nifcloud-sdk-python Documentation

Release 1.6.0

FUJITSU CLOUD TECHNOLOGIES

Apr 20, 2023

Contents

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

Contents:

CHAPTER 1

Available Services

1.1 computing

Table of Contents

1.1.1 Client

class computing.**Client**

A low-level client representing NIFCLOUD Computing

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

These are the available methods:

- *allocate_address()*
- *associate_address()*
- *associate_multi_ip_address_group()*
- *associate_route_table()*
- *associate_users()*
- *attach_iso_image()*
- *attach_network_interface()*

- *attach_volume()*
- *authorize_security_group_ingress()*
- *can_paginate()*
- *cancel_copy_instances()*
- *cancel_upload()*
- *clear_load_balancer_session()*
- *configure_health_check()*
- *copy_from_backup_instance()*
- *copy_instances()*
- *create_backup_instances()*
- *create_customer_gateway()*
- *create_dhcp_options()*
- *create_image()*
- *create_instance_backup_rule()*
- *create_key_pair()*
- *create_load_balancer()*
- *create_multi_ip_address_group()*
- *create_network_interface()*
- *create_remote_access_vpn_gateway()*
- *create_remote_access_vpn_gateway_users()*
- *create_route()*
- *create_route_table()*
- *create_security_group()*
- *create_ssl_certificate()*
- *create_volume()*
- *create_vpn_connection()*
- *create_vpn_gateway()*
- *delete_customer_gateway()*
- *delete_dhcp_options()*
- *delete_image()*
- *delete_instance_backup_rule()*
- *delete_iso_image()*
- *delete_key_pair()*
- *delete_load_balancer()*
- *delete_multi_ip_address_group()*
- *delete_network_interface()*

- *delete_remote_access_vpn_gateway()*
- *delete_remote_access_vpn_gateway_connections()*
- *delete_remote_access_vpn_gateway_users()*
- *delete_route()*
- *delete_route_table()*
- *delete_security_group()*
- *delete_ssl_certificate()*
- *delete_volume()*
- *delete_vpn_connection()*
- *delete_vpn_gateway()*
- *deregister_instances_from_load_balancer()*
- *deregister_instances_from_security_group()*
- *describe_addresses()*
- *describe_associated_users()*
- *describe_availability_zones()*
- *describe_customer_gateways()*
- *describe_dhcp_options()*
- *describe_images()*
- *describe_instance_attribute()*
- *describe_instance_backup_rule_activities()*
- *describe_instance_backup_rules()*
- *describe_instance_health()*
- *describe_instances()*
- *describe_iso_images()*
- *describe_key_pairs()*
- *describe_load_balancers()*
- *describe_multi_ip_address_groups()*
- *describe_network_interfaces()*
- *describe_regions()*
- *describe_remote_access_vpn_gateway_activities()*
- *describe_remote_access_vpn_gateway_client_config()*
- *describe_remote_access_vpn_gateway_connections()*
- *describe_remote_access_vpn_gateways()*
- *describe_resources()*
- *describe_route_tables()*
- *describe_security_activities()*

- describe_security_groups()
- describe_service_status()
- describe_ssl_certificate_attribute()
- describe_ssl_certificates()
- describe_uploads()
- describe_usage()
- describe_user_activities()
- describe_volumes()
- describe_vpn_connections()
- describe_vpn_gateways()
- detach_iso_image()
- detach_network_interface()
- detach_volume()
- disassociate_address()
- disassociate_multi_ip_address_group()
- disassociate_route_table()
- dissociate_users()
- download_ssl_certificate()
- extend_volume_size()
- generate_presigned_url()
- get_paginator()
- get_waiter()
- import_instance()
- import_key_pair()
- increase_multi_ip_address_count()
- modify_image_attribute()
- modify_instance_attribute()
- modify_instance_backup_rule_attribute()
- modify_multi_ip_address_group_attribute()
- modify_network_interface_attribute()
- modify_remote_access_vpn_gateway_attribute()
- modify_remote_access_vpn_gateway_user_attribute()
- modify_ssl_certificate_attribute()
- modify_volume_attribute()
- nifty_associate_image()
- nifty_associate_nat_table()

- *nifty_associate_route_table_with_elastic_load_balancer()*
- *nifty_associate_route_table_with_vpn_gateway()*
- *nifty_configure_elastic_load_balancer_health_check()*
- *nifty_create_alarm()*
- *nifty_create_auto_scaling_group()*
- *nifty_create_dhcp_config()*
- *nifty_create_dhcp_ip_address_pool()*
- *nifty_create_dhcp_static_mapping()*
- *nifty_create_elastic_load_balancer()*
- *nifty_create_instance_snapshot()*
- *nifty_create_nat_rule()*
- *nifty_create_nat_table()*
- *nifty_create_private_lan()*
- *nifty_create_router()*
- *nifty_create_separate_instance_rule()*
- *nifty_create_web_proxy()*
- *nifty_delete_alarm()*
- *nifty_delete_auto_scaling_group()*
- *nifty_delete_dhcp_config()*
- *nifty_delete_dhcp_ip_address_pool()*
- *nifty_delete_dhcp_static_mapping()*
- *nifty_delete_elastic_load_balancer()*
- *nifty_delete_instance_snapshot()*
- *nifty_delete_nat_rule()*
- *nifty_delete_nat_table()*
- *nifty_delete_private_lan()*
- *nifty_delete_router()*
- *nifty_delete_separate_instance_rule()*
- *nifty_delete_web_proxy()*
- *nifty_deregister_instances_from_elastic_load_balancer()*
- *nifty_deregister_instances_from_separate_instance_rule()*
- *nifty_deregister_routers_from_security_group()*
- *nifty_deregister_vpn_gateways_from_security_group()*
- *nifty_describe_alarm_history()*
- *nifty_describe_alarm_rules_activities()*
- *nifty_describe_alarms()*

- *nifty_describe_alarms_partitions()*
- *nifty_describe_auto_scaling_groups()*
- *nifty_describe_corporate_info_for_certificate()*
- *nifty_describe_dhcp_configs()*
- *nifty_describe_dhcp_status()*
- *nifty_describe_elastic_load_balancers()*
- *nifty_describe_instance_elastic_load_balancer_health()*
- *nifty_describe_instance_snapshots()*
- *nifty_describe_load_balancer_ssl_policies()*
- *nifty_describe_nat_tables()*
- *nifty_describe_performance_chart()*
- *nifty_describe_private_lans()*
- *nifty_describe_routers()*
- *nifty_describe_scaling_activities()*
- *nifty_describe_separate_instance_rules()*
- *nifty_describe_vpn_gateway_activities()*
- *nifty_describe_web_proxies()*
- *nifty_disable_dhcp()*
- *nifty_disassociate_nat_table()*
- *nifty_disassociate_route_table_from_elastic_load_balancer()*
- *nifty_disassociate_route_table_from_vpn_gateway()*
- *nifty_enable_dhcp()*
- *nifty_modify_address_attribute()*
- *nifty_modify_customer_gateway_attribute()*
- *nifty_modify_elastic_load_balancer_attributes()*
- *nifty_modify_instance_snapshot_attribute()*
- *nifty_modify_key_pair_attribute()*
- *nifty_modify_private_lan_attribute()*
- *nifty_modify_router_attribute()*
- *nifty_modify_vpn_gateway_attribute()*
- *nifty_modify_web_proxy_attribute()*
- *nifty_reboot_routers()*
- *nifty_reboot_vpn_gateways()*
- *nifty_register_instances_with_elastic_load_balancer()*
- *nifty_register_instances_with_separate_instance_rule()*
- *nifty_register_port_with_elastic_load_balancer()*

- *nifty_register_routers_with_security_group()*
- *nifty_register_vpn_gateways_with_security_group()*
- *nifty_release_router_backup_state()*
- *nifty_release_vpn_gateway_backup_state()*
- *nifty_replace_dhcp_config()*
- *nifty_replace_dhcp_option()*
- *nifty_replace_elastic_load_balancer_latest_version()*
- *nifty_replace_elastic_load_balancer_listener_ssl_certificate()*
- *nifty_replace_nat_rule()*
- *nifty_replace_nat_table_association()*
- *nifty_replace_route_table_association_with_elastic_load_balancer()*
- *nifty_replace_route_table_association_with_vpn_gateway()*
- *nifty_replace_router_latest_version()*
- *nifty_replace_vpn_gateway_latest_version()*
- *nifty_restore_instance_snapshot()*
- *nifty_restore_router_previous_version()*
- *nifty_restore_vpn_gateway_previous_version()*
- *nifty_retry_import_instance()*
- *nifty_set_load_balancer_ssl_policies_of_listener()*
- *nifty_unset_load_balancer_ssl_policies_of_listener()*
- *nifty_update_alarm()*
- *nifty_update_auto_scaling_group()*
- *nifty_update_elastic_load_balancer()*
- *nifty_update_instance_network_interfaces()*
- *nifty_update_router_network_interfaces()*
- *nifty_update_separate_instance_rule()*
- *nifty_update_vpn_gateway_network_interfaces()*
- *reboot_instances()*
- *reboot_remote_access_vpn_gateway()*
- *refresh_instance_backup_rule()*
- *register_corporate_info_for_certificate()*
- *register_instances_with_load_balancer()*
- *register_instances_with_security_group()*
- *register_port_with_load_balancer()*
- *release_address()*
- *release_multi_ip_addresses()*

- *replace_remote_access_vpn_gateway_latest_version()*
- *replace_route()*
- *replace_route_table_association()*
- *revoke_security_group_ingress()*
- *run_instances()*
- *set_filter_for_load_balancer()*
- *set_load_balancer_listener_ssl_certificate()*
- *set_remote_access_vpn_gateway_ca_certificate()*
- *set_remote_access_vpn_gateway_ssl_certificate()*
- *start_instances()*
- *stop_instances()*
- *terminate_instances()*
- *unset_load_balancer_listener_ssl_certificate()*
- *unset_remote_access_vpn_gateway_ca_certificate()*
- *unset_remote_access_vpn_gateway_ssl_certificate()*
- *update_load_balancer()*
- *update_load_balancer_option()*
- *update_security_group()*
- *upload_iso_image()*
- *upload_ssl_certificate()*

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)* –

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)* –

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)* –

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',
```

(continues on next page)

(continued from previous page)

```
    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)* –

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)* –

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)* –

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)* –

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)* –

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
        },
```

(continues on next page)

(continued from previous page)

```
    ],
    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)* –

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.

