aws

API Reference

# Amazon Keyspaces (for Apache Cassandra)

**API Version 2022-02-10**

# Amazon Keyspaces (for Apache Cassandra): API Reference

# Table of Contents

# Welcome

Amazon Keyspaces (for Apache Cassandra) is a scalable, highly available, and managed Apache Cassandra-compatible database service. Amazon Keyspaces makes it easy to migrate, run, and scale Cassandra workloads in the AWS Cloud. With just a few clicks on the AWS Management Console or a few lines of code, you can create keyspaces and tables in Amazon Keyspaces, without deploying any infrastructure or installing software.

In addition to supporting Cassandra Query Language (CQL) requests via open-source Cassandra drivers, Amazon Keyspaces supports data definition language (DDL) operations to manage keyspaces and tables using the AWS SDK and AWS CLI, as well as infrastructure as code (IaC) services and tools such as AWS CloudFormation and Terraform. This API reference describes the supported DDL operations in detail.

For the list of all supported CQL APIs, see Supported Cassandra APIs, operations, and data types in Amazon Keyspaces in the *Amazon Keyspaces Developer Guide*.

To learn how Amazon Keyspaces API actions are recorded with AWS CloudTrail, see Amazon Keyspaces information in CloudTrail in the *Amazon Keyspaces Developer Guide*.

For more information about AWS APIs, for example how to implement retry logic or how to sign AWS API requests, see AWS APIs in the *General Reference*.

This document was last published on July 18, 2025.

# Actions

The following actions are supported:

- CreateKeyspace
- CreateTable
- CreateType
- DeleteKeyspace
- DeleteTable
- DeleteType
- GetKeyspace
- GetTable
- GetTableAutoScalingSettings
- GetType
- ListKeyspaces
- ListTables
- ListTagsForResource
- ListTypes
- RestoreTable
- TagResource
- UntagResource
- UpdateKeyspace
- UpdateTable

# CreateKeyspace

The `CreateKeyspace` operation adds a new keyspace to your account. In an AWS account, keyspace names must be unique within each Region.

`CreateKeyspace` is an asynchronous operation. You can monitor the creation status of the new keyspace by using the `GetKeyspace` operation.

For more information, see Create a keyspace in the *Amazon Keyspaces Developer Guide*.

## Request Syntax

```
{
   "keyspaceName": "string",
   "replicationSpecification": {
      "regionList": [ "string" ],
      "replicationStrategy": "string"
   },
   "tags": [
      {
         "key": "string",
         "value": "string"
      }
   ]
}
```

## Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

### keyspaceName

The name of the keyspace to be created.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

### replicationSpecification

The replication specification of the keyspace includes:

- `replicationStrategy` - the required value is `SINGLE_REGION` or `MULTI_REGION`.
- `regionList` - if the `replicationStrategy` is `MULTI_REGION`, the `regionList` requires the current Region and at least one additional AWS Region where the keyspace is going to be replicated in.

Type: ReplicationSpecification object

Required: No

### tags

A list of key-value pair tags to be attached to the keyspace.

For more information, see Adding tags and labels to Amazon Keyspaces resources in the *Amazon Keyspaces Developer Guide*.

Type: Array of Tag objects

Array Members: Minimum number of 1 item. Maximum number of 60 items.

Required: No

## Response Syntax

```
{
   "resourceArn": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### resourceArn

The unique identifier of the keyspace in the format of an Amazon Resource Name (ARN).

Type: String

Length Constraints: Minimum length of 20. Maximum length of 1000.

Pattern: `arn:(aws[a-zA-Z0-9-]*):cassandra:.+.*`

## Errors

For information about the errors that are common to all actions, see Common Errors.

AccessDeniedException

You don't have sufficient access permissions to perform this action.

HTTP Status Code: 400

ConflictException

Amazon Keyspaces couldn't complete the requested action. This error may occur if you try to perform an action and the same or a different action is already in progress, or if you try to create a resource that already exists.

HTTP Status Code: 400

InternalServerException

Amazon Keyspaces was unable to fully process this request because of an internal server error.

HTTP Status Code: 500

ServiceQuotaExceededException

The operation exceeded the service quota for this resource. For more information on service quotas, see Quotas in the *Amazon Keyspaces Developer Guide*.

HTTP Status Code: 400

ValidationException

The operation failed due to an invalid or malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# CreateTable

The `CreateTable` operation adds a new table to the specified keyspace. Within a keyspace, table names must be unique.

`CreateTable` is an asynchronous operation. When the request is received, the status of the table is set to `CREATING`. You can monitor the creation status of the new table by using the `GetTable` operation, which returns the current `status` of the table. You can start using a table when the status is `ACTIVE`.

For more information, see Create a table in the *Amazon Keyspaces Developer Guide*.

## Request Syntax

```
{
   "autoScalingSpecification": {
      "readCapacityAutoScaling": {
         "autoScalingDisabled": boolean,
         "maximumUnits": number,
         "minimumUnits": number,
         "scalingPolicy": {
            "targetTrackingScalingPolicyConfiguration": {
               "disableScaleIn": boolean,
               "scaleInCooldown": number,
               "scaleOutCooldown": number,
               "targetValue": number
            }
         }
      },
      "writeCapacityAutoScaling": {
         "autoScalingDisabled": boolean,
         "maximumUnits": number,
         "minimumUnits": number,
         "scalingPolicy": {
            "targetTrackingScalingPolicyConfiguration": {
               "disableScaleIn": boolean,
               "scaleInCooldown": number,
               "scaleOutCooldown": number,
               "targetValue": number
            }
         }
      }
```

```
    },
    "capacitySpecification": {
       "readCapacityUnits": number,
       "throughputMode": "string",
       "writeCapacityUnits": number
    },
    "cdcSpecification": {
       "propagateTags": "string",
       "status": "string",
       "tags": [
          {
             "key": "string",
             "value": "string"
          }
       ],
       "viewType": "string"
    },
    "clientSideTimestamps": {
       "status": "string"
    },
    "comment": {
       "message": "string"
    },
    "defaultTimeToLive": number,
    "encryptionSpecification": {
       "kmsKeyIdentifier": "string",
       "type": "string"
    },
    "keyspaceName": "string",
    "pointInTimeRecovery": {
       "status": "string"
    },
    "replicaSpecifications": [
       {
          "readCapacityAutoScaling": {
             "autoScalingDisabled": boolean,
             "maximumUnits": number,
             "minimumUnits": number,
             "scalingPolicy": {
                "targetTrackingScalingPolicyConfiguration": {
                   "disableScaleIn": boolean,
                   "scaleInCooldown": number,
                   "scaleOutCooldown": number,
                   "targetValue": number
```

