestTag** (*dict*) –
      - **Key** (*string*) –

- **Value** (*string*) –
- **VersionId** (*string*) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'XAmzVersionId': 'string'
}
```

#### Response Structure

- (*dict*) –
  - **XAmzVersionId** (*string*) –

*storage* / Client / upload\_part

### upload\_part

`storage.Client.upload_part` (*\*\*kwargs*)

See also: [NIFCLOUD API Documentation](#)

#### Request Syntax

```
response = client.upload_part(
    Body=b'bytes'|file,
    Bucket='string',
    ContentMd5='string',
    Object='string',
    PartNumber='string',
    UploadId='string',
    XAmzServerSideEncryptionCustomerAlgorithm='AES256',
    XAmzServerSideEncryptionCustomerKey='string',
    XAmzServerSideEncryptionCustomerKeyMd5='string'
)
```

#### Parameters

- **Body** (*bytes or seekable file-like object*) –
- **Bucket** (*string*) – [REQUIRED]
- **ContentMd5** (*string*) –
- **Object** (*string*) – [REQUIRED]
- **PartNumber** (*string*) – [REQUIRED]
- **UploadId** (*string*) – [REQUIRED]
- **XAmzServerSideEncryptionCustomerAlgorithm** (*string*) –
- **XAmzServerSideEncryptionCustomerKey** (*string*) –
- **XAmzServerSideEncryptionCustomerKeyMd5** (*string*) –

**Return type** dict

**Returns**

#### Response Syntax

```
{
    'ETag': 'string'
}
```

#### Response Structure

- (*dict*) –

– ETag (*string*) –

*storage* / Client / upload\_part\_copy

## upload\_part\_copy

`storage.Client.upload_part_copy(**kwargs)`

See also: [NIFCLOUD API Documentation](#)

### Request Syntax

```
response = client.upload_part_copy(
    Bucket='string',
    Object='string',
    PartNumber='string',
    UploadId='string',
    XAmzCopySource='string',
    XAmzCopySourceIfMatch='string',
    XAmzCopySourceIfModifiedSince='string',
    XAmzCopySourceIfNoneMatch='string',
    XAmzCopySourceIfUnmodifiedSince='string',
    XAmzCopySourceRange='string',
    XAmzCopySourceServerSideEncryptionCustomerAlgorithm='AES256',
    XAmzCopySourceServerSideEncryptionCustomerKey='string',
    XAmzCopySourceServerSideEncryptionCustomerKeyMd5='string',
    XAmzServerSideEncryptionCustomerAlgorithm='AES256',
    XAmzServerSideEncryptionCustomerKey='string',
    XAmzServerSideEncryptionCustomerKeyMd5='string'
)
```

### Parameters

- **Bucket** (*string*) – [REQUIRED]
- **Object** (*string*) – [REQUIRED]
- **PartNumber** (*string*) – [REQUIRED]
- **UploadId** (*string*) – [REQUIRED]
- **XAmzCopySource** (*string*) – [REQUIRED]
- **XAmzCopySourceIfMatch** (*string*) –
- **XAmzCopySourceIfModifiedSince** (*string*) –
- **XAmzCopySourceIfNoneMatch** (*string*) –
- **XAmzCopySourceIfUnmodifiedSince** (*string*) –
- **XAmzCopySourceRange** (*string*) –
- **XAmzCopySourceServerSideEncryptionCustomerAlgorithm** (*string*) –
- **XAmzCopySourceServerSideEncryptionCustomerKey** (*string*) –
- **XAmzCopySourceServerSideEncryptionCustomerKeyMd5** (*string*) –
- **XAmzServerSideEncryptionCustomerAlgorithm** (*string*) –
- **XAmzServerSideEncryptionCustomerKey** (*string*) –
- **XAmzServerSideEncryptionCustomerKeyMd5** (*string*) –

Return type dict

Returns

### Response Syntax

```
{
    'ContentType': 'string',
    'ETag': 'string',
```

(continues on next page)



(continued from previous page)

```
'LastModified': datetime(2015, 1, 1),
'XAmzCopySourceVersionId': 'string',
'XAmzServerSideEncryption': 'string'
}
```

**Response Structure**

- *(dict)* –
  - **ContentType** (*string*) –
  - **ETag** (*string*) –
  - **LastModified** (*datetime*) –
  - **XAmzCopySourceVersionId** (*string*) –
  - **XAmzServerSideEncryption** (*string*) –

## 1.9.2 Client Exceptions

Client exceptions are available on a client instance via the `exceptions` property. For more detailed instructions and examples on the exact usage of client exceptions, see the error handling [user guide](#).

This client has no modeled exception classes.