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)* –

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)* –

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)* –

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',
    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)* –

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'|
↪septa-large128'|'septa-large128',
```

(continues on next page)

(continued from previous page)

```
    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'
                }
            },
```

(continues on next page)

(continued from previous page)

```
        ],
        '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',
                    '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',
```

(continues on next page)

(continued from previous page)

```
                            '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': [
            {
                '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)* –

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-
```

(continues on next page)

(continued from previous page)

```
        'IpType': 'static'|'none',
        'ListOfRequestLoadBalancers': [
            {
                'InstancePort': 123,
                '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)* –

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'
                    },
                ],
                'InstanceId': 'string',
```

(continues on next page)

(continued from previous page)

```
                '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)* –

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)* –

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',
            ]
```

(continues on next page)

(continued from previous page)

```
        },
    ]
)
```

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)* –

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',
```

(continues on next page)

(continued from previous page)

```
        '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)* –

create_instance_backup_rule(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.create_instance_backup_rule(
    BackupInstanceMaxCount=123,
    Description='string',
    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)* –

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)* –

create_load_balancer(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

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

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'
```

(continues on next page)

(continued from previous page)

```
        },
        '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)* –

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',
```

(continues on next page)

(continued from previous page)

```
            'PublicIp': 'string',
            'PublicIpV6': 'string'
        },
        'Attachment': {
            'AttachTime': 'string',
            'AttachmentId': 'string',
            'DeleteOnTermination': 'string',
            'DeviceIndex': 'string',
            'InstanceId': 'string',
            'InstanceOwnerId': 'string',
            'Status': 'string'
        },
        'AvailabilityZone': 'string',
        'Description': 'string',
        'GroupSet': [
            {
                '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'
```

(continues on next page)

(continued from previous page)

```
            },
        ],
        '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)* –

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'
            },
        ],
        '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',
```

(continues on next page)

(continued from previous page)

```
                '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': {
            '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)* –

`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'
        },
    ]
)
```

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)* –

create_route(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

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

(continues on next page)

(continued from previous page)

```
    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)* –

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',
        'RouteTableId': 'string',
        'TagSet': [
            {
                'Key': 'string',
```

(continues on next page)

(continued from previous page)

```
                'Value': 'string'
            },
        ]
    }
}
```

Response Structure

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

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)* –

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

Response Structure

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

– **ValidityTerm** *(integer)* –

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',
    '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)* –

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',
    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',
            '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',
```

(continues on next page)

(continued from previous page)

```
        '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)* –

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',
```

(continues on next page)

(continued from previous page)

```
                '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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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

Response Structure

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

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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',
```

(continues on next page)

(continued from previous page)

```
                'InstanceUniqueId': 'string'
            },
        ]
    },
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

Response Structure

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

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)* –

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)* –

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

Response Structure

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

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)* –

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': [
        {
            '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)* –

describe_dhcp_options (***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

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

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',
            'ImageState': 'string',
```

(continues on next page)

(continued from previous page)

```
            '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)* –

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': {
        'Value': 'string'
    },
    'Autoscaling': {
        'AutoScalingGroupName': 'string',
        'ExpireTime': datetime(2015, 1, 1)
    },
```

(continues on next page)

(continued from previous page)

```
    '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',
                'DeleteOnTermination': 'string',
                'DeviceIndex': 'string',
                'Status': 'string'
            },
            'Description': 'string',
            'GroupSet': 'string',
```

(continues on next page)

(continued from previous page)

```
            '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)* –

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)* –
- · **EndDateTime** *(string)* –
- · **InstanceUniqueId** *(string)* –
- · **Operation** *(string)* –
- · **StartDateTime** *(string)* –
- · **Status** *(string)* –

- **InstanceBackupRuleId** *(string)* –
- **InstanceBackupRuleName** *(string)* –
- **RequestId** *(string)* –

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'
                },
            ],
            '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)* –

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': {
```

(continues on next page)

(continued from previous page)

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

Response Structure

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

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': {
```

(continues on next page)

(continued from previous page)

```
                        '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',
                    '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'
                    },
```

(continues on next page)

(continued from previous page)

```
                    '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',
                            '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',
```

(continues on next page)

(continued from previous page)

```
                            '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',
                    '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)* –

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)* –

describe_key_pairs (***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

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

(continues on next page)

(continued from previous page)

```
    ]
)
```

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)* –

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': {
                    '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,
```

(continues on next page)

(continued from previous page)

```
                            '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
                    }
                },
                '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)* –

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)* –

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': {
```

(continues on next page)

(continued from previous page)

```
                'AttachTime': 'string',
                'AttachmentId': 'string',
                'DeleteOnTermination': 'string',
                'DeviceIndex': 'string',
                'InstanceId': 'string',
                'InstanceOwnerId': 'string',
                'Status': 'string'
            },
            'AvailabilityZone': 'string',
            'Description': 'string',
            'GroupSet': [
                {
                    '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'
        },
```

(continues on next page)

(continued from previous page)

```
    ],
    '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)* –

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)* –

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)* –

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',
```

(continues on next page)

(continued from previous page)

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

Response Structure

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

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)* –

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',
                        'PublicDnsName': 'string',
                        'PublicIp': 'string',
                        'PublicIpV6': 'string'
                    },
                    'Attachment': {
                        'AttachTime': 'string',
                        'AttachmentId': 'string',
                        'DeleteOnTermination': 'string',
                        'DeviceIndex': 'string',
                        'InstanceId': 'string',
                        'InstanceOwnerId': 'string',
                        'Status': 'string'
```

(continues on next page)

(continued from previous page)

```
                    },
                    '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'
            }
        },
    ],
    '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)* –

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'
            },
        ],
        'ElasticLoadBalancerCount': 123,
        'InstanceBackupRuleCount': 123,
        'InstanceItemSet': [
            {
                'Count': 123,
                'Type': 'string'
            },
        ],
        'LoadBalancerCount': 123,
        'MigrationHubItemSet': [
            {
                'Count': 123,
                'Type': 'string'
            },
        ],
        'MonitoringRuleCount': 123,
        'MultiIpAddressItemSet': [
            {
```

(continues on next page)

(continued from previous page)

```
                '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'
            },
        ]
    }
}
```

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)* –

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'
```

(continues on next page)

(continued from previous page)

```
        },
    ],
    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',
                    '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',
```

(continues on next page)

(continued from previous page)

```
                    '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)* –

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)* –

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,
            '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': [
```

(continues on next page)

(continued from previous page)

```
                {
                    '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)* –

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)* –

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'
```

(continues on next page)

(continued from previous page)

```
)
```

Parameters

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

Return type dict

Returns

Response Syntax

```
{
    'CaState': {
        '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)* –

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',
```

(continues on next page)

(continued from previous page)

```
                'LocationName': 'string',
                'OrganizationName': 'string',
                'OrganizationUnitName': 'string',
                'StateName': 'string'
            },
            'CertState': 'string',
            'Count': 123,
            '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)* –

describe_uploads(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.describe_uploads(
    ConversionTaskId=[
```

(continues on next page)

(continued from previous page)

```
        'string',
    ]
)
```

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)* –

describe_usage (***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.describe_usage(
    IsCharge=True|False,
    Region='string',
    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,
            'Value': 123
        }
    },
    'CopyInfo': {
        'InstanceCopy': {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
```

(continues on next page)

(continued from previous page)

```
            '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': [
            {
                'Charge': 123,
                'Type': 'string',
                'Unit': 'string',
                'Value': 123
            },
        ]
    },
```

(continues on next page)

(continued from previous page)

```
    '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': [
            {
                'Charge': 123,
                'Type': 'string',
                'Unit': 'string',
                'Value': 123
            },
        ],
        'OsMonthlyRate': [
            {
```

(continues on next page)

(continued from previous page)

```
                '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': {
        'LiveMigrationHubMonthlyRateSet': [
            {
                'Charge': 123,
                'Type': 'string',
                'Unit': 'string',
                'Value': 123
            },
        ],
```

(continues on next page)

(continued from previous page)