```
                }
             }
          },
          "readCapacityUnits": number,
          "region": "string"
       }
    ],
    "schemaDefinition": {
       "allColumns": [
          {
             "name": "string",
             "type": "string"
          }
       ],
       "clusteringKeys": [
          {
             "name": "string",
             "orderBy": "string"
          }
       ],
       "partitionKeys": [
          {
             "name": "string"
          }
       ],
       "staticColumns": [
          {
             "name": "string"
          }
       ]
    },
    "tableName": "string",
    "tags": [
       {
          "key": "string",
          "value": "string"
       }
    ],
    "ttl": {
       "status": "string"
    }
}
```

## Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

**autoScalingSpecification**

The optional auto scaling settings for a table in provisioned capacity mode. Specifies if the service can manage throughput capacity automatically on your behalf.

Auto scaling helps you provision throughput capacity for variable workloads efficiently by increasing and decreasing your table's read and write capacity automatically in response to application traffic. For more information, see Managing throughput capacity automatically with Amazon Keyspaces auto scaling in the *Amazon Keyspaces Developer Guide*.

By default, auto scaling is disabled for a table.

Type: AutoScalingSpecification object

Required: No

**capacitySpecification**

Specifies the read/write throughput capacity mode for the table. The options are:

- `throughputMode:PAY_PER_REQUEST` and
- `throughputMode:PROVISIONED` - Provisioned capacity mode requires `readCapacityUnits` and `writeCapacityUnits` as input.

The default is `throughput_mode:PAY_PER_REQUEST`.

For more information, see Read/write capacity modes in the *Amazon Keyspaces Developer Guide*.

Type: CapacitySpecification object

Required: No

**cdcSpecification**

The CDC stream settings of the table.

Type: CdcSpecification object

Required: No

**clientSideTimestamps**

Enables client-side timestamps for the table. By default, the setting is disabled. You can enable client-side timestamps with the following option:

- `status: "enabled"`

Once client-side timestamps are enabled for a table, this setting cannot be disabled.

Type: ClientSideTimestamps object

Required: No

**comment**

This parameter allows to enter a description of the table.

Type: Comment object

Required: No

**defaultTimeToLive**

The default Time to Live setting in seconds for the table.

For more information, see Setting the default TTL value for a table in the *Amazon Keyspaces Developer Guide*.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 630720000.

Required: No

**encryptionSpecification**

Specifies how the encryption key for encryption at rest is managed for the table. You can choose one of the following KMS key (KMS key):

- `type:AWS_OWNED_KMS_KEY` - This key is owned by Amazon Keyspaces.
- `type:CUSTOMER_MANAGED_KMS_KEY` - This key is stored in your account and is created, owned, and managed by you. This option requires the `kms_key_identifier` of the KMS key in Amazon Resource Name (ARN) format as input.

The default is `type:AWS_OWNED_KMS_KEY`.

For more information, see Encryption at rest in the *Amazon Keyspaces Developer Guide*.

Type: EncryptionSpecification object

Required: No

**keyspaceName**

The name of the keyspace that the table is going to be created in.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

**pointInTimeRecovery**

Specifies if `pointInTimeRecovery` is enabled or disabled for the table. The options are:

- `status=ENABLED`
- `status=DISABLED`

If it's not specified, the default is `status=DISABLED`.

For more information, see Point-in-time recovery in the *Amazon Keyspaces Developer Guide*.

Type: PointInTimeRecovery object

Required: No

**replicaSpecifications**

The optional AWS Region specific settings of a multi-Region table. These settings overwrite the general settings of the table for the specified Region.

For a multi-Region table in provisioned capacity mode, you can configure the table's read capacity differently for each Region's replica. The write capacity, however, remains synchronized between all replicas to ensure that there's enough capacity to replicate writes across all Regions. To define the read capacity for a table replica in a specific Region, you can do so by configuring the following parameters.

- `region`: The Region where these settings are applied. (Required)
- `readCapacityUnits`: The provisioned read capacity units. (Optional)
- `readCapacityAutoScaling`: The read capacity auto scaling settings for the table. (Optional)

Type: Array of ReplicaSpecification objects

Array Members: Minimum number of 1 item.

Required: No

**schemaDefinition**

The `schemaDefinition` consists of the following parameters.

For each column to be created:

- `name` - The name of the column.
- `type` - An Amazon Keyspaces data type. For more information, see Data types in the *Amazon Keyspaces Developer Guide*.

The primary key of the table consists of the following columns:

- `partitionKeys` - The partition key can be a single column, or it can be a compound value composed of two or more columns. The partition key portion of the primary key is required and determines how Amazon Keyspaces stores your data.
- `name` - The name of each partition key column.
- `clusteringKeys` - The optional clustering column portion of your primary key determines how the data is clustered and sorted within each partition.
- `name` - The name of the clustering column.
- `orderBy` - Sets the ascendant (`ASC`) or descendant (`DESC`) order modifier.

To define a column as static use `staticColumns` - Static columns store values that are shared by all rows in the same partition:

- `name` - The name of the column.
- `type` - An Amazon Keyspaces data type.

Type: SchemaDefinition object

Required: Yes

**tableName**

The name of the table.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

**tags**

A list of key-value pair tags to be attached to the resource.

For more information, see Adding tags and labels to Amazon Keyspaces resources in the *Amazon Keyspaces Developer Guide*.

Type: Array of Tag objects

Array Members: Minimum number of 1 item. Maximum number of 60 items.

Required: No

**ttl**

Enables Time to Live custom settings for the table. The options are:

- `status:enabled`
- `status:disabled`

The default is `status:disabled`. After `ttl` is enabled, you can't disable it for the table.

For more information, see Expiring data by using Amazon Keyspaces Time to Live (TTL) in the *Amazon Keyspaces Developer Guide*.

Type: TimeToLive object

Required: No

## Response Syntax

```
{
```

```
   "resourceArn": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**resourceArn**

The unique identifier of the table in the format of an Amazon Resource Name (ARN).

Type: String

Length Constraints: Minimum length of 20. Maximum length of 1000.

Pattern: `arn:(aws[a-zA-Z0-9-]*):cassandra:.+.*`

## Errors

For information about the errors that are common to all actions, see Common Errors.

AccessDeniedException

You don't have sufficient access permissions to perform this action.

HTTP Status Code: 400

ConflictException

Amazon Keyspaces couldn't complete the requested action. This error may occur if you try to perform an action and the same or a different action is already in progress, or if you try to create a resource that already exists.

HTTP Status Code: 400

InternalServerException

Amazon Keyspaces was unable to fully process this request because of an internal server error.

HTTP Status Code: 500

ResourceNotFoundException

The operation tried to access a keyspace, table, or type that doesn't exist. The resource might not be specified correctly, or its status might not be `ACTIVE`.

HTTP Status Code: 400

ServiceQuotaExceededException

The operation exceeded the service quota for this resource. For more information on service quotas, see Quotas in the *Amazon Keyspaces Developer Guide*.

HTTP Status Code: 400

ValidationException

The operation failed due to an invalid or malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# CreateType

The `CreateType` operation creates a new user-defined type in the specified keyspace.

To configure the required permissions, see Permissions to create a UDT in the *Amazon Keyspaces Developer Guide*.

For more information, see User-defined types (UDTs) in the *Amazon Keyspaces Developer Guide*.

## Request Syntax

```
{
   "fieldDefinitions": [
      {
         "name": "string",
         "type": "string"
      }
   ],
   "keyspaceName": "string",
   "typeName": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

**fieldDefinitions**

The field definitions, consisting of names and types, that define this type.

Type: Array of FieldDefinition objects

Array Members: Minimum number of 1 item.

Required: Yes

**keyspaceName**

The name of the keyspace.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

**typeName**

The name of the user-defined type.

UDT names must contain 48 characters or less, must begin with an alphabetic character, and can only contain alpha-numeric characters and underscores. Amazon Keyspaces converts upper case characters automatically into lower case characters.

Alternatively, you can declare a UDT name in double quotes. When declaring a UDT name inside double quotes, Amazon Keyspaces preserves upper casing and allows special characters.

You can also use double quotes as part of the name when you create the UDT, but you must escape each double quote character with an additional double quote character.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Required: Yes

## Response Syntax

```
{
   "keyspaceArn": "string",
   "typeName": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**keyspaceArn**

The unique identifier of the keyspace that contains the new type in the format of an Amazon Resource Name (ARN).