## CHAPTER 2

---

### Indices and tables

---

- `genindex`
- `search`



## A

`abort_multipart_upload()` (*storage.Client method*), 563  
`add_source_identifier_to_subscription()` (*rdb.Client method*), 453  
`allocate_address()` (*computing.Client method*), 3  
`associate_address()` (*computing.Client method*), 4  
`associate_multi_ip_address_group()` (*computing.Client method*), 5  
`associate_route_table()` (*computing.Client method*), 6  
`associate_users()` (*computing.Client method*), 6  
`attach_disk()` (*hatoba.Client method*), 352  
`attach_iso_image()` (*computing.Client method*), 7  
`attach_network_interface()` (*computing.Client method*), 8  
`attach_volume()` (*computing.Client method*), 9  
`authorize_db_security_group_ingress()` (*rdb.Client method*), 454  
`authorize_firewall_group()` (*hatoba.Client method*), 353  
`authorize_nas_security_group_ingress()` (*nas.Client method*), 424  
`authorize_security_group_ingress()` (*computing.Client method*), 10

## C

`can_paginate()` (*computing.Client method*), 11  
`can_paginate()` (*dns.Client method*), 332  
`can_paginate()` (*ess.Client method*), 340  
`can_paginate()` (*hatoba.Client method*), 354  
`can_paginate()` (*nas.Client method*), 425  
`can_paginate()` (*rdb.Client method*), 455  
`can_paginate()` (*script.Client method*), 556  
`can_paginate()` (*serviceactivity.Client method*), 558  
`can_paginate()` (*storage.Client method*), 564  
`cancel_copy_instances()` (*computing.Client method*), 11

`cancel_upload()` (*computing.Client method*), 12  
`change_resource_record_sets()` (*dns.Client method*), 332  
`clear_load_balancer_session()` (*computing.Client method*), 12  
`clear_nas_session()` (*nas.Client method*), 426  
`close()` (*computing.Client method*), 13  
`close()` (*dns.Client method*), 334  
`close()` (*ess.Client method*), 341  
`close()` (*hatoba.Client method*), 355  
`close()` (*nas.Client method*), 427  
`close()` (*rdb.Client method*), 455  
`close()` (*script.Client method*), 556  
`close()` (*serviceactivity.Client method*), 558  
`close()` (*storage.Client method*), 564  
`complete_multipart_upload()` (*storage.Client method*), 564  
`computing.Client` (*built-in class*), 3  
`computing.Waiter.CustomerGatewayAvailable` (*built-in class*), 287  
`computing.Waiter.CustomerGatewayDeleted` (*built-in class*), 288  
`computing.Waiter.CustomerGatewayExists` (*built-in class*), 289  
`computing.Waiter.CustomerGatewayStopped` (*built-in class*), 290  
`computing.Waiter.CustomerGatewayWarning` (*built-in class*), 291  
`computing.Waiter.ElasticLoadBalancerAvailable` (*built-in class*), 292  
`computing.Waiter.ElasticLoadBalancerDeleted` (*built-in class*), 293  
`computing.Waiter.ElasticLoadBalancerExists` (*built-in class*), 295  
`computing.Waiter.InstanceDeleted` (*built-in class*), 296  
`computing.Waiter.InstanceExists` (*built-in class*), 297  
`computing.Waiter.InstanceImportError` (*built-in class*), 298

<code>computing.Waiter.InstanceRunning</code> ( <i>built-in class</i> ), 298	<code>computing.Waiter.VolumeExists</code> ( <i>built-in class</i> ), 322
<code>computing.Waiter.InstanceStopped</code> ( <i>built-in class</i> ), 299	<code>computing.Waiter.VolumeInUse</code> ( <i>built-in class</i> ), 322
<code>computing.Waiter.InstanceSuspending</code> ( <i>built-in class</i> ), 300	<code>computing.Waiter.VpnConnectionAvailable</code> ( <i>built-in class</i> ), 323
<code>computing.Waiter.InstanceWarning</code> ( <i>built-in class</i> ), 301	<code>computing.Waiter.VpnConnectionDeleted</code> ( <i>built-in class</i> ), 324
<code>computing.Waiter.LoadBalancerDeleted</code> ( <i>built-in class</i> ), 301	<code>computing.Waiter.VpnConnectionExists</code> ( <i>built-in class</i> ), 325
<code>computing.Waiter.LoadBalancerExists</code> ( <i>built-in class</i> ), 302	<code>computing.Waiter.VpnGatewayAvailable</code> ( <i>built-in class</i> ), 326
<code>computing.Waiter.LoadBalancerInService</code> ( <i>built-in class</i> ), 303	<code>computing.Waiter.VpnGatewayDeleted</code> ( <i>built-in class</i> ), 327
<code>computing.Waiter.PrivateLanAvailable</code> ( <i>built-in class</i> ), 304	<code>computing.Waiter.VpnGatewayExists</code> ( <i>built-in class</i> ), 328
<code>computing.Waiter.PrivateLanDeleted</code> ( <i>built-in class</i> ), 305	<code>computing.Waiter.VpnGatewayStopped</code> ( <i>built-in class</i> ), 329
<code>computing.Waiter.PrivateLanExists</code> ( <i>built-in class</i> ), 306	<code>computing.Waiter.VpnGatewayWarning</code> ( <i>built-in class</i> ), 330
<code>computing.Waiter.RemoteAccessVpnGatewayAvailable</code> ( <i>built-in class</i> ), 307	<code>create_instance_health_check()</code> ( <i>computing.Client method</i> ), 13
<code>computing.Waiter.RemoteAccessVpnGatewayDeleted</code> ( <i>built-in class</i> ), 308	<code>create_db_snapshot()</code> ( <i>rdb.Client method</i> ), 455
<code>computing.Waiter.RemoteAccessVpnGatewayExists</code> ( <i>built-in class</i> ), 309	<code>copy_from_backup_instance()</code> ( <i>computing.Client method</i> ), 15
<code>computing.Waiter.RouterAvailable</code> ( <i>built-in class</i> ), 309	<code>copy_instances()</code> ( <i>computing.Client method</i> ), 21
<code>computing.Waiter.RouterDeleted</code> ( <i>built-in class</i> ), 310	<code>create_backup_instances()</code> ( <i>computing.Client method</i> ), 23
<code>computing.Waiter.RouterExists</code> ( <i>built-in class</i> ), 311	<code>create_cluster()</code> ( <i>hatoba.Client method</i> ), 355
<code>computing.Waiter.RouterStopped</code> ( <i>built-in class</i> ), 312	<code>create_customer_gateway()</code> ( <i>computing.Client method</i> ), 24