```
        '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
            },
        ]
    },
    'NetworkInfo': {
        'NetworkFlowSet': [
            {
                'Charge': 123,
                'Type': 'string',
```

(continues on next page)

(continued from previous page)

```
                '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
            },
        ]
    },
    'PatternAuthInfo': {
        'PatternAuthSet': [
            {
                'Charge': 123,
                'Type': 'string',
                'Unit': 'string',
```

(continues on next page)

(continued from previous page)

```
                '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
            },
        ]
    },
    'RequestId': 'string',
    'RouterInfo': {
        'RouterMeasuredRateSet': [
            {
                'Charge': 123,
                'Type': 'string',
```

(continues on next page)

(continued from previous page)

```
                '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,
            'Type': 'string',
            'Unit': 'string',
            'Value': 123
        },
        'ImportInstanceDiskMonthlyRate': {
            'Charge': 123,
            'Type': 'string',
            'Unit': 'string',
```

(continues on next page)

(continued from previous page)

```
            '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)* –

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'
        },
    ]
```

(continues on next page)

(continued from previous page)

```
}
```

Response Structure

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

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': [
        {
            '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',
```

(continues on next page)

(continued from previous page)

```
            '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)* –

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=[
```

(continues on next page)

(continued from previous page)

```
        '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',
            'NiftyIpsecConfiguration': {
                'DiffieHellmanGroup': 123,
                'EncapsulatingSecurityPayloadLifetime': 123,
                'EncryptionAlgorithm': 'string',
                'HashingAlgorithm': 'string',
                'InternetKeyExchange': 'string',
                'InternetKeyExchangeLifetime': 123,
                'Mtu': 'string',
                'PreSharedKey': 'string'
            },
            'NiftyTunnel': {
                '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': [
                {
                    'Key': 'string',
                    'Value': 'string'
                },
            ],
            'Type': 'string',
```

(continues on next page)

(continued from previous page)

```
            '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)* –

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
            },
```

(continues on next page)

(continued from previous page)

```
            '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'
        },
    ]
}
```

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)* –
- **Descriprion** *(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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

generate_presigned_url(*ClientMethod*, *Params=None*, *ExpiresIn=3600*, *HttpMethod=None*)

Generate a presigned url given a client, its method, and arguments

Parameters

- **ClientMethod** (*string*) – The client method to presign for
- **Params** (*dict*) – The parameters normally passed to `ClientMethod`.
- **ExpiresIn** (*int*) – The number of seconds the presigned url is valid for. By default it expires in an hour (3600 seconds)
- **HttpMethod** (*string*) – The http method to use on the generated url. By default, the http method is whatever is used in the method's model.

Returns The presigned url

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.

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

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'|'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',
    Monitoring={
        'Enabled': 'string'
    },
    NetworkInterface=[
        {
            'DeviceIndex': 123,
```

(continues on next page)

(continued from previous page)

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

(continues on next page)

(continued from previous page)

```
        ],
        '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)* –

import_key_pair(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.import_key_pair(
    Description='string',
```

(continues on next page)

(continued from previous page)

```
    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)* –

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)* –

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)* –

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='dafault'|'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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

modify_volume_attribute(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.modify_volume_attribute(
    Attribute='accountingType'|'volumeName'|'description',
```

(continues on next page)

(continued from previous page)

```
    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)* –

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

Response Structure

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

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)* –

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)* –

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)* –

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': 123
            },
        ],
        '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** *(integer)* –
  - **Path** *(string)* –
  - **Target** *(string)* – **[REQUIRED]**
  - **UnhealthyThreshold** *(integer)* – **[REQUIRED]**
- **InstancePort** (*integer*) – **[REQUIRED]**
- **Protocol** (*string*) – **[REQUIRED]**

Return type dict

Returns

Response Syntax

```
{
    'NiftyConfigureElasticLoadBalancerHealthCheckResult': {
        'HealthCheck': {
            'Expectation': [
                {
                    'HttpCode': 123
                },
            ],
            'Interval': 123,
            'Path': 'string',
            'Target': 'string',
            'UnhealthyThreshold': 123
        }
    },
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

Response Structure

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

nifty_create_alarm(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.nifty_create_alarm(
    AlarmCondition='and'|'or',
    Description='string',
```

(continues on next page)

(continued from previous page)

```
    ElasticLoadBalancerName=[
        'string',
    ],
    ElasticLoadBalancerPort=[
        123,
    ],
    ElasticLoadBalancerProtocol=[
        'string',
    ],
    EmailAddress=[
        'string',
    ],
    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',
    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

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

Response Structure

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

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-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',
```

(continues on next page)

(continued from previous page)

```
    LoadBalancers=[
        {
            'InstancePort': 123,
            'LoadBalancerPort': 123,
            'Name': 'string'
        },
    ],
    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]**
- **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)* –

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)* –

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)* –

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)* –

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',
            'RequestHealthCheck': {
                'Interval': 123,
                'ListOfRequestExpectation': [
                    {
                        'HttpCode': 123
                    },
```

(continues on next page)

(continued from previous page)

```
                ],
                '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,
            '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** *(integer)* –
      - **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)* –
    * **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)* –

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)* –

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'
    },
    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)* –

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)* –

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': {
        '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',
        'PrivateLanName': 'string',
        'RemoteAccessVpnGatewaySet': [
            {
                'DeviceIndex': 'string',
                'IpAddress': 'string',
                'RemoteAccessVpnGatewayId': 'string',
                'RemoteAccessVpnGatewayName': 'string'
```

(continues on next page)

(continued from previous page)

```
            },
        ],
        '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)* –
  - **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)* –

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'
        }
```

(continues on next page)

(continued from previous page)

```
    }
}
```

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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

Response Structure

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

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',
            'UpdatedTime': '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)* –

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',
```

(continues on next page)

(continued from previous page)

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

Response Structure

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

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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'
        },
    ]
}
```

Response Structure

- *(dict)* –
  - **RequestId** *(string)* –
  - **RouterSet** *(list)* –
    - *(dict)* –
      - **RouterId** *(string)* –
      - **RouterName** *(string)* –

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)* –

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)* –

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': [
                                {
                                    '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)* –
      - **AlarmRulesActivitiesDateSet** *(list)* –
      - *(dict)* –
      - **AlarmRulesActivitiesEventSet** *(list)* –
      - *(dict)* –
      - **AlarmRulesActivitiesEvent** *(string)* –
      - **AlarmRulesActivitiesEventDatetime** *(datetime)* –
      - **ResourceName** *(string)* –
      - **Value** *(string)* –
      - **Date** *(string)* –
      - **DataType** *(string)* –
      - **FunctionName** *(string)* –

· **RuleName** *(string)* –

nifty_describe_alarms(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.nifty_describe_alarms(
    Rule=[
        {
            'FunctionName': 'Server'|'LoadBalancer'|'DiskPartition'|
↪'ElasticLoadBalancer',
            '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)* –

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)* –

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)* –

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)* –

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',
                    'StopIpAddress': 'string'
                },
            ],
            'StaticMappingsSet': [
                {
                    'Description': 'string',
                    'IpAddress': 'string',
                    'MacAddress': 'string'
                },
            ]
        },
    ],
    'RequestId': 'string'
```

(continues on next page)

(continued from previous page)

```
}
```

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)* –

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',
```

(continues on next page)

(continued from previous page)

```
                        '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)* –

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,
```

(continues on next page)

(continued from previous page)

```
        ],
        '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',
```

(continues on next page)

(continued from previous page)

```
                            'ElasticLoadBalancerPort': 123,
                            'HealthCheck': {
                                'Expectation': [
                                    {
                                        'HttpCode': 123
                                    },
                                ],
                                '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'
                            }
                        }
                    },
                ],
                'ElasticLoadBalancerName': 'string',
                'NetworkInterfaces': [
                    {
                        'Description': 'string',
                        'DeviceIndex': 'string',
                        'IpAddress': 'string',
                        'IsVipNetwork': True|False,
                        'NetworkId': 'string',
                        'NetworkName': 'string'
                    },
                ],
                'NetworkVolume': 'string',
```

(continues on next page)

(continued from previous page)

```
                '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** *(integer)* –
      - **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)* –
- **NetworkVolume** *(string)* –
- **NextMonthAccountingType** *(string)* –
- **RouteTableAssociationId** *(string)* –
- **RouteTableId** *(string)* –
- **State** *(string)* –
- **VersionInformation** *(dict)* –
- **IsLatest** *(boolean)* –
- **Version** *(string)* –

– **ResponseMetadata** *(dict)* –

* **RequestId** *(string)* –

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)* –

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',
            'UpdatedTime': '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)* –

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,
```

(continues on next page)

(continued from previous page)

```
                '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)* –

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': [
                {
                    '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'
```

(continues on next page)

(continued from previous page)

```
}
```

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)* –

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',
    ],
```

(continues on next page)

(continued from previous page)

```
    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'
                },
            ],
            'DataType': '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)* –

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'
        },
    ],
    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'
                },
```

(continues on next page)

(continued from previous page)