Type: String

Length Constraints: Minimum length of 20. Maximum length of 1000.

Pattern: `arn:(aws[a-zA-Z0-9-]*):cassandra:.+.*`

**typeName**

The formatted name of the user-defined type that was created. Note that Amazon Keyspaces requires the formatted name of the type for other operations, for example `GetType`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

## Errors

For information about the errors that are common to all actions, see Common Errors.

AccessDeniedException

You don't have sufficient access permissions to perform this action.

HTTP Status Code: 400

ConflictException

Amazon Keyspaces couldn't complete the requested action. This error may occur if you try to perform an action and the same or a different action is already in progress, or if you try to create a resource that already exists.

HTTP Status Code: 400

InternalServerException

Amazon Keyspaces was unable to fully process this request because of an internal server error.

HTTP Status Code: 500

ResourceNotFoundException

The operation tried to access a keyspace, table, or type that doesn't exist. The resource might not be specified correctly, or its status might not be `ACTIVE`.

HTTP Status Code: 400

ServiceQuotaExceededException

The operation exceeded the service quota for this resource. For more information on service quotas, see Quotas in the *Amazon Keyspaces Developer Guide*.

HTTP Status Code: 400

ValidationException

The operation failed due to an invalid or malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# DeleteKeyspace

The `DeleteKeyspace` operation deletes a keyspace and all of its tables.

## Request Syntax

```
{
   "keyspaceName": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

**keyspaceName**

The name of the keyspace to be deleted.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

## Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see Common Errors.

AccessDeniedException

You don't have sufficient access permissions to perform this action.

HTTP Status Code: 400

ConflictException

Amazon Keyspaces couldn't complete the requested action. This error may occur if you try to perform an action and the same or a different action is already in progress, or if you try to create a resource that already exists.

HTTP Status Code: 400

InternalServerException

Amazon Keyspaces was unable to fully process this request because of an internal server error.

HTTP Status Code: 500

ResourceNotFoundException

The operation tried to access a keyspace, table, or type that doesn't exist. The resource might not be specified correctly, or its status might not be `ACTIVE`.

HTTP Status Code: 400

ServiceQuotaExceededException

The operation exceeded the service quota for this resource. For more information on service quotas, see Quotas in the *Amazon Keyspaces Developer Guide*.

HTTP Status Code: 400

ValidationException

The operation failed due to an invalid or malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2

- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# DeleteTable

The `DeleteTable` operation deletes a table and all of its data. After a `DeleteTable` request is received, the specified table is in the `DELETING` state until Amazon Keyspaces completes the deletion. If the table is in the `ACTIVE` state, you can delete it. If a table is either in the `CREATING` or `UPDATING` states, then Amazon Keyspaces returns a `ResourceInUseException`. If the specified table does not exist, Amazon Keyspaces returns a `ResourceNotFoundException`. If the table is already in the `DELETING` state, no error is returned.

## Request Syntax

```
{
   "keyspaceName": "string",
   "tableName": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

**keyspaceName**

The name of the keyspace of the to be deleted table.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

**tableName**

The name of the table to be deleted.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

## Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see Common Errors.

AccessDeniedException

You don't have sufficient access permissions to perform this action.

HTTP Status Code: 400

ConflictException

Amazon Keyspaces couldn't complete the requested action. This error may occur if you try to perform an action and the same or a different action is already in progress, or if you try to create a resource that already exists.

HTTP Status Code: 400

InternalServerException

Amazon Keyspaces was unable to fully process this request because of an internal server error.

HTTP Status Code: 500

ResourceNotFoundException

The operation tried to access a keyspace, table, or type that doesn't exist. The resource might not be specified correctly, or its status might not be `ACTIVE`.

HTTP Status Code: 400

ServiceQuotaExceededException

The operation exceeded the service quota for this resource. For more information on service quotas, see Quotas in the *Amazon Keyspaces Developer Guide*.

HTTP Status Code: 400

ValidationException

The operation failed due to an invalid or malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# DeleteType

The `DeleteType` operation deletes a user-defined type (UDT). You can only delete a type that is not used in a table or another UDT.

To configure the required permissions, see Permissions to delete a UDT in the *Amazon Keyspaces Developer Guide*.

## Request Syntax

```
{
   "keyspaceName": "string",
   "typeName": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

**keyspaceName**

The name of the keyspace of the to be deleted type.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

**typeName**

The name of the type to be deleted.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Required: Yes

## Response Syntax

```
{
   "keyspaceArn": "string",
   "typeName": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**keyspaceArn**

The unique identifier of the keyspace from which the type was deleted in the format of an Amazon Resource Name (ARN).

Type: String

Length Constraints: Minimum length of 20. Maximum length of 1000.

Pattern: `arn:(aws[a-zA-Z0-9-]*):cassandra:.+.*`

**typeName**

The name of the type that was deleted.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

## Errors

For information about the errors that are common to all actions, see Common Errors.

AccessDeniedException

You don't have sufficient access permissions to perform this action.

HTTP Status Code: 400

ConflictException

Amazon Keyspaces couldn't complete the requested action. This error may occur if you try to perform an action and the same or a different action is already in progress, or if you try to create a resource that already exists.

HTTP Status Code: 400

InternalServerException

Amazon Keyspaces was unable to fully process this request because of an internal server error.

HTTP Status Code: 500

ResourceNotFoundException

The operation tried to access a keyspace, table, or type that doesn't exist. The resource might not be specified correctly, or its status might not be `ACTIVE`.

HTTP Status Code: 400

ServiceQuotaExceededException

The operation exceeded the service quota for this resource. For more information on service quotas, see Quotas in the *Amazon Keyspaces Developer Guide*.

HTTP Status Code: 400

ValidationException

The operation failed due to an invalid or malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2

- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# GetKeyspace

Returns the name of the specified keyspace, the Amazon Resource Name (ARN), the replication strategy, the AWS Regions of a multi-Region keyspace, and the status of newly added Regions after an `UpdateKeyspace` operation.

## Request Syntax

```
{
   "keyspaceName": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

**keyspaceName**

The name of the keyspace.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

## Response Syntax

```
{
   "keyspaceName": "string",
   "replicationGroupStatuses": [
      {
         "keyspaceStatus": "string",
         "region": "string",
         "tablesReplicationProgress": "string"
      }
   ],
```

```
   "replicationRegions": [ "string" ],
   "replicationStrategy": "string",
   "resourceArn": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**keyspaceName**

The name of the keyspace.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

**replicationGroupStatuses**

A list of all Regions the keyspace is replicated in after the update keyspace operation and their status.

Type: Array of ReplicationGroupStatus objects

Array Members: Minimum number of 2 items.

**replicationRegions**

If the `replicationStrategy` of the keyspace is `MULTI_REGION`, a list of replication Regions is returned.