<code>computing.Waiter.RouterWarning</code> ( <i>built-in class</i> ), 313	<code>create_db_instance()</code> ( <i>rdb.Client method</i> ), 457
<code>computing.Waiter.SecurityGroupApplied</code> ( <i>built-in class</i> ), 314	<code>create_db_instance_read_replica()</code> ( <i>rdb.Client method</i> ), 461
<code>computing.Waiter.SecurityGroupDeleted</code> ( <i>built-in class</i> ), 315	<code>create_db_parameter_group()</code> ( <i>rdb.Client method</i> ), 465
<code>computing.Waiter.SecurityGroupExists</code> ( <i>built-in class</i> ), 316	<code>create_db_security_group()</code> ( <i>rdb.Client method</i> ), 466
<code>computing.Waiter.SnapshotDeleted</code> ( <i>built-in class</i> ), 317	<code>create_db_snapshot()</code> ( <i>rdb.Client method</i> ), 467
<code>computing.Waiter.SnapshotExists</code> ( <i>built-in class</i> ), 318	<code>create_dhcp_options()</code> ( <i>computing.Client method</i> ), 25
<code>computing.Waiter.SnapshotNormal</code> ( <i>built-in class</i> ), 319	<code>create_disk()</code> ( <i>hatoba.Client method</i> ), 358
<code>computing.Waiter.VolumeAttached</code> ( <i>built-in class</i> ), 320	<code>create_event_subscription()</code> ( <i>rdb.Client method</i> ), 469
<code>computing.Waiter.VolumeAvailable</code> ( <i>built-in class</i> ), 320	<code>create_firewall_group()</code> ( <i>hatoba.Client method</i> ), 360
<code>computing.Waiter.VolumeDeleted</code> ( <i>built-in class</i> ), 321	<code>create_hosted_zone()</code> ( <i>dns.Client method</i> ), 334
	<code>create_image()</code> ( <i>computing.Client method</i> ), 27
	<code>create_instance_backup_rule()</code> ( <i>computing.Client method</i> ), 27
	<code>create_key_pair()</code> ( <i>computing.Client method</i> ), 29
	<code>create_load_balancer()</code> ( <i>computing.Client method</i> ), 29
	<code>create_multi_ip_address_group()</code> ( <i>computing.Client method</i> ), 29

- ing.Client method*), 31
- `create_nas_instance()` (*nas.Client method*), 427
- `create_nas_security_group()` (*nas.Client method*), 429
- `create_network_interface()` (*computing.Client method*), 32
- `create_node_pool()` (*hatoba.Client method*), 361
- `create_remote_access_vpn_gateway()` (*computing.Client method*), 35
- `create_remote_access_vpn_gateway_users()` (*computing.Client method*), 38
- `create_route()` (*computing.Client method*), 39
- `create_route_table()` (*computing.Client method*), 40
- `create_security_group()` (*computing.Client method*), 41
- `create_snapshot()` (*hatoba.Client method*), 363
- `create_ssl_certificate()` (*computing.Client method*), 42
- `create_tags()` (*hatoba.Client method*), 365
- `create_volume()` (*computing.Client method*), 43
- `create_vpn_connection()` (*computing.Client method*), 44
- `create_vpn_gateway()` (*computing.Client method*), 47
- ## D
- `delete_bucket()` (*storage.Client method*), 565
- `delete_bucket_cors()` (*storage.Client method*), 565
- `delete_bucket_lifecycle()` (*storage.Client method*), 566
- `delete_bucket_policy()` (*storage.Client method*), 566
- `delete_bucket_tagging()` (*storage.Client method*), 566
- `delete_cluster()` (*hatoba.Client method*), 366
- `delete_clusters()` (*hatoba.Client method*), 368
- `delete_customer_gateway()` (*computing.Client method*), 49
- `delete_db_instance()` (*rdb.Client method*), 470
- `delete_db_parameter_group()` (*rdb.Client method*), 474
- `delete_db_security_group()` (*rdb.Client method*), 474
- `delete_db_snapshot()` (*rdb.Client method*), 475
- `delete_dhcp_options()` (*computing.Client method*), 50
- `delete_disk()` (*hatoba.Client method*), 370
- `delete_disks()` (*hatoba.Client method*), 372
- `delete_event_subscription()` (*rdb.Client method*), 476
- `delete_firewall_group()` (*hatoba.Client method*), 373
- `delete_firewall_groups()` (*hatoba.Client method*), 374
- `delete_hosted_zone()` (*dns.Client method*), 335
- `delete_identity()` (*ess.Client method*), 341
- `delete_image()` (*computing.Client method*), 50
- `delete_instance_backup_rule()` (*computing.Client method*), 51
- `delete_iso_image()` (*computing.Client method*), 51
- `delete_key_pair()` (*computing.Client method*), 52
- `delete_load_balancer()` (*computing.Client method*), 52
- `delete_multi_ip_address_group()` (*computing.Client method*), 53
- `delete_multiple_objects()` (*storage.Client method*), 567
- `delete_nas_instance()` (*nas.Client method*), 430
- `delete_nas_security_group()` (*nas.Client method*), 432
- `delete_network_interface()` (*computing.Client method*), 53
- `delete_node_pool()` (*hatoba.Client method*), 375
- `delete_node_pools()` (*hatoba.Client method*), 377
- `delete_object()` (*storage.Client method*), 568
- `delete_object_tagging()` (*storage.Client method*), 568
- `delete_remote_access_vpn_gateway()` (*computing.Client method*), 54
- `delete_remote_access_vpn_gateway_connections()` (*computing.Client method*), 54
- `delete_remote_access_vpn_gateway_users()` (*computing.Client method*), 55
- `delete_route()` (*computing.Client method*), 56
- `delete_route_table()` (*computing.Client method*), 56
- `delete_security_group()` (*computing.Client method*), 57
- `delete_snapshot()` (*hatoba.Client method*), 378
- `delete_snapshots()` (*hatoba.Client method*), 379
- `delete_ssl_certificate()` (*computing.Client method*), 57
- `delete_tags()` (*hatoba.Client method*), 381
- `delete_volume()` (*computing.Client method*), 58
- `delete_vpn_connection()` (*computing.Client method*), 58
- `delete_vpn_gateway()` (*computing.Client method*), 59
- `deregister_instances_from_load_balancer()` (*computing.Client method*), 59
- `deregister_instances_from_security_group()` (*computing.Client method*), 60
- `describe_addresses()` (*computing.Client method*), 61
- `describe_associated_users()` (*comput-*

*ing.Client method*), 62  