```
            ],
            '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',
                    '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)* –

nifty_describe_routers(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.nifty_describe_routers(
    Filter=[
        {
            'ListOfRequestValue': [
                'string',
```

(continues on next page)

(continued from previous page)

```
            ],
            'Name': 'availability-zone'|'state'|'router-id'|'router-name'|
↪'description'|'accountingType'|'type'|'ip-address'|'version'|'latest-
↪version-information'
        },
    ],
    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',
```

(continues on next page)

(continued from previous page)

```
                    'IpAddress': 'string',
                    'NetworkId': 'string',
                    'NetworkName': 'string'
                },
            ],
            '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)* –

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',
            'Time': datetime(2015, 1, 1)
        },
```

(continues on next page)

(continued from previous page)

```
    ],
    '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)* –

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',
```

(continues on next page)

(continued from previous page)

```
            '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)* –

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'
```

(continues on next page)

(continued from previous page)

```
}
```

Response Structure

- *(dict)* –
  - **AnalyzeResultSet** *(list)* –
    - *(dict)* –
      - **AnalyzeCode** *(string)* –
      - **Line** *(string)* –
  - **Log** *(string)* –
  - **NiftyVpnGatewayName** *(string)* –
  - **RequestId** *(string)* –
  - **VpnGatewayId** *(string)* –

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': [
        {
```

(continues on next page)

(continued from previous page)

```
            '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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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',
```

(continues on next page)

(continued from previous page)

```
    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)* –

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)* –

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

Response Structure

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

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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'
        },
```

(continues on next page)

(continued from previous page)

```
    ]
)
```

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)* –

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)* –

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)* –

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,
            '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)* –

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)* –

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',
    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)* –

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)* –

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)* –

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)* –

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)* –

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'
)
```

Parameters

- **ElasticLoadBalancerId** (*string*) –
- **ElasticLoadBalancerName** (*string*) –

Return type dict

Returns

Response Syntax

```
{
    'NiftyReplaceElasticLoadBalancerLatestVersionResult':
↪'string',
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

Response Structure

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

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',
```

(continues on next page)

(continued from previous page)

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

Response Structure

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

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)* –

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)* –

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

Response Structure

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

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)* –

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',
    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)* –

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)* –

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)* –

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)* –

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',
```

(continues on next page)

(continued from previous page)

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

Response Structure

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

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)* –

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)* –

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)* –

nifty_update_alarm(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.nifty_update_alarm(
    AlarmCondition='and'|'or',
    Description='string',
    ElasticLoadBalancerName=[
        'string',
```

(continues on next page)

(continued from previous page)

```
    ],
    ElasticLoadBalancerPort=[
        123,
    ],
    ElasticLoadBalancerProtocol=[
        'string',
    ],
    EmailAddress=[
        'string',
    ],
    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)* –

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',
```

(continues on next page)

(continued from previous page)

```
    LoadBalancers=[
        {
            'InstancePort': 123,
            'LoadBalancerPort': 123,
            'Name': 'string'
        },
    ],
    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)* –

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)* –

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)* –

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)* –

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)* –

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'
    },
```

(continues on next page)

(continued from previous page)

```
    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)* –

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)* –

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)* –

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)* –

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',
    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',
    '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)* –

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',
    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)* –

register_instances_with_security_group(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.register_instances_with_security_group(
    GroupName='string',
    InstanceId=[
        'string',
```

(continues on next page)

(continued from previous page)

```
    ]
)
```

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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': [
                {
                    '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)* –

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',
```

(continues on next page)

(continued from previous page)

```
    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
    },
    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': [
                {
```

(continues on next page)

(continued from previous page)

```
                'DeviceName': 'string',
                'Ebs': {
                    'DeleteOnTermination': 'string',
                    'Status': 'string',
                    'VolumeId': 'string',
                    'VolumeUniqueId': 'string'
                }
            },
        ],
        'Description': 'string',
        'DnsName': 'string',
        'ImageId': 'string',
        'InstanceId': 'string',
        'InstanceState': {
            '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',
```

(continues on next page)

(continued from previous page)

```
                    'PrivateDnsName': 'string',
                    'PrivateIpAddressesSet': [
                        {
                            'Association': {
                                'IpOwnerId': 'string',
                                'PublicDnsName': 'string',
                                'PublicIp': 'string'
                            },
                            'Primary': True|False,
                            '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',
            '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)* –

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)* –

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)* –

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)* –

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)* –

start_instances(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.start_instances(
    AccountingType=[
        'string',
    ],
    InstanceId=[
        'string',
    ],
    InstanceType=[
        'string',
    ],
```

(continues on next page)

(continued from previous page)

```
    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)* –

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,
                '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)* –

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': {
                '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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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)* –

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

The available waiters are:

- *computing.Waiter.CustomerGatewayAvailable*
- *computing.Waiter.CustomerGatewayDeleted*
- *computing.Waiter.CustomerGatewayExists*
- *computing.Waiter.CustomerGatewayStopped*
- *computing.Waiter.CustomerGatewayWarning*
- *computing.Waiter.ElasticLoadBalancerAvailable*
- *computing.Waiter.ElasticLoadBalancerDeleted*
- *computing.Waiter.ElasticLoadBalancerExists*
- *computing.Waiter.InstanceDeleted*
- *computing.Waiter.InstanceExists*
- *computing.Waiter.InstanceImportError*
- *computing.Waiter.InstanceRunning*
- *computing.Waiter.InstanceStopped*
- *computing.Waiter.InstanceSuspending*
- *computing.Waiter.InstanceWarning*
- *computing.Waiter.LoadBalancerDeleted*
- *computing.Waiter.LoadBalancerExists*
- *computing.Waiter.LoadBalancerInService*
- *computing.Waiter.PrivateLanAvailable*
- *computing.Waiter.PrivateLanDeleted*
- *computing.Waiter.PrivateLanExists*
- *computing.Waiter.RemoteAccessVpnGatewayAvailable*
- *computing.Waiter.RemoteAccessVpnGatewayDeleted*
- *computing.Waiter.RemoteAccessVpnGatewayExists*
- *computing.Waiter.RouterAvailable*
- *computing.Waiter.RouterDeleted*
- *computing.Waiter.RouterExists*
- *computing.Waiter.RouterStopped*
- *computing.Waiter.RouterWarning*
- *computing.Waiter.SecurityGroupApplied*
- *computing.Waiter.SecurityGroupDeleted*
- *computing.Waiter.SecurityGroupExists*
- *computing.Waiter.SnapshotDeleted*
- *computing.Waiter.SnapshotExists*

- *computing.Waiter.SnapshotNormal*
- *computing.Waiter.VolumeAttached*
- *computing.Waiter.VolumeAvailable*
- *computing.Waiter.VolumeDeleted*
- *computing.Waiter.VolumeExists*
- *computing.Waiter.VolumeInUse*
- *computing.Waiter.VpnConnectionAvailable*
- *computing.Waiter.VpnConnectionDeleted*
- *computing.Waiter.VpnConnectionExists*
- *computing.Waiter.VpnGatewayAvailable*
- *computing.Waiter.VpnGatewayDeleted*
- *computing.Waiter.VpnGatewayExists*
- *computing.Waiter.VpnGatewayStopped*
- *computing.Waiter.VpnGatewayWarning*

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

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

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

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

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

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'
        },
    ],
    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

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': [
            '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={
```

(continues on next page)

(continued from previous page)

```
        '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

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,
        ],
```

(continues on next page)

(continued from previous page)

```
        '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

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

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

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

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

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',
    ],
```

(continues on next page)

(continued from previous page)

```
    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

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

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

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

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
```

(continues on next page)

(continued from previous page)

```
    }
)
```

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

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
        },
    ],
    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

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=[
        '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

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'
        },
    ],
    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

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'|'accountingType'|
↪'description'
        },
    ],
    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

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={
        '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

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

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

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

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=[
```

(continues on next page)

(continued from previous page)

```
        {
            '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

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=[
        {
```

(continues on next page)

(continued from previous page)

```
            '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

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': [
```

(continues on next page)

(continued from previous page)

```
                '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

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',
```

(continues on next page)

(continued from previous page)

```
            ],
            '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

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=[
        {
            'ListOfRequestValue': [
                'string',
            ],
```

(continues on next page)

(continued from previous page)

```
            '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

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,
```

(continues on next page)

(continued from previous page)

```
        '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

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

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',
    ],
    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

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

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
```

(continues on next page)

(continued from previous page)

```
    }
)
```

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

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

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',
    ],
    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

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

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

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

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',
    ],
    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

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',
    ],
    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

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=[
        '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

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=[
        '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

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-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

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',
            ],
            '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

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

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

Table of Contents

1.2.1 Client

class dns.**Client**

A low-level client representing NIFCLOUD DNS

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

These are the available methods:

- *can_paginate()*
- *change_resource_record_sets()*
- *create_hosted_zone()*
- *delete_hosted_zone()*
- *generate_presigned_url()*
- *get_change()*
- *get_hosted_zone()*
- *get_paginator()*
- *get_waiter()*
- *list_hosted_zones()*
- *list_resource_record_sets()*

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.

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': [
```

(continues on next page)

(continued from previous page)

```
                                {
                                    '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,
                            '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)* –

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'
```

(continues on next page)