Type: Array of strings

Array Members: Minimum number of 2 items.

Length Constraints: Minimum length of 2. Maximum length of 25.

**replicationStrategy**

Returns the replication strategy of the keyspace. The options are `SINGLE_REGION` or `MULTI_REGION`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 20.

Valid Values: `SINGLE_REGION | MULTI_REGION`

**resourceArn**

Returns the ARN of the keyspace.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 1000.

Pattern: `arn:(aws[a-zA-Z0-9-]*):cassandra:.+.*`

## Errors

For information about the errors that are common to all actions, see Common Errors.

AccessDeniedException

You don't have sufficient access permissions to perform this action.

HTTP Status Code: 400

InternalServerException

Amazon Keyspaces was unable to fully process this request because of an internal server error.

HTTP Status Code: 500

ResourceNotFoundException

The operation tried to access a keyspace, table, or type that doesn't exist. The resource might not be specified correctly, or its status might not be `ACTIVE`.

HTTP Status Code: 400

ServiceQuotaExceededException

The operation exceeded the service quota for this resource. For more information on service quotas, see Quotas in the *Amazon Keyspaces Developer Guide*.

HTTP Status Code: 400

ValidationException

The operation failed due to an invalid or malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# GetTable

Returns information about the table, including the table's name and current status, the keyspace name, configuration settings, and metadata.

To read table metadata using `GetTable`, the IAM principal needs `Select` action permissions for the table and the system keyspace.

## Request Syntax

```
{
   "keyspaceName": "string",
   "tableName": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

**keyspaceName**

The name of the keyspace that the table is stored in.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

**tableName**

The name of the table.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

## Response Syntax

```
{
   "capacitySpecification": {
      "lastUpdateToPayPerRequestTimestamp": number,
      "readCapacityUnits": number,
      "throughputMode": "string",
      "writeCapacityUnits": number
   },
   "cdcSpecification": {
      "status": "string",
      "viewType": "string"
   },
   "clientSideTimestamps": {
      "status": "string"
   },
   "comment": {
      "message": "string"
   },
   "creationTimestamp": number,
   "defaultTimeToLive": number,
   "encryptionSpecification": {
      "kmsKeyIdentifier": "string",
      "type": "string"
   },
   "keyspaceName": "string",
   "latestStreamArn": "string",
   "pointInTimeRecovery": {
      "earliestRestorableTimestamp": number,
      "status": "string"
   },
   "replicaSpecifications": [
      {
         "capacitySpecification": {
            "lastUpdateToPayPerRequestTimestamp": number,
            "readCapacityUnits": number,
            "throughputMode": "string",
            "writeCapacityUnits": number
         },
         "region": "string",
         "status": "string"
      }
   ],
```

```
   "resourceArn": "string",
   "schemaDefinition": {
      "allColumns": [
         {
            "name": "string",
            "type": "string"
         }
      ],
      "clusteringKeys": [
         {
            "name": "string",
            "orderBy": "string"
         }
      ],
      "partitionKeys": [
         {
            "name": "string"
         }
      ],
      "staticColumns": [
         {
            "name": "string"
         }
      ]
   },
   "status": "string",
   "tableName": "string",
   "ttl": {
      "status": "string"
   }
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### capacitySpecification

The read/write throughput capacity mode for a table. The options are:

- `throughputMode:PAY_PER_REQUEST`
- `throughputMode:PROVISIONED`

Type: CapacitySpecificationSummary object

**cdcSpecification**

The CDC stream settings of the table.

Type: CdcSpecificationSummary object

**clientSideTimestamps**

The client-side timestamps setting of the table.

Type: ClientSideTimestamps object

**comment**

The the description of the specified table.

Type: Comment object

**creationTimestamp**

The creation timestamp of the specified table.

Type: Timestamp

**defaultTimeToLive**

The default Time to Live settings in seconds of the specified table.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 630720000.

**encryptionSpecification**

The encryption settings of the specified table.

Type: EncryptionSpecification object

**keyspaceName**

The name of the keyspace that the specified table is stored in.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

**latestStreamArn**

The Amazon Resource Name (ARN) of the stream.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `arn:(aws[a-zA-Z0-9-]*):cassandra:.+.*`

**pointInTimeRecovery**

The point-in-time recovery status of the specified table.

Type: PointInTimeRecoverySummary object

**replicaSpecifications**

Returns the AWS Region specific settings of all Regions a multi-Region table is replicated in.

Type: Array of ReplicaSpecificationSummary objects

Array Members: Minimum number of 0 items.

**resourceArn**

The Amazon Resource Name (ARN) of the specified table.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 1000.

Pattern: `arn:(aws[a-zA-Z0-9-]*):cassandra:.+.*`

**schemaDefinition**

The schema definition of the specified table.

Type: SchemaDefinition object

**status**

The current status of the specified table.

Type: String

Valid Values: `ACTIVE | CREATING | UPDATING | DELETING | DELETED | RESTORING | INACCESSIBLE_ENCRYPTION_CREDENTIALS`

**tableName**

The name of the specified table.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

**ttl**

The custom Time to Live settings of the specified table.

Type: TimeToLive object

## Errors

For information about the errors that are common to all actions, see Common Errors.

AccessDeniedException

You don't have sufficient access permissions to perform this action.

HTTP Status Code: 400

InternalServerException

Amazon Keyspaces was unable to fully process this request because of an internal server error.

HTTP Status Code: 500

ResourceNotFoundException

The operation tried to access a keyspace, table, or type that doesn't exist. The resource might not be specified correctly, or its status might not be `ACTIVE`.

HTTP Status Code: 400

ServiceQuotaExceededException

The operation exceeded the service quota for this resource. For more information on service quotas, see Quotas in the *Amazon Keyspaces Developer Guide*.

HTTP Status Code: 400

ValidationException

The operation failed due to an invalid or malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# GetTableAutoScalingSettings

Returns auto scaling related settings of the specified table in JSON format. If the table is a multi-Region table, the AWS Region specific auto scaling settings of the table are included.

Amazon Keyspaces auto scaling helps you provision throughput capacity for variable workloads efficiently by increasing and decreasing your table's read and write capacity automatically in response to application traffic. For more information, see Managing throughput capacity automatically with Amazon Keyspaces auto scaling in the *Amazon Keyspaces Developer Guide*.

> **Important**
>
> `GetTableAutoScalingSettings` can't be used as an action in an IAM policy.