describe\_availability\_zones() (*computing.Client method*), 63  
describe\_certificates() (*rdb.Client method*), 477  
describe\_customer\_gateways() (*computing.Client method*), 64  
describe\_db\_engine\_versions() (*rdb.Client method*), 478  
describe\_db\_instances() (*rdb.Client method*), 479  
describe\_db\_log\_files() (*rdb.Client method*), 483  
describe\_db\_parameter\_groups() (*rdb.Client method*), 484  
describe\_db\_parameters() (*rdb.Client method*), 485  
describe\_db\_security\_groups() (*rdb.Client method*), 486  
describe\_db\_snapshots() (*rdb.Client method*), 488  
describe\_dhcp\_options() (*computing.Client method*), 65  
describe\_engine\_default\_parameters() (*rdb.Client method*), 489  
describe\_event\_attributes() (*serviceactivity.Client method*), 558  
describe\_event\_calendar() (*serviceactivity.Client method*), 560  
describe\_event\_categories() (*rdb.Client method*), 491  
describe\_event\_subscriptions() (*rdb.Client method*), 491  
describe\_events() (*rdb.Client method*), 493  
describe\_images() (*computing.Client method*), 67  
describe\_instance\_attribute() (*computing.Client method*), 69  
describe\_instance\_backup\_rule\_activities() (*computing.Client method*), 73  
describe\_instance\_backup\_rules() (*computing.Client method*), 74  
describe\_instance\_health() (*computing.Client method*), 75  
describe\_instances() (*computing.Client method*), 76  
describe\_iso\_images() (*computing.Client method*), 83  
describe\_key\_pairs() (*computing.Client method*), 84  
describe\_load\_balancers() (*computing.Client method*), 85  
describe\_multi\_ip\_address\_groups() (*computing.Client method*), 88  
describe\_nas\_instances() (*nas.Client method*), 432  
describe\_nas\_security\_groups() (*nas.Client method*), 434  
describe\_network\_interfaces() (*computing.Client method*), 90  
describe\_orderable\_db\_instance\_options() (*rdb.Client method*), 494  
describe\_regions() (*computing.Client method*), 93  
describe\_remote\_access\_vpn\_gateway\_activities() (*computing.Client method*), 94  
describe\_remote\_access\_vpn\_gateway\_client\_config() (*computing.Client method*), 94  
describe\_remote\_access\_vpn\_gateway\_connections() (*computing.Client method*), 95  
describe\_remote\_access\_vpn\_gateways() (*computing.Client method*), 96  
describe\_resources() (*computing.Client method*), 99  
describe\_route\_tables() (*computing.Client method*), 102  
describe\_security\_activities() (*computing.Client method*), 104  
describe\_security\_groups() (*computing.Client method*), 105  
describe\_service\_status() (*computing.Client method*), 107  
describe\_service\_statuses() (*serviceactivity.Client method*), 561  
describe\_ssl\_certificate\_attribute() (*computing.Client method*), 108  
describe\_ssl\_certificates() (*computing.Client method*), 110  
describe\_uploads() (*computing.Client method*), 111  
describe\_usage() (*computing.Client method*), 112  
describe\_user\_activities() (*computing.Client method*), 128  
describe\_volumes() (*computing.Client method*), 129  
describe\_vpn\_connections() (*computing.Client method*), 131  
describe\_vpn\_gateways() (*computing.Client method*), 133  
detach\_disk() (*hatoba.Client method*), 381  
detach\_iso\_image() (*computing.Client method*), 135  
detach\_network\_interface() (*computing.Client method*), 136  
detach\_volume() (*computing.Client method*), 137  
disassociate\_address() (*computing.Client method*), 137  
disassociate\_multi\_ip\_address\_group() (*computing.Client method*), 138



- disassociate\_route\_table() (*computing.Client method*), 139
- dissociate\_users() (*computing.Client method*), 139
- dns.Client (*built-in class*), 331
- download\_db\_log\_file\_portion() (*rdb.Client method*), 496
- download\_ssl\_certificate() (*computing.Client method*), 140
- ## E
- ess.Client (*built-in class*), 340
- execute\_script() (*script.Client method*), 556
- extend\_volume\_size() (*computing.Client method*), 141
- ## G
- get\_bucket() (*storage.Client method*), 569
- get\_bucket\_acl() (*storage.Client method*), 570
- get\_bucket\_consistency() (*storage.Client method*), 571
- get\_bucket\_cors() (*storage.Client method*), 572
- get\_bucket\_lifecycle\_configuration() (*storage.Client method*), 572
- get\_bucket\_object\_versions() (*storage.Client method*), 573
- get\_bucket\_policy() (*storage.Client method*), 575
- get\_bucket\_tagging() (*storage.Client method*), 575
- get\_bucket\_version2() (*storage.Client method*), 576
- get\_bucket\_versioning() (*storage.Client method*), 578
- get\_change() (*dns.Client method*), 335
- get\_cluster() (*hatoba.Client method*), 383
- get\_cluster\_credentials() (*hatoba.Client method*), 385
- get\_delivery\_log() (*ess.Client method*), 341
- get\_disk() (*hatoba.Client method*), 385
- get\_firewall\_group() (*hatoba.Client method*), 387
- get\_hosted\_zone() (*dns.Client method*), 336
- get\_identity\_dkim\_attributes() (*ess.Client method*), 342
- get\_identity\_verification\_attributes() (*ess.Client method*), 343
- get\_load\_balancer() (*hatoba.Client method*), 388
- get\_metric\_statistics() (*nas.Client method*), 435
- get\_node\_pool() (*hatoba.Client method*), 390
- get\_object() (*storage.Client method*), 578
- get\_object\_acl() (*storage.Client method*), 579
- get\_object\_tagging() (*storage.Client method*), 580
- get\_paginator() (*computing.Client method*), 141
- get\_paginator() (*dns.Client method*), 337
- get\_paginator() (*ess.Client method*), 344
- get\_paginator() (*hatoba.Client method*), 392
- get\_paginator() (*nas.Client method*), 436
- get\_paginator() (*rdb.Client method*), 497
- get\_paginator() (*script.Client method*), 557