(continued from previous page)

```
        },
        '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)* –

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)* –

generate_presigned_url(*ClientMethod*, *Params=None*, *ExpiresIn=3600*, *HttpMethod=None*)

Generate a presigned url given a client, its method, and arguments

Parameters

- **ClientMethod** (*string*) – The client method to presign for
- **Params** (*dict*) – The parameters normally passed to `ClientMethod`.
- **ExpiresIn** (*int*) – The number of seconds the presigned url is valid for. By default it expires in an hour (3600 seconds)
- **HttpMethod** (*string*) – The http method to use on the generated url. By default, the http method is whatever is used in the method's model.

Returns The presigned url

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)* –

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',
```

(continues on next page)

(continued from previous page)

```
        ]
    },
    'HostedZone': {
        'CallerReference': 'string',
        'Config': {
            'Comment': 'string'
        },
        '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)* –

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.

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

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)* –

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

Table of Contents

1.3.1 Client

class ess.**Client**

A low-level client representing NIFCLOUD ESS

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

These are the available methods:

- *can_paginate()*
- *delete_identity()*
- *generate_presigned_url()*
- *get_delivery_log()*
- *get_identity_dkim_attributes()*
- *get_identity_verification_attributes()*
- *get_paginator()*
- *get_send_quota()*
- *get_send_statistics()*
- *get_waiter()*
- *list_identities()*
- *send_email()*
- *send_raw_email()*
- *set_identity_dkim_enabled()*
- *verify_domain_dkim()*
- *verify_domain_identity()*
- *verify_email_identity()*

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.

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)* –

generate_presigned_url (*ClientMethod*, *Params=None*, *ExpiresIn=3600*, *HttpMethod=None*)

Generate a presigned url given a client, its method, and arguments

Parameters

- **ClientMethod** (*string*) – The client method to presign for
- **Params** (*dict*) – The parameters normally passed to `ClientMethod`.
- **ExpiresIn** (*int*) – The number of seconds the presigned url is valid for. By default it expires in an hour (3600 seconds)
- **HttpMethod** (*string*) – The http method to use on the generated url. By default, the http method is whatever is used in the method's model.

Returns The presigned url

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),
```

(continues on next page)

(continued from previous page)

```
    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)* –

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)* –

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',
                'VerificationToken': 'string'
            }
        },
    ]
}
```

Response Structure

- *(dict)* –
  - **ResponseMetadata** *(dict)* –
    - **RequestId** *(string)* –
  - **VerificationAttributes** *(list)* –
    - *(dict)* –
      - **Key** *(string)* –
      - **Value** *(dict)* –
      - **VerificationStatus** *(string)* –
      - **VerificationToken** *(string)* –

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.

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'
    },
    'SentLast24Hours': 123.0
}
```

Response Structure

- *(dict)* –
  - **Max24HourSend** *(float)* –
  - **MaxSendRate** *(float)* –
  - **ResponseMetadata** *(dict)* –
    - **RequestId** *(string)* –
  - **SentLast24Hours** *(float)* –

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)* –

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

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)* –

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',
    ],
```

(continues on next page)

(continued from previous page)

```
    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)* –

send_raw_email(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.send_raw_email(
    Destinations=[
        'string',
    ],
    RawMessage={
```

(continues on next page)

(continued from previous page)

```
        '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)* –

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)* –

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)* –

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)* –

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

Table of Contents

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:

- *attach_disk()*
- *authorize_firewall_group()*
- *can_paginate()*
- *create_cluster()*
- *create_disk()*
- *create_firewall_group()*
- *create_node_pool()*
- *create_snapshot()*
- *create_tags()*
- *delete_cluster()*
- *delete_clusters()*
- *delete_disk()*
- *delete_disks()*
- *delete_firewall_group()*
- *delete_firewall_groups()*
- *delete_node_pool()*
- *delete_node_pools()*
- *delete_snapshot()*
- *delete_snapshots()*
- *delete_tags()*
- *detach_disk()*
- *generate_presigned_url()*
- *get_cluster()*
- *get_cluster_credentials()*
- *get_disk()*
- *get_firewall_group()*
- *get_load_balancer()*
- *get_node_pool()*
- *get_paginator()*
- *get_server_config()*
- *get_snapshot()*
- *get_waiter()*
- *list_clusters()*
- *list_disks()*
- *list_firewall_groups()*
- *list_load_balancers()*
- *list_node_pools()*
- *list_snapshots()*
- *list_tags()*
- *reboot_node()*
- *restore_cluster_from_snapshot()*
- *revoke_firewall_group()*
- *set_node_pool_size()*
- *update_cluster()*
- *update_disk()*
- *update_firewall_group()*
- *update_node_pool()*
- *update_snapshot()*
- *update_tags()*

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)* –

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': [
            {
```

(continues on next page)

(continued from previous page)

```
                '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)* –

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.

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'
            },
        ],
        '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',
        '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'
                    },
                ],
```

(continues on next page)

(continued from previous page)

```
                '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)* –

create_disk(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.create_disk(
    Disk={
        'AvailabilityZone': 'string',
        'Description': 'string',
        'ListOfRequestTags': [
            {
                'Key': 'string',
                'Value': 'string'
            },
        ],
        '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',
```

(continues on next page)

(continued from previous page)

```
                '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)* –

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)* –

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',
        '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)* –

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'
        }
    }
)
```

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'
```

(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)* –

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)* –

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': [
```

(continues on next page)

(continued from previous page)

```
            {
                '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)* –

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': [
```

(continues on next page)

(continued from previous page)

```
                '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)* –

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': [
            {
```

(continues on next page)

(continued from previous page)

```
                '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)* –

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)* –

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
            },
        ],
        '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)* –

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': [
        {
            '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'
                },
```

(continues on next page)

(continued from previous page)

```
            ]
        },
    ]
}
```

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)* –

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',
```

(continues on next page)

(continued from previous page)

```
                '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)* –

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)* –

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',
        '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)* –

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)* –

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)* –

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)* –

generate_presigned_url(*ClientMethod*, *Params=None*, *ExpiresIn=3600*, *HttpMethod=None*)

Generate a presigned url given a client, its method, and arguments

Parameters

- **ClientMethod** (*string*) – The client method to presign for
- **Params** (*dict*) – The parameters normally passed to `ClientMethod`.
- **ExpiresIn** (*int*) – The number of seconds the presigned url is valid for. By default it expires in an hour (3600 seconds)
- **HttpMethod** (*string*) – The http method to use on the generated url. By default, the http method is whatever is used in the method's model.

Returns The presigned url

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',
        ],
```

(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)* –
  - **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)* –

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)* –

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)* –

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)* –

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,
                        '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** *(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)* –

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)* –

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.

`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)* –

`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': [
```

(continues on next page)

(continued from previous page)

```
                {
                    '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)* –

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

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',
            '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)* –
  - **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)* –

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)* –

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
                },
            ],
```

(continues on next page)

(continued from previous page)

```
            '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)* –

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',
```

(continues on next page)

(continued from previous page)

```
                    'NodePools': [
                        {
                            'Name': 'string',
                            'NodeCount': 123,
                            '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'
```

(continues on next page)

(continued from previous page)

```
        },
    ]
}
```

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)* –

list_node_pools(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.list_node_pools(
    ClusterName='string',
    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)* –

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'
                },
            ]
        },
```

(continues on next page)

(continued from previous page)

```
    ]
}
```

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)* –

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)* –

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)* –

restore_cluster_from_snapshot (***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.restore_cluster_from_snapshot(
    Cluster={
        'Description': 'string',
        'FirewallGroup': 'string',
        'ListOfRequestLocations': [
```

(continues on next page)

(continued from previous page)

```
            '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': {
        'AddonsConfig': {
            'HttpLoadBalancing': {
                'Disabled': True|False
            }
        },
        'CreateTime': 'string',
        'Description': 'string',
        'FirewallGroup': 'string',
        'InitialKubernetesVersion': 'string',
        'InitialNodeCount': 123,
```

(continues on next page)

(continued from previous page)

```
        '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)* –

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)* –

set_node_pool_size(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.set_node_pool_size(
    ClusterName='string',
```

(continues on next page)

(continued from previous page)

```
    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)* –

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',
```

(continues on next page)

(continued from previous page)

```
        '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)* –

update_disk(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.update_disk(
    Disk={
        'Description': 'string',
        'ListOfRequestTags': [
            {
                'Key': 'string',
                'Value': 'string'
            },
        ],
        'Name': 'string',
```

(continues on next page)

(continued from previous page)

```
        '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'
        },
        '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)* –

update_firewall_group(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.update_firewall_group(
    FirewallGroup={
        'Description': 'string',
        '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',
```

(continues on next page)

(continued from previous page)

```
        '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)* –

update_node_pool(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.update_node_pool(
    ClusterName='string',
    NodePool={
        'ListOfRequestTags': [
```

(continues on next page)

(continued from previous page)

```
            {
                '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,
        '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)* –

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': [
                {
```

(continues on next page)

(continued from previous page)

```
                    '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)* –

update_tags(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.update_tags(
    Tags=[
        {
```

(continues on next page)