To define permissions for `GetTableAutoScalingSettings`, you must allow the following two actions in the IAM policy statement's `Action` element:

- `application-autoscaling:DescribeScalableTargets`
- `application-autoscaling:DescribeScalingPolicies`

## Request Syntax

```
{
   "keyspaceName": "string",
   "tableName": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

**keyspaceName**

The name of the keyspace.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

**tableName**

The name of the table.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

## Response Syntax

```
{
   "autoScalingSpecification": {
      "readCapacityAutoScaling": {
         "autoScalingDisabled": boolean,
         "maximumUnits": number,
         "minimumUnits": number,
         "scalingPolicy": {
            "targetTrackingScalingPolicyConfiguration": {
               "disableScaleIn": boolean,
               "scaleInCooldown": number,
               "scaleOutCooldown": number,
               "targetValue": number
            }
         }
      },
      "writeCapacityAutoScaling": {
         "autoScalingDisabled": boolean,
         "maximumUnits": number,
         "minimumUnits": number,
         "scalingPolicy": {
            "targetTrackingScalingPolicyConfiguration": {
               "disableScaleIn": boolean,
               "scaleInCooldown": number,
               "scaleOutCooldown": number,
```

```
                "targetValue": number
            }
         }
      }
   },
   "keyspaceName": "string",
   "replicaSpecifications": [
      {
         "autoScalingSpecification": {
            "readCapacityAutoScaling": {
               "autoScalingDisabled": boolean,
               "maximumUnits": number,
               "minimumUnits": number,
               "scalingPolicy": {
                  "targetTrackingScalingPolicyConfiguration": {
                     "disableScaleIn": boolean,
                     "scaleInCooldown": number,
                     "scaleOutCooldown": number,
                     "targetValue": number
                  }
               }
            },
            "writeCapacityAutoScaling": {
               "autoScalingDisabled": boolean,
               "maximumUnits": number,
               "minimumUnits": number,
               "scalingPolicy": {
                  "targetTrackingScalingPolicyConfiguration": {
                     "disableScaleIn": boolean,
                     "scaleInCooldown": number,
                     "scaleOutCooldown": number,
                     "targetValue": number
                  }
               }
            }
         },
         "region": "string"
      }
   ],
   "resourceArn": "string",
   "tableName": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**autoScalingSpecification**

The auto scaling settings of the table.

Type: AutoScalingSpecification object

**keyspaceName**

The name of the keyspace.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

**replicaSpecifications**

The AWS Region specific settings of a multi-Region table. Returns the settings for all Regions the table is replicated in.

Type: Array of ReplicaAutoScalingSpecification objects

Array Members: Minimum number of 0 items.

**resourceArn**

The Amazon Resource Name (ARN) of the table.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 1000.

Pattern: `arn:(aws[a-zA-Z0-9-]*):cassandra:.+.*`

**tableName**

The name of the table.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

## Errors

For information about the errors that are common to all actions, see Common Errors.

AccessDeniedException

You don't have sufficient access permissions to perform this action.

HTTP Status Code: 400

InternalServerException

Amazon Keyspaces was unable to fully process this request because of an internal server error.

HTTP Status Code: 500

ResourceNotFoundException

The operation tried to access a keyspace, table, or type that doesn't exist. The resource might not be specified correctly, or its status might not be `ACTIVE`.

HTTP Status Code: 400

ServiceQuotaExceededException

The operation exceeded the service quota for this resource. For more information on service quotas, see Quotas in the *Amazon Keyspaces Developer Guide*.

HTTP Status Code: 400

ValidationException

The operation failed due to an invalid or malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# GetType

The `GetType` operation returns information about the type, for example the field definitions, the timestamp when the type was last modified, the level of nesting, the status, and details about if the type is used in other types and tables.

To read keyspace metadata using `GetType`, the IAM principal needs `Select` action permissions for the system keyspace. To configure the required permissions, see Permissions to view a UDT in the *Amazon Keyspaces Developer Guide*.

## Request Syntax

```
{
   "keyspaceName": "string",
   "typeName": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

**keyspaceName**

The name of the keyspace that contains this type.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

**typeName**

The formatted name of the type. For example, if the name of the type was created without double quotes, Amazon Keyspaces saved the name in lower-case characters. If the name was created in double quotes, you must use double quotes to specify the type name.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Required: Yes

## Response Syntax

```
{
   "directParentTypes": [ "string" ],
   "directReferringTables": [ "string" ],
   "fieldDefinitions": [
      {
         "name": "string",
         "type": "string"
      }
   ],
   "keyspaceArn": "string",
   "keyspaceName": "string",
   "lastModifiedTimestamp": number,
   "maxNestingDepth": number,
   "status": "string",
   "typeName": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### directParentTypes

The types that use this type.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 48.

### directReferringTables

The tables that use this type.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

**fieldDefinitions**

The names and types that define this type.

Type: Array of FieldDefinition objects

Array Members: Minimum number of 1 item.

**keyspaceArn**

The unique identifier of the keyspace that contains this type in the format of an Amazon Resource Name (ARN).

Type: String

Length Constraints: Minimum length of 20. Maximum length of 1000.

Pattern: `arn:(aws[a-zA-Z0-9-]*):cassandra:.+.*`

**keyspaceName**

The name of the keyspace that contains this type.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

**lastModifiedTimestamp**

The timestamp that shows when this type was last modified.

Type: Timestamp

**maxNestingDepth**

The level of nesting implemented for this type.

Type: Integer

**status**

The status of this type.

Type: String

Valid Values: `ACTIVE | CREATING | DELETING | RESTORING`

**typeName**

The name of the type.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

## Errors

For information about the errors that are common to all actions, see Common Errors.

AccessDeniedException

You don't have sufficient access permissions to perform this action.

HTTP Status Code: 400

InternalServerException

Amazon Keyspaces was unable to fully process this request because of an internal server error.

HTTP Status Code: 500

ResourceNotFoundException

The operation tried to access a keyspace, table, or type that doesn't exist. The resource might not be specified correctly, or its status might not be `ACTIVE`.

HTTP Status Code: 400

ServiceQuotaExceededException

The operation exceeded the service quota for this resource. For more information on service quotas, see Quotas in the *Amazon Keyspaces Developer Guide*.

HTTP Status Code: 400

ValidationException

The operation failed due to an invalid or malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# ListKeyspaces

The `ListKeyspaces` operation returns a list of keyspaces.

## Request Syntax

```
{
   "maxResults": number,
   "nextToken": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

**maxResults**

The total number of keyspaces to return in the output. If the total number of keyspaces available is more than the value specified, a `NextToken` is provided in the output. To resume pagination, provide the `NextToken` value as an argument of a subsequent API invocation.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 1000.

Required: No

**nextToken**

The pagination token. To resume pagination, provide the `NextToken` value as argument of a subsequent API invocation.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: No

## Response Syntax

```
{
   "keyspaces": [
      {
         "keyspaceName": "string",
         "replicationRegions": [ "string" ],
         "replicationStrategy": "string",
         "resourceArn": "string"
      }
   ],
   "nextToken": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**keyspaces**

A list of keyspaces.

Type: Array of KeyspaceSummary objects

**nextToken**

A token to specify where to start paginating. This is the `NextToken` from a previously truncated response.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

## Errors

For information about the errors that are common to all actions, see Common Errors.

AccessDeniedException

You don't have sufficient access permissions to perform this action.

HTTP Status Code: 400

InternalServerException

Amazon Keyspaces was unable to fully process this request because of an internal server error.

HTTP Status Code: 500

ResourceNotFoundException

The operation tried to access a keyspace, table, or type that doesn't exist. The resource might not be specified correctly, or its status might not be `ACTIVE`.

HTTP Status Code: 400

ServiceQuotaExceededException

The operation exceeded the service quota for this resource. For more information on service quotas, see Quotas in the *Amazon Keyspaces Developer Guide*.

HTTP Status Code: 400

ValidationException

The operation failed due to an invalid or malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3

- AWS SDK for Python
- AWS SDK for Ruby V3

# ListTables

The `ListTables` operation returns a list of tables for a specified keyspace.

To read keyspace metadata using `ListTables`, the IAM principal needs `Select` action permissions for the system keyspace.