- get\_paginator() (*serviceactivity.Client method*), 562
- get\_paginator() (*storage.Client method*), 581
- get\_send\_quota() (*ess.Client method*), 344
- get\_send\_statistics() (*ess.Client method*), 345
- get\_server\_config() (*hatoba.Client method*), 392
- get\_service() (*storage.Client method*), 582
- get\_snapshot() (*hatoba.Client method*), 393
- get\_waiter() (*computing.Client method*), 142
- get\_waiter() (*dns.Client method*), 337
- get\_waiter() (*ess.Client method*), 346
- get\_waiter() (*hatoba.Client method*), 394
- get\_waiter() (*nas.Client method*), 437
- get\_waiter() (*rdb.Client method*), 497
- get\_waiter() (*script.Client method*), 557
- get\_waiter() (*serviceactivity.Client method*), 563
- get\_waiter() (*storage.Client method*), 582
- ## H
- hatoba.Client (*built-in class*), 351
- hatoba.Waiter.ClusterDeleted (*built-in class*), 421
- hatoba.Waiter.ClusterRunning (*built-in class*), 422
- hatoba.Waiter.FirewallRuleAuthorized (*built-in class*), 423
- hatoba.Waiter.SnapshotAvailable (*built-in class*), 423
- head\_bucket() (*storage.Client method*), 582
- head\_object() (*storage.Client method*), 583
- ## I
- import\_instance() (*computing.Client method*), 142
- import\_key\_pair() (*computing.Client method*), 146
- increase\_multi\_ip\_address\_count() (*computing.Client method*), 146
- initiate\_multipart\_upload() (*storage.Client method*), 584
- ## L
- list\_clusters() (*hatoba.Client method*), 394
- list\_disks() (*hatoba.Client method*), 396

`list_firewall_groups()` (*hatoba.Client method*), 398  
`list_hosted_zones()` (*dns.Client method*), 337  
`list_identities()` (*ess.Client method*), 346  
`list_load_balancers()` (*hatoba.Client method*), 399  
`list_multipart_uploads()` (*storage.Client method*), 585  
`list_node_pools()` (*hatoba.Client method*), 401  
`list_parts()` (*storage.Client method*), 586  
`list_resource_record_sets()` (*dns.Client method*), 338  
`list_snapshots()` (*hatoba.Client method*), 403  
`list_tags()` (*hatoba.Client method*), 404

## M

`modify_db_instance()` (*rdb.Client method*), 497  
`modify_db_instance_network()` (*rdb.Client method*), 502  
`modify_db_parameter_group()` (*rdb.Client method*), 505  
`modify_event_subscription()` (*rdb.Client method*), 506  
`modify_image_attribute()` (*computing.Client method*), 147  
`modify_instance_attribute()` (*computing.Client method*), 148  
`modify_instance_backup_rule_attribute()` (*computing.Client method*), 149  
`modify_multi_ip_address_group_attribute()` (*computing.Client method*), 149  
`modify_nas_instance()` (*nas.Client method*), 437  
`modify_nas_security_group()` (*nas.Client method*), 439  
`modify_network_interface_attribute()` (*computing.Client method*), 150  
`modify_remote_access_vpn_gateway_attribute()` (*computing.Client method*), 150  
`modify_remote_access_vpn_gateway_user_attribute()` (*computing.Client method*), 151  
`modify_ssl_certificate_attribute()` (*computing.Client method*), 152  
`modify_volume_attribute()` (*computing.Client method*), 152

## N

`nas.Client` (*built-in class*), 424  
`nas.Waiter.NASInstanceAvailable` (*built-in class*), 443  
`nas.Waiter.NASInstanceDeleted` (*built-in class*), 444  
`nas.Waiter.NASInstanceExists` (*built-in class*), 444

`nas.Waiter.NASInstanceFailed` (*built-in class*), 445  
`nas.Waiter.NASInstanceStorageFull` (*built-in class*), 445  
`nas.Waiter.NASSecurityGroupDeleted` (*built-in class*), 446  
`nas.Waiter.NASSecurityGroupExists` (*built-in class*), 447  
`nas.Waiter.NASSecurityGroupIPRangesAuthFailed` (*built-in class*), 447  
`nas.Waiter.NASSecurityGroupIPRangesAuthorized` (*built-in class*), 448  
`nas.Waiter.NASSecurityGroupIPRangesEmptied` (*built-in class*), 449  
`nas.Waiter.NASSecurityGroupIPRangesRevokeFailed` (*built-in class*), 449  
`nas.Waiter.NASSecurityGroupSecurityGroupsAuthFailed` (*built-in class*), 450  
`nas.Waiter.NASSecurityGroupSecurityGroupsAuthorized` (*built-in class*), 451  
`nas.Waiter.NASSecurityGroupSecurityGroupsEmptied` (*built-in class*), 451  
`nas.Waiter.NASSecurityGroupSecurityGroupsRevokeFailed` (*built-in class*), 452  
`nifty_associate_image()` (*computing.Client method*), 153  
`nifty_associate_nat_table()` (*computing.Client method*), 154  
`nifty_associate_route_table_with_elastic_load_balancer()` (*computing.Client method*), 154  
`nifty_associate_route_table_with_vpn_gateway()` (*computing.Client method*), 155  
`nifty_configure_elastic_load_balancer_health_check()` (*computing.Client method*), 156  
`nifty_create_alarm()` (*computing.Client method*), 157  
`nifty_create_auto_scaling_group()` (*computing.Client method*), 159  
`nifty_create_dhcp_config()` (*computing.Client method*), 161  
`nifty_create_dhcp_ip_address_pool()` (*computing.Client method*), 162  
`nifty_create_dhcp_static_mapping()` (*computing.Client method*), 162  
`nifty_create_elastic_load_balancer()` (*computing.Client method*), 163  
`nifty_create_instance_snapshot()` (*computing.Client method*), 165  
`nifty_create_nat_rule()` (*computing.Client method*), 166  
`nifty_create_nat_table()` (*computing.Client method*), 168  
`nifty_create_private_lan()` (*computing.Client method*), 169

`nifty_create_router()` (*computing.Client method*), 172  
`nifty_create_separate_instance_rule()` (*computing.Client method*), 174  
`nifty_create_web_proxy()` (*computing.Client method*), 175  
`nifty_delete_alarm()` (*computing.Client method*), 176  
`nifty_delete_auto_scaling_group()` (*computing.Client method*), 177  
`nifty_delete_dhcp_config()` (*computing.Client method*), 177  
`nifty_delete_dhcp_ip_address_pool()` (*computing.Client method*), 178  
`nifty_delete_dhcp_static_mapping()` (*computing.Client method*), 178  