(continued from previous page)

```
            '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

The available waiters are:

- *hatoba.Waiter.ClusterDeleted*
- *hatoba.Waiter.ClusterRunning*
- *hatoba.Waiter.FirewallRuleAuthorized*
- *hatoba.Waiter.SnapshotAvailable*

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={
        '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

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

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

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

Table of Contents

1.5.1 Client

class nas.**Client**

A low-level client representing NIFCLOUD NAS

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

These are the available methods:

- *authorize_nas_security_group_ingress()*
- *can_paginate()*
- *clear_nas_session()*
- *create_nas_instance()*
- *create_nas_security_group()*
- *delete_nas_instance()*
- *delete_nas_security_group()*
- *describe_nas_instances()*
- *describe_nas_security_groups()*
- *generate_presigned_url()*
- *get_metric_statistics()*
- *get_paginator()*
- *get_waiter()*
- *modify_nas_instance()*
- *modify_nas_security_group()*
- *revoke_nas_security_group_ingress()*
- *upgrade_nas_instance()*

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'
            },
        ],
        '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)* –

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.

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',
        'DirectoryServiceDomainName': 'string',
        'DomainControllers': [
            {
                'Hostname': 'string',
                'IPAddress': 'string'
            },
        ],
        'Endpoint': {
            'Address': 'string',
            'PrivateAddress': 'string'
        },
        'MasterUsername': 'string',
        'NASInstanceClass': 'string',
        'NASInstanceDescription': 'string',
        'NASInstanceErrorInfo': {
            'NASInstanceErrorCode': 'string',
            'NASInstanceErrorMessage': 'string'
        },
        'NASInstanceIdentifier': 'string',
        'NASInstanceStatus': 'string',
        'NASInstanceType': 123,
        'NASSecurityGroups': [
            {
```

(continues on next page)

(continued from previous page)

```
                '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)* –
    - **DirectoryServiceDomainName** *(string)* –
    - **DomainControllers** *(list)* –
      - *(dict)* –
      - **Hostname** *(string)* –
      - **IPAddress** *(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)* –

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',
    ],
    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',
        'DirectoryServiceDomainName': 'string',
        'DomainControllers': [
            {
                'Hostname': 'string',
                'IPAddress': 'string'
            },
        ],
        'Endpoint': {
            'Address': 'string',
            'PrivateAddress': 'string'
        },
        'MasterUsername': 'string',
        'NASInstanceClass': 'string',
        'NASInstanceDescription': 'string',
        'NASInstanceErrorInfo': {
            'NASInstanceErrorCode': 'string',
            'NASInstanceErrorMessage': 'string'
```

(continues on next page)

(continued from previous page)

```
        },
        '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)* –
    - **DirectoryServiceDomainName** *(string)* –
    - **DomainControllers** *(list)* –
      - *(dict)* –
      - **Hostname** *(string)* –
      - **IPAddress** *(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)* –

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)* –

delete_nas_instance(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.delete_nas_instance(
    DirectoryServiceAdministratorName='string',
    DirectoryServiceAdministratorPassword='string',
    NASInstanceIdentifier='string'
)
```

Parameters

- **DirectoryServiceAdministratorName** (*string*) –
- **DirectoryServiceAdministratorPassword** (*string*) –
- **NASInstanceIdentifier** (*string*) – **[REQUIRED]**

Return type dict

Returns

Response Syntax

```
{
    'NASInstance': {
        'AllocatedStorage': 123,
        'AuthenticationType': 123,
        'AvailabilityZone': 'string',
        'DirectoryServiceDomainName': 'string',
        'DomainControllers': [
            {
                'Hostname': 'string',
                'IPAddress': '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'
```

(continues on next page)

(continued from previous page)

```
            },
        ],
        '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)* –
    - **DirectoryServiceDomainName** *(string)* –
    - **DomainControllers** *(list)* –
      - *(dict)* –
      - **Hostname** *(string)* –
      - **IPAddress** *(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)* –

delete_nas_security_group(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.delete_nas_security_group(
```

(continues on next page)

(continued from previous page)

```
    NASSecurityGroupName='string'
)
```

Parameters **NASSecurityGroupName** (*string*) – **[REQUIRED]**
Return type dict
Returns

Response Syntax

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

Response Structure

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

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',
            'DirectoryServiceDomainName': 'string',
            'DomainControllers': [
                {
                    'Hostname': 'string',
                    'IPAddress': 'string'
                },
            ],
            'Endpoint': {
                'Address': 'string',
                'PrivateAddress': 'string'
            },
            'MasterUsername': 'string',
            'NASInstanceClass': 'string',
            'NASInstanceDescription': 'string',
            'NASInstanceErrorInfo': {
                'NASInstanceErrorCode': 'string',
```

(continues on next page)

(continued from previous page)

```
                '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)* –
      - **DirectoryServiceDomainName** *(string)* –
      - **DomainControllers** *(list)* –
      - *(dict)* –
      - **Hostname** *(string)* –
      - **IPAddress** *(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)* –

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': [
                {
                    'SecurityGroupName': 'string',
                    'SecurityGroupOwnerId': 'string',
                    'Status': 'string'
                },
            ]
        },
    ],
    'ResponseMetadata': {
        'RequestId': 'string'
    }
}
```

Response Structure

- *(dict)* –
  - – **NASSecurityGroups** *(list)* –
    - ∗ *(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)* –

generate_presigned_url(*ClientMethod*, *Params=None*, *ExpiresIn=3600*, *HttpMethod=None*)

Generate a presigned url given a client, its method, and arguments

Parameters

- **ClientMethod** (*string*) – The client method to presign for
- **Params** (*dict*) – The parameters normally passed to `ClientMethod`.
- **ExpiresIn** (*int*) – The number of seconds the presigned url is valid for. By default it expires in an hour (3600 seconds)
- **HttpMethod** (*string*) – The http method to use on the generated url. By default, the http method is whatever is used in the method's model.

Returns The presigned url

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,
```

(continues on next page)

(continued from previous page)

```
            '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)* –

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.

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

modify_nas_instance(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.modify_nas_instance(
    AllocatedStorage=123,
    AuthenticationType=123,
    DirectoryServiceAdministratorName='string',
    DirectoryServiceAdministratorPassword='string',
    DirectoryServiceDomainName='string',
    DomainControllers=[
        {
            'Hostname': 'string',
            'IPAddress': 'string'
```

(continues on next page)

(continued from previous page)

```
        },
    ],
    MasterPrivateAddress='string',
    MasterUserPassword='string',
    NASInstanceDescription='string',
    NASInstanceIdentifier='string',
    NASSecurityGroups=[
        'string',
    ],
    NetworkId='string',
    NewNASInstanceIdentifier='string',
    NoRootSquash=True|False
)
```

Parameters

- **AllocatedStorage** (*integer*) –
- **AuthenticationType** (*integer*) –
- **DirectoryServiceAdministratorName** (*string*) –
- **DirectoryServiceAdministratorPassword** (*string*) –
- **DirectoryServiceDomainName** (*string*) –
- **DomainControllers** (*list*) –
  - *(dict)* –
    - **Hostname** *(string)* –
    - **IPAddress** *(string)* –
- **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,
        'AuthenticationType': 123,
        'AvailabilityZone': 'string',
        'DirectoryServiceDomainName': 'string',
        'DomainControllers': [
            {
                'Hostname': 'string',
                'IPAddress': 'string'
            },
        ],
        'Endpoint': {
            'Address': 'string',
            'PrivateAddress': 'string'
        },
        'MasterUsername': 'string',
```

(continues on next page)

(continued from previous page)

```
        '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)* –
    - **DirectoryServiceDomainName** *(string)* –
    - **DomainControllers** *(list)* –
      - *(dict)* –
      - **Hostname** *(string)* –
      - **IPAddress** *(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)* –

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)* –
    - **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)* –

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': [
            {
                '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)* –

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',
        'DirectoryServiceDomainName': 'string',
        'DomainControllers': [
            {
                'Hostname': 'string',
                'IPAddress': 'string'
            },
        ],
        'Endpoint': {
            'Address': 'string',
            'PrivateAddress': 'string'
        },
        'MasterUsername': 'string',
        'NASInstanceClass': 'string',
        'NASInstanceDescription': 'string',
        'NASInstanceErrorInfo': {
            'NASInstanceErrorCode': 'string',
            'NASInstanceErrorMessage': 'string'
        },
        'NASInstanceIdentifier': 'string',
```

(continues on next page)

(continued from previous page)

```
        '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)* –
    - **DirectoryServiceDomainName** *(string)* –
    - **DomainControllers** *(list)* –
      - *(dict)* –
      - **Hostname** *(string)* –
      - **IPAddress** *(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

The available waiters are:

- *nas.Waiter.NASInstanceAvailable*
- *nas.Waiter.NASInstanceDeleted*
- *nas.Waiter.NASInstanceExists*
- *nas.Waiter.NASInstanceFailed*
- *nas.Waiter.NASInstanceStorageFull*
- *nas.Waiter.NASSecurityGroupDeleted*
- *nas.Waiter.NASSecurityGroupExists*
- *nas.Waiter.NASSecurityGroupIPRangesAuthFailed*
- *nas.Waiter.NASSecurityGroupIPRangesAuthorized*
- *nas.Waiter.NASSecurityGroupIPRangesEmptied*
- *nas.Waiter.NASSecurityGroupIPRangesRevokeFailed*
- *nas.Waiter.NASSecurityGroupSecurityGroupsAuthFailed*
- *nas.Waiter.NASSecurityGroupSecurityGroupsAuthorized*
- *nas.Waiter.NASSecurityGroupSecurityGroupsEmptied*
- *nas.Waiter.NASSecurityGroupSecurityGroupsRevokeFailed*