## Request Syntax

```
{
   "keyspaceName": "string",
   "maxResults": number,
   "nextToken": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

**keyspaceName**

The name of the keyspace.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

**maxResults**

The total number of tables to return in the output. If the total number of tables available is more than the value specified, a `NextToken` is provided in the output. To resume pagination, provide the `NextToken` value as an argument of a subsequent API invocation.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 1000.

Required: No

**nextToken**

The pagination token. To resume pagination, provide the `NextToken` value as an argument of a subsequent API invocation.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: No

## Response Syntax

```
{
   "nextToken": "string",
   "tables": [
      {
         "keyspaceName": "string",
         "resourceArn": "string",
         "tableName": "string"
      }
   ]
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**nextToken**

A token to specify where to start paginating. This is the `NextToken` from a previously truncated response.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

**tables**

A list of tables.

Type: Array of TableSummary objects

## Errors

For information about the errors that are common to all actions, see Common Errors.

AccessDeniedException

You don't have sufficient access permissions to perform this action.

HTTP Status Code: 400

InternalServerException

Amazon Keyspaces was unable to fully process this request because of an internal server error.

HTTP Status Code: 500

ResourceNotFoundException

The operation tried to access a keyspace, table, or type that doesn't exist. The resource might not be specified correctly, or its status might not be `ACTIVE`.

HTTP Status Code: 400

ServiceQuotaExceededException

The operation exceeded the service quota for this resource. For more information on service quotas, see Quotas in the *Amazon Keyspaces Developer Guide*.

HTTP Status Code: 400

ValidationException

The operation failed due to an invalid or malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface

- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# ListTagsForResource

Returns a list of all tags associated with the specified Amazon Keyspaces resource.

To read keyspace metadata using `ListTagsForResource`, the IAM principal needs `Select` action permissions for the specified resource and the system keyspace.

## Request Syntax

```
{
   "maxResults": number,
   "nextToken": "string",
   "resourceArn": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

**maxResults**

The total number of tags to return in the output. If the total number of tags available is more than the value specified, a `NextToken` is provided in the output. To resume pagination, provide the `NextToken` value as an argument of a subsequent API invocation.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 1000.

Required: No

**nextToken**

The pagination token. To resume pagination, provide the `NextToken` value as argument of a subsequent API invocation.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: No

**resourceArn**

The Amazon Resource Name (ARN) of the Amazon Keyspaces resource.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 1000.

Pattern: `arn:(aws[a-zA-Z0-9-]*):cassandra:.+.*`

Required: Yes

## Response Syntax

```
{
   "nextToken": "string",
   "tags": [
      {
         "key": "string",
         "value": "string"
      }
   ]
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**nextToken**

A token to specify where to start paginating. This is the `NextToken` from a previously truncated response.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

**tags**

A list of tags.

Type: Array of Tag objects

Array Members: Minimum number of 1 item. Maximum number of 60 items.

## Errors

For information about the errors that are common to all actions, see Common Errors.

AccessDeniedException

You don't have sufficient access permissions to perform this action.

HTTP Status Code: 400

InternalServerException

Amazon Keyspaces was unable to fully process this request because of an internal server error.

HTTP Status Code: 500

ResourceNotFoundException

The operation tried to access a keyspace, table, or type that doesn't exist. The resource might not be specified correctly, or its status might not be `ACTIVE`.

HTTP Status Code: 400

ServiceQuotaExceededException

The operation exceeded the service quota for this resource. For more information on service quotas, see Quotas in the *Amazon Keyspaces Developer Guide*.

HTTP Status Code: 400

ValidationException

The operation failed due to an invalid or malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# ListTypes

The `ListTypes` operation returns a list of types for a specified keyspace.

To read keyspace metadata using `ListTypes`, the IAM principal needs `Select` action permissions for the system keyspace. To configure the required permissions, see Permissions to view a UDT in the *Amazon Keyspaces Developer Guide*.

## Request Syntax

```
{
   "keyspaceName": "string",
   "maxResults": number,
   "nextToken": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

**keyspaceName**

The name of the keyspace that contains the listed types.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

**maxResults**

The total number of types to return in the output. If the total number of types available is more than the value specified, a `NextToken` is provided in the output. To resume pagination, provide the `NextToken` value as an argument of a subsequent API invocation.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 1000.

Required: No

**nextToken**

The pagination token. To resume pagination, provide the `NextToken` value as an argument of a subsequent API invocation.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: No

## Response Syntax

```
{
   "nextToken": "string",
   "types": [ "string" ]
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**nextToken**

The pagination token. To resume pagination, provide the `NextToken` value as an argument of a subsequent API invocation.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

**types**

The list of types contained in the specified keyspace.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 48.

## Errors

For information about the errors that are common to all actions, see Common Errors.

AccessDeniedException

You don't have sufficient access permissions to perform this action.

HTTP Status Code: 400

InternalServerException

Amazon Keyspaces was unable to fully process this request because of an internal server error.

HTTP Status Code: 500

ResourceNotFoundException

The operation tried to access a keyspace, table, or type that doesn't exist. The resource might not be specified correctly, or its status might not be `ACTIVE`.

HTTP Status Code: 400

ServiceQuotaExceededException

The operation exceeded the service quota for this resource. For more information on service quotas, see Quotas in the *Amazon Keyspaces Developer Guide*.

HTTP Status Code: 400

ValidationException

The operation failed due to an invalid or malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface

- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# RestoreTable

Restores the table to the specified point in time within the `earliest_restorable_timestamp` and the current time. For more information about restore points, see Time window for PITR continuous backups in the *Amazon Keyspaces Developer Guide*.

Any number of users can execute up to 4 concurrent restores (any type of restore) in a given account.

When you restore using point in time recovery, Amazon Keyspaces restores your source table's schema and data to the state based on the selected timestamp (`day:hour:minute:second`) to a new table. The Time to Live (TTL) settings are also restored to the state based on the selected timestamp.

In addition to the table's schema, data, and TTL settings, `RestoreTable` restores the capacity mode, auto scaling settings, encryption settings, and point-in-time recovery settings from the source table. Unlike the table's schema data and TTL settings, which are restored based on the selected timestamp, these settings are always restored based on the table's settings as of the current time or when the table was deleted.

You can also overwrite these settings during restore:

- Read/write capacity mode
- Provisioned throughput capacity units
- Auto scaling settings
- Point-in-time (PITR) settings
- Tags

For more information, see PITR restore settings in the *Amazon Keyspaces Developer Guide*.