`nifty_delete_elastic_load_balancer()` (*computing.Client method*), 179  
`nifty_delete_instance_snapshot()` (*computing.Client method*), 180  
`nifty_delete_nat_rule()` (*computing.Client method*), 181  
`nifty_delete_nat_table()` (*computing.Client method*), 181  
`nifty_delete_private_lan()` (*computing.Client method*), 182  
`nifty_delete_router()` (*computing.Client method*), 182  
`nifty_delete_separate_instance_rule()` (*computing.Client method*), 183  
`nifty_delete_web_proxy()` (*computing.Client method*), 184  
`nifty_deregister_instances_from_elastic_load_balancer()` (*computing.Client method*), 184  
`nifty_deregister_instances_from_separate_instance_rule()` (*computing.Client method*), 185  
`nifty_deregister_routers_from_security_group()` (*computing.Client method*), 186  
`nifty_deregister_vpn_gateways_from_security_group()` (*computing.Client method*), 187  
`nifty_describe_alarm_history()` (*computing.Client method*), 188  
`nifty_describe_alarm_rules_activities()` (*computing.Client method*), 189  
`nifty_describe_alarms()` (*computing.Client method*), 190  
`nifty_describe_alarms_partitions()` (*computing.Client method*), 192  
`nifty_describe_auto_scaling_groups()` (*computing.Client method*), 193  
`nifty_describe_corporate_info_for_certificate()` (*computing.Client method*), 196  
`nifty_describe_dhcp_configs()` (*computing.Client method*), 197  
`nifty_describe_dhcp_status()` (*computing.Client method*), 198  
`nifty_describe_elastic_load_balancers()` (*computing.Client method*), 200  
`nifty_describe_instance_elastic_load_balancer_health()` (*computing.Client method*), 204  
`nifty_describe_instance_snapshots()` (*computing.Client method*), 205  
`nifty_describe_load_balancer_ssl_policies()` (*computing.Client method*), 206  
`nifty_describe_nat_tables()` (*computing.Client method*), 207  
`nifty_describe_performance_chart()` (*computing.Client method*), 209  
`nifty_describe_private_lans()` (*computing.Client method*), 210  
`nifty_describe_routers()` (*computing.Client method*), 213  
`nifty_describe_scaling_activities()` (*computing.Client method*), 216  
`nifty_describe_separate_instance_rules()` (*computing.Client method*), 217  
`nifty_describe_vpn_gateway_activities()` (*computing.Client method*), 218  
`nifty_describe_web_proxies()` (*computing.Client method*), 219  
`nifty_disable_dhcp()` (*computing.Client method*), 221  
`nifty_disassociate_nat_table()` (*computing.Client method*), 221  
`nifty_disassociate_route_table_from_elastic_load_balancer()` (*computing.Client method*), 222  
`nifty_disassociate_route_table_from_vpn_gateway()` (*computing.Client method*), 222  
`nifty_enable_dhcp()` (*computing.Client method*), 223  
`nifty_get_metric_statistics()` (*rdb.Client method*), 507  
`nifty_modify_address_attribute()` (*computing.Client method*), 224  
`nifty_modify_customer_gateway_attribute()` (*computing.Client method*), 224  
`nifty_modify_elastic_load_balancer_attributes()` (*computing.Client method*), 225  
`nifty_modify_instance_snapshot_attribute()` (*computing.Client method*), 226  
`nifty_modify_key_pair_attribute()` (*computing.Client method*), 227  
`nifty_modify_private_lan_attribute()` (*computing.Client method*), 227  
`nifty_modify_router_attribute()` (*computing.Client method*), 228  
`nifty_modify_vpn_gateway_attribute()` (*computing.Client method*), 229

nifty\_modify\_web\_proxy\_attribute() (*computing.Client method*), 229  
 nifty\_reboot\_routers() (*computing.Client method*), 230  
 nifty\_reboot\_vpn\_gateways() (*computing.Client method*), 231  
 nifty\_register\_instances\_with\_elastic\_load\_balancer() (*computing.Client method*), 232  
 nifty\_register\_instances\_with\_separate\_instance\_profile() (*computing.Client method*), 232  
 nifty\_register\_port\_with\_elastic\_load\_balancer() (*computing.Client method*), 233  
 nifty\_register\_routers\_with\_security\_group() (*computing.Client method*), 235  
 nifty\_register\_vpn\_gateways\_with\_security\_group() (*computing.Client method*), 235  
 nifty\_release\_router\_backup\_state() (*computing.Client method*), 236  
 nifty\_release\_vpn\_gateway\_backup\_state() (*computing.Client method*), 237  
 nifty\_replace\_dhcp\_config() (*computing.Client method*), 237  
 nifty\_replace\_dhcp\_option() (*computing.Client method*), 238  
 nifty\_replace\_elastic\_load\_balancer\_latest\_version() (*computing.Client method*), 239  
 nifty\_replace\_elastic\_load\_balancer\_listener() (*computing.Client method*), 240  
 nifty\_replace\_nat\_rule() (*computing.Client method*), 241  
 nifty\_replace\_nat\_table\_association() (*computing.Client method*), 243  
 nifty\_replace\_route\_table\_association\_with\_elastic\_load\_balancer() (*computing.Client method*), 243  
 nifty\_replace\_route\_table\_association\_with\_vpn\_gateway() (*computing.Client method*), 244  
 nifty\_replace\_router\_latest\_version() (*computing.Client method*), 244  
 nifty\_replace\_vpn\_gateway\_latest\_version() (*computing.Client method*), 245  
 nifty\_restore\_instance\_snapshot() (*computing.Client method*), 246  
 nifty\_restore\_router\_previous\_version() (*computing.Client method*), 246  
 nifty\_restore\_vpn\_gateway\_previous\_version() (*computing.Client method*), 247  
 nifty\_retry\_import\_instance() (*computing.Client method*), 247  
 nifty\_set\_load\_balancer\_ssl\_policies\_of\_listener() (*computing.Client method*), 248  
 nifty\_unset\_load\_balancer\_ssl\_policies\_of\_listener() (*computing.Client method*), 249  
 nifty\_update\_alarm() (*computing.Client method*), 249  
 nifty\_update\_auto\_scaling\_group() (*computing.Client method*), 251  
 nifty\_update\_elastic\_load\_balancer() (*computing.Client method*), 253  