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

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

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,
```

(continues on next page)

(continued from previous page)

```
        '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

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

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

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

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

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

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

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

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

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

class nas.Waiter.**NASSecurityGroupSecurityGroupsAuthorized**

```
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

- **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

class nas.Waiter.**NASSecurityGroupSecurityGroupsEmptied**

```
waiter = client.get_waiter('nas_security_group_security_groups_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={
```

(continues on next page)

(continued from previous page)

```
        '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

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

- **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

1.6 rdb

Table of Contents

1.6.1 Client

class rdb.**Client**

A low-level client representing NIFCLOUD RDB

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

These are the available methods:

- *add_source_identifier_to_subscription()*
- *authorize_db_security_group_ingress()*
- *can_paginate()*
- *copy_db_snapshot()*
- *create_db_instance()*
- *create_db_instance_read_replica()*
- *create_db_parameter_group()*
- *create_db_security_group()*
- *create_db_snapshot()*
- *create_event_subscription()*
- *delete_db_instance()*
- *delete_db_parameter_group()*
- *delete_db_security_group()*
- *delete_db_snapshot()*
- *delete_event_subscription()*
- *describe_certificates()*
- *describe_db_engine_versions()*
- *describe_db_instances()*
- *describe_db_log_files()*
- *describe_db_parameter_groups()*
- *describe_db_parameters()*
- *describe_db_security_groups()*
- *describe_db_snapshots()*
- *describe_engine_default_parameters()*
- *describe_event_categories()*
- *describe_event_subscriptions()*
- *describe_events()*
- *describe_orderable_db_instance_options()*
- *download_db_log_file_portion()*
- *generate_presigned_url()*
- *get_paginator()*
- *get_waiter()*
- *modify_db_instance()*
- *modify_db_instance_network()*
- *modify_db_parameter_group()*
- *modify_event_subscription()*
- *nifty_failover_db_instance()*
- *nifty_get_metric_statistics()*

- *reboot_db_instance()*
- *remove_source_identifier_from_subscription()*
- *reset_db_parameter_group()*
- *reset_external_master()*
- *restore_db_instance_from_db_snapshot()*
- *restore_db_instance_to_point_in_time()*
- *revoke_db_security_group_ingress()*
- *set_external_master()*
- *start_replication()*
- *stop_replication()*
- *upgrade_db_engine_version()*

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)* –

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',
        'OwnerId': 'string'
    },
    'ResponseMetadata': {
```

(continues on next page)

(continued from previous page)

```
        '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)* –

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.

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',
```

(continues on next page)

(continued from previous page)

```
        '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)* –

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-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'|
```

(continues on next page)

(continued from previous page)

```
    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',
    ]
)
```

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',
        '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',
            ]
        },
```

(continues on next page)

(continued from previous page)

```
        '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)* –

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-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',
    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',
        '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)* –

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)* –

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)* –

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)* –

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',
```

(continues on next page)

(continued from previous page)

```
        '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)* –

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',
        '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)* –

delete_db_parameter_group(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.delete_db_parameter_group(
```

(continues on next page)

(continued from previous page)

```
    DBParameterGroupName='string'
)
```

Parameters **DBParameterGroupName** (*string*) – **[REQUIRED]**
Return type dict
Returns

Response Syntax

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

Response Structure

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

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)* –

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)* –

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)* –

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',
```

(continues on next page)

(continued from previous page)

```
    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)* –

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'|'mariadb10.1',
    DefaultOnly=True|False,
    Engine='MySQL'|'postgres'|'MariaDB',
```

(continues on next page)

(continued from previous page)

```
    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',
            '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)* –

describe_db_instances(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.describe_db_instances(
    DBInstanceIdentifier='string',
    Filter='string',
    FilterName='string',
    FilterValue='string',
    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',
```

(continues on next page)

(continued from previous page)

```
            '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',
                'MultiAZ': True|False,
```

(continues on next page)

(continued from previous page)

```
                '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)* –
  - **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)* –

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)* –

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=[
```

(continues on next page)

(continued from previous page)

```
        '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)* –

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)* –

describe_db_security_groups(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.describe_db_security_groups(
    DBSecurityGroupName='string',
    Filter='string',
```

(continues on next page)

(continued from previous page)

```
    FilterName='string',
    FilterValue='string',
    Filters=[
        '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)* –

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,
            'AvailabilityZone': 'string',
            'DBInstanceIdentifier': 'string',
```

(continues on next page)

(continued from previous page)

```
            '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)* –

describe_engine_default_parameters(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.describe_engine_default_parameters(
    DBParameterGroupFamily='string',
    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)* –

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)* –

describe_event_subscriptions (***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.describe_event_subscriptions(
    Marker='string',
    MaxRecords=123,
    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)* –

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)* –

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',
    Engine='MySQL'|'postgres'|'MariaDB',
    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': [
```

(continues on next page)

(continued from previous page)

```
                        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)* –

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)* –

generate_presigned_url (*ClientMethod*, *Params=None*, *ExpiresIn=3600*, *HttpMethod=None*)

Generate a presigned url given a client, its method, and arguments

Parameters

- **ClientMethod** (*string*) – The client method to presign for
- **Params** (*dict*) – The parameters normally passed to `ClientMethod`.
- **ExpiresIn** (*int*) – The number of seconds the presigned url is valid for. By default it expires in an hour (3600 seconds)
- **HttpMethod** (*string*) – The http method to use on the generated url. By default, the http method is whatever is used in the method's model.

Returns The presigned url

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.

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

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-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',
    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': {
        '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': [
```

(continues on next page)

(continued from previous page)

```
                '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)* –

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',
```

(continues on next page)

(continued from previous page)

```
        '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',
            'MultiAZ': True|False,
```

(continues on next page)

(continued from previous page)

```
            '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)* –

modify_db_parameter_group(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.modify_db_parameter_group(
    DBParameterGroupName='string',
    Parameters=[
```

(continues on next page)

(continued from previous page)

```
        {
            '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)* –

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)* –

nifty_failover_db_instance (***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.nifty_failover_db_instance(
    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',
```

(continues on next page)

(continued from previous page)

```
        '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)* –

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': {
```

(continues on next page)

(continued from previous page)

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

Response Structure

- *(dict)* –
  - **Datapoints** *(list)* –
    - *(dict)* –
      - **NiftyTargetName** *(string)* –
      - **SampleCount** *(integer)* –
      - **Sum** *(float)* –
      - **Timestamp** *(datetime)* –
  - **Label** *(string)* –
  - **ResponseMetadata** *(dict)* –
    - **RequestId** *(string)* –

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': [
```

(continues on next page)

(continued from previous page)

```
            {
                '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',
        ],
        'ReadReplicaSourceDBInstanceIdentifier': 'string',
```

(continues on next page)

(continued from previous page)

```
        '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)* –

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',
        ],
        '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)* –

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)* –

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)* –

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',
    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.
```

(continues on next page)

(continued from previous page)

```
    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': {
```

(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)* –

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-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',
    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',
    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': [
            {
                'DBParameterGroupName': 'string',
                'ParameterApplyStatus': 'string'
            },
        ],
        'DBSecurityGroups': [
            {
                'DBSecurityGroupName': 'string',
                'Status': 'string'
            },
        ],
        'Endpoint': {
            'Address': 'string',
            'NiftyPrivateAddress': 'string',
            'Port': 123
        },
        'Engine': 'string',
        'EngineVersion': 'string',
        'ExternalReplicationInfo': {
```

(continues on next page)

(continued from previous page)

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

(continues on next page)

(continued from previous page)

```
    }
}
```

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)* –

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': [
```

(continues on next page)

(continued from previous page)

```
            {
                '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)* –

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'
            },
        ],
        '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)* –

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
        },
```

(continues on next page)

(continued from previous page)

```
        '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'
            },
        ],
        'VpcSecurityGroups': 'string'
```

(continues on next page)

(continued from previous page)

```
    },
    '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)* –

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',
```

(continues on next page)

(continued from previous page)

```
        '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',
            'MultiAZ': True|False,
```

(continues on next page)

(continued from previous page)

```
            '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)* –

upgrade_db_engine_version(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.upgrade_db_engine_version(
    AllowMajorVersionUpgrade=True|False,
    DBInstanceIdentifier='string',
```

(continues on next page)

(continued from previous page)

```
    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',
```

(continues on next page)