Note that the following settings are not restored, and you must configure them manually for the new table:

- AWS Identity and Access Management (IAM) policies
- Amazon CloudWatch metrics and alarms

## Request Syntax

```
{
   "autoScalingSpecification": {
      "readCapacityAutoScaling": {
         "autoScalingDisabled": boolean,
         "maximumUnits": number,
         "minimumUnits": number,
         "scalingPolicy": {
            "targetTrackingScalingPolicyConfiguration": {
               "disableScaleIn": boolean,
               "scaleInCooldown": number,
               "scaleOutCooldown": number,
               "targetValue": number
            }
         }
      },
      "writeCapacityAutoScaling": {
         "autoScalingDisabled": boolean,
         "maximumUnits": number,
         "minimumUnits": number,
         "scalingPolicy": {
            "targetTrackingScalingPolicyConfiguration": {
               "disableScaleIn": boolean,
               "scaleInCooldown": number,
               "scaleOutCooldown": number,
               "targetValue": number
            }
         }
      }
   },
   "capacitySpecificationOverride": {
      "readCapacityUnits": number,
      "throughputMode": "string",
      "writeCapacityUnits": number
   },
   "encryptionSpecificationOverride": {
      "kmsKeyIdentifier": "string",
      "type": "string"
   },
   "pointInTimeRecoveryOverride": {
      "status": "string"
   },
```

```
   "replicaSpecifications": [
      {
         "readCapacityAutoScaling": {
            "autoScalingDisabled": boolean,
            "maximumUnits": number,
            "minimumUnits": number,
            "scalingPolicy": {
               "targetTrackingScalingPolicyConfiguration": {
                  "disableScaleIn": boolean,
                  "scaleInCooldown": number,
                  "scaleOutCooldown": number,
                  "targetValue": number
               }
            }
         },
         "readCapacityUnits": number,
         "region": "string"
      }
   ],
   "restoreTimestamp": number,
   "sourceKeyspaceName": "string",
   "sourceTableName": "string",
   "tagsOverride": [
      {
         "key": "string",
         "value": "string"
      }
   ],
   "targetKeyspaceName": "string",
   "targetTableName": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

### autoScalingSpecification

The optional auto scaling settings for the restored table in provisioned capacity mode. Specifies if the service can manage throughput capacity of a provisioned table automatically on your behalf. Amazon Keyspaces auto scaling helps you provision throughput capacity for

variable workloads efficiently by increasing and decreasing your table's read and write capacity automatically in response to application traffic.

For more information, see Managing throughput capacity automatically with Amazon Keyspaces auto scaling in the *Amazon Keyspaces Developer Guide*.

Type: AutoScalingSpecification object

Required: No

**capacitySpecificationOverride**

Specifies the read/write throughput capacity mode for the target table. The options are:

- `throughputMode:PAY_PER_REQUEST`
- `throughputMode:PROVISIONED` - Provisioned capacity mode requires `readCapacityUnits` and `writeCapacityUnits` as input.

The default is `throughput_mode:PAY_PER_REQUEST`.

For more information, see Read/write capacity modes in the *Amazon Keyspaces Developer Guide*.

Type: CapacitySpecification object

Required: No

**encryptionSpecificationOverride**

Specifies the encryption settings for the target table. You can choose one of the following KMS key (KMS key):

- `type:AWS_OWNED_KMS_KEY` - This key is owned by Amazon Keyspaces.
- `type:CUSTOMER_MANAGED_KMS_KEY` - This key is stored in your account and is created, owned, and managed by you. This option requires the `kms_key_identifier` of the KMS key in Amazon Resource Name (ARN) format as input.

The default is `type:AWS_OWNED_KMS_KEY`.

For more information, see Encryption at rest in the *Amazon Keyspaces Developer Guide*.

Type: EncryptionSpecification object

Required: No

### pointInTimeRecoveryOverride

Specifies the `pointInTimeRecovery` settings for the target table. The options are:

- `status=ENABLED`
- `status=DISABLED`

If it's not specified, the default is `status=DISABLED`.

For more information, see Point-in-time recovery in the *Amazon Keyspaces Developer Guide*.

Type: PointInTimeRecovery object

Required: No

### replicaSpecifications

The optional Region specific settings of a multi-Regional table.

Type: Array of ReplicaSpecification objects

Array Members: Minimum number of 1 item.

Required: No

### restoreTimestamp

The restore timestamp in ISO 8601 format.

Type: Timestamp

Required: No

### sourceKeyspaceName

The keyspace name of the source table.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

**sourceTableName**

The name of the source table.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

**tagsOverride**

A list of key-value pair tags to be attached to the restored table.

For more information, see Adding tags and labels to Amazon Keyspaces resources in the *Amazon Keyspaces Developer Guide*.

Type: Array of Tag objects

Array Members: Minimum number of 1 item. Maximum number of 60 items.

Required: No

**targetKeyspaceName**

The name of the target keyspace.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

**targetTableName**

The name of the target table.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

## Response Syntax

```
{
   "restoredTableARN": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**restoredTableARN**

The Amazon Resource Name (ARN) of the restored table.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 1000.

Pattern: `arn:(aws[a-zA-Z0-9-]*):cassandra:.+.*`

## Errors

For information about the errors that are common to all actions, see Common Errors.

AccessDeniedException

You don't have sufficient access permissions to perform this action.

HTTP Status Code: 400

ConflictException

Amazon Keyspaces couldn't complete the requested action. This error may occur if you try to perform an action and the same or a different action is already in progress, or if you try to create a resource that already exists.

HTTP Status Code: 400

InternalServerException

Amazon Keyspaces was unable to fully process this request because of an internal server error.

HTTP Status Code: 500

ResourceNotFoundException

The operation tried to access a keyspace, table, or type that doesn't exist. The resource might not be specified correctly, or its status might not be `ACTIVE`.

HTTP Status Code: 400

ServiceQuotaExceededException

The operation exceeded the service quota for this resource. For more information on service quotas, see Quotas in the *Amazon Keyspaces Developer Guide*.

HTTP Status Code: 400

ValidationException

The operation failed due to an invalid or malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python

- AWS SDK for Ruby V3

# TagResource

Associates a set of tags with a Amazon Keyspaces resource. You can then activate these user-defined tags so that they appear on the Cost Management Console for cost allocation tracking. For more information, see Adding tags and labels to Amazon Keyspaces resources in the *Amazon Keyspaces Developer Guide*.

For IAM policy examples that show how to control access to Amazon Keyspaces resources based on tags, see Amazon Keyspaces resource access based on tags in the *Amazon Keyspaces Developer Guide*.

## Request Syntax

```
{
   "resourceArn": "string",
   "tags": [
      {
         "key": "string",
         "value": "string"
      }
   ]
}
```

## Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

**resourceArn**

The Amazon Resource Name (ARN) of the Amazon Keyspaces resource to which to add tags.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 1000.

Pattern: `arn:(aws[a-zA-Z0-9-]*):cassandra:.+.*`

Required: Yes

**tags**

The tags to be assigned to the Amazon Keyspaces resource.

Type: Array of Tag objects

Array Members: Minimum number of 1 item. Maximum number of 60 items.

Required: Yes

## Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see Common Errors.

AccessDeniedException

You don't have sufficient access permissions to perform this action.

HTTP Status Code: 400

ConflictException

Amazon Keyspaces couldn't complete the requested action. This error may occur if you try to perform an action and the same or a different action is already in progress, or if you try to create a resource that already exists.

HTTP Status Code: 400

InternalServerException

Amazon Keyspaces was unable to fully process this request because of an internal server error.

HTTP Status Code: 500

ResourceNotFoundException

The operation tried to access a keyspace, table, or type that doesn't exist. The resource might not be specified correctly, or its status might not be `ACTIVE`.

HTTP Status Code: 400

ServiceQuotaExceededException

The operation exceeded the service quota for this resource. For more information on service quotas, see Quotas in the *Amazon Keyspaces Developer Guide*.

HTTP Status Code: 400

ValidationException

The operation failed due to an invalid or malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# UntagResource

Removes the association of tags from a Amazon Keyspaces resource.