 nifty\_update\_instance\_network\_interfaces() (*computing.Client method*), 254  
 nifty\_update\_router\_network\_interfaces() (*computing.Client method*), 255  
 nifty\_update\_separate\_instance\_rule() (*computing.Client method*), 256  
 nifty\_update\_vpn\_gateway\_network\_interfaces() (*computing.Client method*), 257

## P

put\_bucket() (*storage.Client method*), 588  
 put\_bucket\_consistency() (*storage.Client method*), 588  
 put\_bucket\_cors() (*storage.Client method*), 589  
 put\_bucket\_lifecycle\_configuration() (*storage.Client method*), 589  
 put\_bucket\_policy() (*storage.Client method*), 590  
 put\_bucket\_tagging() (*storage.Client method*), 591  
 put\_bucket\_versioning() (*storage.Client method*), 591  
 put\_object() (*storage.Client method*), 592  
 put\_object\_copy() (*storage.Client method*), 593  
 put\_object\_tagging() (*storage.Client method*), 594

## R

rdb.Client (*built-in class*), 542  
 rdb.Waiter.DBInstanceAvailable (*built-in class*), 542  
 rdb.Waiter.DBInstanceDeleted (*built-in class*), 543  
 rdb.Waiter.DBInstanceExists (*built-in class*), 544  
 rdb.Waiter.DBInstanceFailed (*built-in class*), 545  
 rdb.Waiter.DBInstanceStorageFull (*built-in class*), 546  
 rdb.Waiter.DBSecurityGroupDeleted (*built-in class*), 546  
 rdb.Waiter.DBSecurityGroupEC2SecurityGroupsAuthFailed (*built-in class*), 547  
 rdb.Waiter.DBSecurityGroupEC2SecurityGroupsAuthorized (*built-in class*), 548  
 rdb.Waiter.DBSecurityGroupEC2SecurityGroupsEmpty (*built-in class*), 549  
 rdb.Waiter.DBSecurityGroupEC2SecurityGroupsRevokeFailed (*built-in class*), 550



`rdb.Waiter.DBSecurityGroupExists` (*built-in class*), 551  
`rdb.Waiter.DBSecurityGroupIPRangesAuthFailed` (*built-in class*), 552  
`rdb.Waiter.DBSecurityGroupIPRangesAuthorized` (*built-in class*), 553  
`rdb.Waiter.DBSecurityGroupIPRangesEmpty` (*built-in class*), 554  
`rdb.Waiter.DBSecurityGroupIPRangesRevokeFailed` (*built-in class*), 555  
`reboot_db_instance()` (*rdb.Client method*), 509  
`reboot_instances()` (*computing.Client method*), 257  
`reboot_node()` (*hatoba.Client method*), 405  
`reboot_remote_access_vpn_gateway()` (*computing.Client method*), 258  
`refresh_instance_backup_rule()` (*computing.Client method*), 259  
`register_corporate_info_for_certificate()` (*computing.Client method*), 259  
`register_instances_with_load_balancer()` (*computing.Client method*), 261  
`register_instances_with_security_group()` (*computing.Client method*), 262  
`register_port_with_load_balancer()` (*computing.Client method*), 263  
`release_address()` (*computing.Client method*), 264  
`release_multi_ip_addresses()` (*computing.Client method*), 265  
`remove_source_identifier_from_subscription()` (*rdb.Client method*), 512  
`replace_remote_access_vpn_gateway_latest_version()` (*computing.Client method*), 265  
`replace_route()` (*computing.Client method*), 266  
`replace_route_table_association()` (*computing.Client method*), 267  
`reset_db_parameter_group()` (*rdb.Client method*), 513  
`reset_external_master()` (*rdb.Client method*), 514  
`restore_cluster_from_snapshot()` (*hatoba.Client method*), 406  
`restore_db_instance_from_db_snapshot()` (*rdb.Client method*), 517  
`restore_db_instance_to_point_in_time()` (*rdb.Client method*), 522  
`revoke_db_security_group_ingress()` (*rdb.Client method*), 526  
`revoke_firewall_group()` (*hatoba.Client method*), 408  
`revoke_nas_security_group_ingress()` (*nas.Client method*), 440  
`revoke_security_group_ingress()` (*computing.Client method*), 267  
`run_instances()` (*computing.Client method*), 268

## S

`send_email()` (*ess.Client method*), 347  
`send_raw_email()` (*ess.Client method*), 348  
`serviceactivity.Client` (*built-in class*), 558  
`set_external_master()` (*rdb.Client method*), 528  
`set_filter_for_load_balancer()` (*computing.Client method*), 275  
`set_identity_dkim_enabled()` (*ess.Client method*), 349  
`set_load_balancer_listener_ssl_certificate()` (*computing.Client method*), 276  
`set_node_pool_size()` (*hatoba.Client method*), 410  
`set_remote_access_vpn_gateway_ca_certificate()` (*computing.Client method*), 276  
`set_remote_access_vpn_gateway_ssl_certificate()` (*computing.Client method*), 277  
`start_instances()` (*computing.Client method*), 277  
`start_replication()` (*rdb.Client method*), 531  
`stop_instances()` (*computing.Client method*), 279  
`stop_replication()` (*rdb.Client method*), 535  
`storage.Client` (*built-in class*), 563

## T

`terminate_instances()` (*computing.Client method*), 280

## U

`unset_load_balancer_listener_ssl_certificate()` (*computing.Client method*), 281  
`unset_remote_access_vpn_gateway_ca_certificate()` (*computing.Client method*), 282  
`unset_remote_access_vpn_gateway_ssl_certificate()` (*computing.Client method*), 282  
`update_cluster()` (*hatoba.Client method*), 411  
`update_disk()` (*hatoba.Client method*), 414  
`update_firewall_group()` (*hatoba.Client method*), 415  
`update_load_balancer()` (*computing.Client method*), 283  
`update_load_balancer_option()` (*computing.Client method*), 284  
`update_node_pool()` (*hatoba.Client method*), 417  
`update_security_group()` (*computing.Client method*), 284  
`update_snapshot()` (*hatoba.Client method*), 418  
`update_tags()` (*hatoba.Client method*), 420  
`upgrade_db_engine_version()` (*rdb.Client method*), 538

`upgrade_nas_instance()` (*nas.Client method*), [441](#)  
`upload_iso_image()` (*computing.Client method*), [285](#)  
`upload_part()` (*storage.Client method*), [595](#)  
`upload_part_copy()` (*storage.Client method*), [596](#)  
`upload_ssl_certificate()` (*computing.Client method*), [286](#)