(continued from previous page)

```
            ],
            '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': {
        '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

The available waiters are:

- *rdb.Waiter.DBInstanceAvailable*
- *rdb.Waiter.DBInstanceDeleted*
- *rdb.Waiter.DBInstanceExists*
- *rdb.Waiter.DBInstanceFailed*
- *rdb.Waiter.DBInstanceStorageFull*
- *rdb.Waiter.DBSecurityGroupDeleted*
- *rdb.Waiter.DBSecurityGroupEC2SecurityGroupsAuthFailed*
- *rdb.Waiter.DBSecurityGroupEC2SecurityGroupsAuthorized*
- *rdb.Waiter.DBSecurityGroupEC2SecurityGroupsEmptied*
- *rdb.Waiter.DBSecurityGroupEC2SecurityGroupsRevokeFailed*
- *rdb.Waiter.DBSecurityGroupExists*
- *rdb.Waiter.DBSecurityGroupIPRangesAuthFailed*
- *rdb.Waiter.DBSecurityGroupIPRangesAuthorized*

- *rdb.Waiter.DBSecurityGroupIPRangesEmptied*
- *rdb.Waiter.DBSecurityGroupIPRangesRevokeFailed*

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

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

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',
```

(continues on next page)

(continued from previous page)

```
    ],
    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

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

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

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

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

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

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',
    FilterValue='string',
    Filters=[
        'string',
    ],
    Marker='string',
    MaxRecords=123,
```

(continues on next page)

(continued from previous page)

```
    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

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',
    ],
    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

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,
        '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

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

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

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',
```

(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

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
```

(continues on next page)

(continued from previous page)

```
    }
)
```

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

Table of Contents

1.7.1 Client

class `script.`**Client**

A low-level client representing NIFCLOUD Script

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

These are the available methods:

- *can_paginate()*
- *execute_script()*
- *generate_presigned_url()*
- *get_paginator()*
- *get_waiter()*

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.

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)* –

generate_presigned_url(*ClientMethod*, *Params=None*, *ExpiresIn=3600*, *HttpMethod=None*)

Generate a presigned url given a client, its method, and arguments

Parameters

- **ClientMethod** (*string*) – The client method to presign for
- **Params** (*dict*) – The parameters normally passed to `ClientMethod`.
- **ExpiresIn** (*int*) – The number of seconds the presigned url is valid for. By default it expires in an hour (3600 seconds)

- **HttpMethod** (*string*) – The http method to use on the generated url. By default, the http method is whatever is used in the method's model.

Returns The presigned url

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.

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

Table of Contents

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:

- *can_paginate()*
- *describe_event_attributes()*
- *describe_event_calendar()*

- *describe_service_statuses()*
- *generate_presigned_url()*
- *get_paginator()*
- *get_waiter()*

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.

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': [
            {
                'AffectedService': [
                    {
                        'EndAt': 'string',
                        'Influence': 'string',
                        'Location': 'string',
                        'Menu': 'string',
                        'Number': 123,
                        'Resource': [
                            {
                                'DiskName': 'string',
                                'ResourceName': 'string',
                                'ResourceType': 'string'
                            },
                        ],
                        'Service': 'string',
                        'StartAt': 'string',
                        'Status': 'string'
                    },
                ],
                'EndAt': 'string',
                'EventHistory': [
```

(continues on next page)

(continued from previous page)

```
                    {
                        '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)* –
      - **AffectedService** *(list)* –
      - *(dict)* –
      - **EndAt** *(string)* –
      - **Influence** *(string)* –
      - **Location** *(string)* –
      - **Menu** *(string)* –
      - **Number** *(integer)* –
      - **Resource** *(list)* –
      - *(dict)* –
      - **DiskName** *(string)* –
      - **ResourceName** *(string)* –
      - **ResourceType** *(string)* –
      - **Service** *(string)* –
      - **StartAt** *(string)* –
      - **Status** *(string)* –
      - **EndAt** *(string)* –
      - **EventHistory** *(list)* –
      - *(dict)* –
      - **Date** *(string)* –
      - **Message** *(string)* –
      - **EventID** *(string)* –
      - **EventStatus** *(string)* –
      - **StartAt** *(string)* –
    - **Mode** *(string)* –
    - **TargetDate** *(string)* –
  - **Datetime** *(string)* –
  - **RequestID** *(string)* –

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',
                '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)* –

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': {
        '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)* –

generate_presigned_url (*ClientMethod*, *Params=None*, *ExpiresIn=3600*, *HttpMethod=None*)

Generate a presigned url given a client, its method, and arguments

Parameters

- **ClientMethod** (*string*) – The client method to presign for
- **Params** (*dict*) – The parameters normally passed to `ClientMethod`.
- **ExpiresIn** (*int*) – The number of seconds the presigned url is valid for. By default it expires in an hour (3600 seconds)
- **HttpMethod** (*string*) – The http method to use on the generated url. By default, the http method is whatever is used in the method's model.

Returns The presigned url

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.

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

Table of Contents

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:

- *abort_multipart_upload()*
- *can_paginate()*
- *complete_multipart_upload()*
- *delete_bucket()*
- *delete_bucket_cors()*
- *delete_bucket_lifecycle()*
- *delete_bucket_policy()*
- *delete_bucket_tagging()*
- *delete_multiple_objects()*
- *delete_object()*
- *delete_object_tagging()*
- *generate_presigned_url()*
- *get_bucket()*
- *get_bucket_acl()*
- *get_bucket_consistency()*
- *get_bucket_cors()*
- *get_bucket_lifecycle_configuration()*
- *get_bucket_object_versions()*
- *get_bucket_policy()*
- *get_bucket_tagging()*
- *get_bucket_version2()*
- *get_bucket_versioning()*
- *get_object()*
- *get_object_acl()*
- *get_object_tagging()*
- *get_paginator()*
- *get_service()*
- *get_waiter()*
- *head_bucket()*
- *head_object()*
- *initiate_multipart_upload()*
- *list_multipart_uploads()*
- *list_parts()*
- *put_bucket()*
- *put_bucket_consistency()*
- *put_bucket_cors()*
- *put_bucket_lifecycle_configuration()*
- *put_bucket_policy()*
- *put_bucket_tagging()*
- *put_bucket_versioning()*
- *put_object()*
- *put_object_copy()*
- *put_object_tagging()*
- *upload_part()*
- *upload_part_copy()*

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

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.

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)* –

delete_bucket(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.delete_bucket(
    Bucket='string'
)
```

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

Returns None

delete_bucket_cors(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.delete_bucket_cors(
    Bucket='string'
)
```

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

Returns None

delete_bucket_lifecycle(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.delete_bucket_lifecycle(
    Bucket='string'
)
```

Parameters **Bucket** (*string*) – **[REQUIRED]**
Returns None

delete_bucket_policy(***kwargs*)
See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.delete_bucket_policy(
    Bucket='string'
)
```

Parameters **Bucket** (*string*) – **[REQUIRED]**
Returns None

delete_bucket_tagging(***kwargs*)
See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.delete_bucket_tagging(
    Bucket='string'
)
```

Parameters **Bucket** (*string*) – **[REQUIRED]**
Returns None

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)* –

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)* –

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)* –

generate_presigned_url (*ClientMethod*, *Params=None*, *ExpiresIn=3600*, *HttpMethod=None*)

Generate a presigned url given a client, its method, and arguments

Parameters

- **ClientMethod** (*string*) – The client method to presign for
- **Params** (*dict*) – The parameters normally passed to `ClientMethod`.
- **ExpiresIn** (*int*) – The number of seconds the presigned url is valid for. By default it expires in an hour (3600 seconds)
- **HttpMethod** (*string*) – The http method to use on the generated url. By default, the http method is whatever is used in the method's model.

Returns The presigned url

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': [
        {
```

(continues on next page)

(continued from previous page)

```
            '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',
    '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)* –

get_bucket_acl(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.get_bucket_acl(
    Bucket='string'
```

(continues on next page)

(continued from previous page)

```
)
```

Parameters **Bucket** (*string*) – **[REQUIRED]**
Return type dict
Returns

Response Syntax

```
{
    'AccessControlList': [
        {
            'Grantee': {
                'DisplayName': 'string',
                '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)* –

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)* –

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)* –

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)* –

get_bucket_object_versions(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.get_bucket_object_versions(
    Bucket='string',
    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)* –

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)* –

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)* –

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'
        },
    ],
```

(continues on next page)

(continued from previous page)

```
    '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)* –

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)* –

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',
    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)* –

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)* –

get_object_tagging(***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.get_object_tagging(
    Bucket='string',
    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)* –

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.

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)* –

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

head_bucket (***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.head_bucket(
    Bucket='string'
)
```

Parameters **Bucket** (*string*) – **[REQUIRED]**
Returns None

head_object (***kwargs*)

See also: NIFCLOUD API Documentation

Request Syntax

```
response = client.head_object(
    Bucket='string',
```

(continues on next page)

(continued from previous page)

```
    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)* –

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',
```

(continues on next page)

(continued from previous page)

```
    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)* –

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'
            },
            '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)* –

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
        },
    ],
```

(continues on next page)

(continued from previous page)

```
    '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)* –

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)* –

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)* –

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

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

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

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

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

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)* –

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)
```

(continues on next page)

(continued from previous page)

```
}
```

Response Structure

- *(dict)* –
  - **ContentType** *(string)* –
  - **ETag** *(string)* –
  - **LastModified** *(datetime)* –

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)* –

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)* –

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'
```

(continues on next page)

(continued from previous page)

```
)
```

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',
    '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

Index

A

C

D

L

M

N

P

R

S

T

U

V

W