## V

`verify_domain_dkim()` (*ess.Client method*), [349](#)  
`verify_domain_identity()` (*ess.Client method*), [350](#)  
`verify_email_identity()` (*ess.Client method*), [351](#)

## W

`wait()` (*computing.Waiter.CustomerGatewayAvailable method*), [287](#)  
`wait()` (*computing.Waiter.CustomerGatewayDeleted method*), [288](#)  
`wait()` (*computing.Waiter.CustomerGatewayExists method*), [289](#)  
`wait()` (*computing.Waiter.CustomerGatewayStopped method*), [290](#)  
`wait()` (*computing.Waiter.CustomerGatewayWarning method*), [291](#)  
`wait()` (*computing.Waiter.ElasticLoadBalancerAvailable method*), [292](#)  
`wait()` (*computing.Waiter.ElasticLoadBalancerDeleted method*), [293](#)  
`wait()` (*computing.Waiter.ElasticLoadBalancerExists method*), [295](#)  
`wait()` (*computing.Waiter.InstanceDeleted method*), [296](#)  
`wait()` (*computing.Waiter.InstanceExists method*), [297](#)  
`wait()` (*computing.Waiter.InstanceImportError method*), [298](#)  
`wait()` (*computing.Waiter.InstanceRunning method*), [298](#)  
`wait()` (*computing.Waiter.InstanceStopped method*), [299](#)  
`wait()` (*computing.Waiter.InstanceSuspending method*), [300](#)  
`wait()` (*computing.Waiter.InstanceWarning method*), [301](#)  
`wait()` (*computing.Waiter.LoadBalancerDeleted method*), [301](#)  
`wait()` (*computing.Waiter.LoadBalancerExists method*), [302](#)  
`wait()` (*computing.Waiter.LoadBalancerInService method*), [303](#)  
`wait()` (*computing.Waiter.PrivateLanAvailable method*), [304](#)

`wait()` (*computing.Waiter.PrivateLanDeleted method*), [305](#)  
`wait()` (*computing.Waiter.PrivateLanExists method*), [306](#)  
`wait()` (*computing.Waiter.RemoteAccessVpnGatewayAvailable method*), [307](#)  
`wait()` (*computing.Waiter.RemoteAccessVpnGatewayDeleted method*), [308](#)  
`wait()` (*computing.Waiter.RemoteAccessVpnGatewayExists method*), [309](#)  
`wait()` (*computing.Waiter.RouterAvailable method*), [309](#)  
`wait()` (*computing.Waiter.RouterDeleted method*), [310](#)  
`wait()` (*computing.Waiter.RouterExists method*), [311](#)  
`wait()` (*computing.Waiter.RouterStopped method*), [312](#)  
`wait()` (*computing.Waiter.RouterWarning method*), [313](#)  
`wait()` (*computing.Waiter.SecurityGroupApplied method*), [315](#)  
`wait()` (*computing.Waiter.SecurityGroupDeleted method*), [315](#)  
`wait()` (*computing.Waiter.SecurityGroupExists method*), [316](#)  
`wait()` (*computing.Waiter.SnapshotDeleted method*), [317](#)  
`wait()` (*computing.Waiter.SnapshotExists method*), [318](#)  
`wait()` (*computing.Waiter.SnapshotNormal method*), [319](#)  
`wait()` (*computing.Waiter.VolumeAttached method*), [320](#)  
`wait()` (*computing.Waiter.VolumeAvailable method*), [320](#)  
`wait()` (*computing.Waiter.VolumeDeleted method*), [321](#)  
`wait()` (*computing.Waiter.VolumeExists method*), [322](#)  
`wait()` (*computing.Waiter.VolumeInUse method*), [322](#)  
`wait()` (*computing.Waiter.VpnConnectionAvailable method*), [323](#)  
`wait()` (*computing.Waiter.VpnConnectionDeleted method*), [324](#)  
`wait()` (*computing.Waiter.VpnConnectionExists method*), [325](#)  
`wait()` (*computing.Waiter.VpnGatewayAvailable method*), [326](#)  
`wait()` (*computing.Waiter.VpnGatewayDeleted method*), [327](#)  
`wait()` (*computing.Waiter.VpnGatewayExists method*), [328](#)  
`wait()` (*computing.Waiter.VpnGatewayStopped method*), [329](#)  
`wait()` (*computing.Waiter.VpnGatewayWarning method*), [330](#)  
`wait()` (*hatoba.Waiter.ClusterDeleted method*), [421](#)  
`wait()` (*hatoba.Waiter.ClusterRunning method*), [422](#)  
`wait()` (*hatoba.Waiter.FirewallRuleAuthorized method*), [423](#)

`wait()` (*hatoba.Waiter.SnapshotAvailable method*), 423

`wait()` (*nas.Waiter.NASInstanceAvailable method*), 443

`wait()` (*nas.Waiter.NASInstanceDeleted method*), 444

`wait()` (*nas.Waiter.NASInstanceExists method*), 444

`wait()` (*nas.Waiter.NASInstanceFailed method*), 445

`wait()` (*nas.Waiter.NASInstanceStorageFull method*), 445

`wait()` (*nas.Waiter.NASSecurityGroupDeleted method*), 446

`wait()` (*nas.Waiter.NASSecurityGroupExists method*), 447

`wait()` (*nas.Waiter.NASSecurityGroupIPRangesAuthFailed method*), 447

`wait()` (*nas.Waiter.NASSecurityGroupIPRangesAuthorized method*), 448

`wait()` (*nas.Waiter.NASSecurityGroupIPRangesEmptied method*), 449

`wait()` (*nas.Waiter.NASSecurityGroupIPRangesRevokeFailed method*), 449

`wait()` (*nas.Waiter.NASSecurityGroupSecurityGroupsAuthFailed method*), 450

`wait()` (*nas.Waiter.NASSecurityGroupSecurityGroupsAuthorized method*), 451

`wait()` (*nas.Waiter.NASSecurityGroupSecurityGroupsEmptied method*), 451

`wait()` (*nas.Waiter.NASSecurityGroupSecurityGroupsRevokeFailed method*), 452

`wait()` (*rdp.Waiter.DBInstanceAvailable method*), 542

`wait()` (*rdp.Waiter.DBInstanceDeleted method*), 543

`wait()` (*rdp.Waiter.DBInstanceExists method*), 544

`wait()` (*rdp.Waiter.DBInstanceFailed method*), 545

`wait()` (*rdp.Waiter.DBInstanceStorageFull method*), 546

`wait()` (*rdp.Waiter.DBSecurityGroupDeleted method*), 547

`wait()` (*rdp.Waiter.DBSecurityGroupEC2SecurityGroupsAuthFailed method*), 547

`wait()` (*rdp.Waiter.DBSecurityGroupEC2SecurityGroupsAuthorized method*), 548

`wait()` (*rdp.Waiter.DBSecurityGroupEC2SecurityGroupsEmptied method*), 549

`wait()` (*rdp.Waiter.DBSecurityGroupEC2SecurityGroupsRevokeFailed method*), 550

`wait()` (*rdp.Waiter.DBSecurityGroupExists method*), 551

`wait()` (*rdp.Waiter.DBSecurityGroupIPRangesAuthFailed method*), 552

`wait()` (*rdp.Waiter.DBSecurityGroupIPRangesAuthorized method*), 553

`wait()` (*rdp.Waiter.DBSecurityGroupIPRangesEmptied method*), 554

`wait()` (*rdp.Waiter.DBSecurityGroupIPRangesRevokeFailed method*), 555