## Request Syntax

```
{
   "resourceArn": "string",
   "tags": [
      {
         "key": "string",
         "value": "string"
      }
   ]
}
```

## Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

**resourceArn**

The Amazon Keyspaces resource that the tags will be removed from. This value is an Amazon Resource Name (ARN).

Type: String

Length Constraints: Minimum length of 20. Maximum length of 1000.

Pattern: `arn:(aws[a-zA-Z0-9-]*):cassandra:.+.*`

Required: Yes

**tags**

A list of existing tags to be removed from the Amazon Keyspaces resource.

Type: Array of Tag objects

Array Members: Minimum number of 1 item. Maximum number of 60 items.

Required: Yes

## Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see Common Errors.

AccessDeniedException

You don't have sufficient access permissions to perform this action.

HTTP Status Code: 400

ConflictException

Amazon Keyspaces couldn't complete the requested action. This error may occur if you try to perform an action and the same or a different action is already in progress, or if you try to create a resource that already exists.

HTTP Status Code: 400

InternalServerException

Amazon Keyspaces was unable to fully process this request because of an internal server error.

HTTP Status Code: 500

ResourceNotFoundException

The operation tried to access a keyspace, table, or type that doesn't exist. The resource might not be specified correctly, or its status might not be `ACTIVE`.

HTTP Status Code: 400

ServiceQuotaExceededException

The operation exceeded the service quota for this resource. For more information on service quotas, see Quotas in the *Amazon Keyspaces Developer Guide*.

HTTP Status Code: 400

ValidationException

The operation failed due to an invalid or malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# UpdateKeyspace

Adds a new AWS Region to the keyspace. You can add a new Region to a keyspace that is either a single or a multi-Region keyspace. Amazon Keyspaces is going to replicate all tables in the keyspace to the new Region. To successfully replicate all tables to the new Region, they must use client-side timestamps for conflict resolution. To enable client-side timestamps, specify `clientSideTimestamps.status = enabled` when invoking the API. For more information about client-side timestamps, see Client-side timestamps in Amazon Keyspaces in the *Amazon Keyspaces Developer Guide*.

To add a Region to a keyspace using the `UpdateKeyspace` API, the IAM principal needs permissions for the following IAM actions:

- `cassandra:Alter`
- `cassandra:AlterMultiRegionResource`
- `cassandra:Create`
- `cassandra:CreateMultiRegionResource`
- `cassandra:Select`
- `cassandra:SelectMultiRegionResource`
- `cassandra:Modify`
- `cassandra:ModifyMultiRegionResource`

If the keyspace contains a table that is configured in provisioned mode with auto scaling enabled, the following additional IAM actions need to be allowed.

- `application-autoscaling:RegisterScalableTarget`
- `application-autoscaling:DeregisterScalableTarget`
- `application-autoscaling:DescribeScalableTargets`
- `application-autoscaling:PutScalingPolicy`
- `application-autoscaling:DescribeScalingPolicies`

To use the `UpdateKeyspace` API, the IAM principal also needs permissions to create a service-linked role with the following elements:

- `iam:CreateServiceLinkedRole` - The **action** the principal can perform.

- `arn:aws:iam::*:role/aws-service-role/replication.cassandra.amazonaws.com/AWSServiceRoleForKeyspacesReplication` - The **resource** that the action can be performed on.
- `iam:AWSServiceName: replication.cassandra.amazonaws.com` - The only AWS service that this role can be attached to is Amazon Keyspaces.

For more information, see Configure the IAM permissions required to add an AWS Region to a keyspace in the *Amazon Keyspaces Developer Guide*.

## Request Syntax

```
{
   "clientSideTimestamps": {
      "status": "string"
   },
   "keyspaceName": "string",
   "replicationSpecification": {
      "regionList": [ "string" ],
      "replicationStrategy": "string"
   }
}
```

## Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

**clientSideTimestamps**

The client-side timestamp setting of the table.

For more information, see How it works: Amazon Keyspaces client-side timestamps in the *Amazon Keyspaces Developer Guide*.

Type: ClientSideTimestamps object

Required: No

**keyspaceName**

The name of the keyspace.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

**replicationSpecification**

The replication specification of the keyspace includes:

- `regionList` - the AWS Regions where the keyspace is replicated in.
- `replicationStrategy` - the required value is `SINGLE_REGION` or `MULTI_REGION`.

Type: ReplicationSpecification object

Required: Yes

## Response Syntax

```
{
   "resourceArn": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**resourceArn**

The unique identifier of the keyspace in the format of an Amazon Resource Name (ARN).

Type: String

Length Constraints: Minimum length of 20. Maximum length of 1000.

Pattern: `arn:(aws[a-zA-Z0-9-]*):cassandra:.+.*`

# Errors

For information about the errors that are common to all actions, see Common Errors.

AccessDeniedException

You don't have sufficient access permissions to perform this action.

HTTP Status Code: 400

ConflictException

Amazon Keyspaces couldn't complete the requested action. This error may occur if you try to perform an action and the same or a different action is already in progress, or if you try to create a resource that already exists.

HTTP Status Code: 400

InternalServerException

Amazon Keyspaces was unable to fully process this request because of an internal server error.

HTTP Status Code: 500

ResourceNotFoundException

The operation tried to access a keyspace, table, or type that doesn't exist. The resource might not be specified correctly, or its status might not be `ACTIVE`.

HTTP Status Code: 400

ServiceQuotaExceededException

The operation exceeded the service quota for this resource. For more information on service quotas, see Quotas in the *Amazon Keyspaces Developer Guide*.

HTTP Status Code: 400

ValidationException

The operation failed due to an invalid or malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# UpdateTable

Adds new columns to the table or updates one of the table's settings, for example capacity mode, auto scaling, encryption, point-in-time recovery, or ttl settings. Note that you can only update one specific table setting per update operation.

## Request Syntax

```
{
   "addColumns": [
      {
         "name": "string",
         "type": "string"
      }
   ],
   "autoScalingSpecification": {
      "readCapacityAutoScaling": {
         "autoScalingDisabled": boolean,
         "maximumUnits": number,
         "minimumUnits": number,
         "scalingPolicy": {
            "targetTrackingScalingPolicyConfiguration": {
               "disableScaleIn": boolean,
               "scaleInCooldown": number,
               "scaleOutCooldown": number,
               "targetValue": number
            }
         }
      },
      "writeCapacityAutoScaling": {
         "autoScalingDisabled": boolean,
         "maximumUnits": number,
         "minimumUnits": number,
         "scalingPolicy": {
            "targetTrackingScalingPolicyConfiguration": {
               "disableScaleIn": boolean,
               "scaleInCooldown": number,
               "scaleOutCooldown": number,
               "targetValue": number
            }
         }
      }
```

```
   },
   "capacitySpecification": {
      "readCapacityUnits": number,
      "throughputMode": "string",
      "writeCapacityUnits": number
   },
   "cdcSpecification": {
      "propagateTags": "string",
      "status": "string",
      "tags": [
         {
            "key": "string",
            "value": "string"
         }
      ],
      "viewType": "string"
   },
   "clientSideTimestamps": {
      "status": "string"
   },
   "defaultTimeToLive": number,
   "encryptionSpecification": {
      "kmsKeyIdentifier": "string",
      "type": "string"
   },
   "keyspaceName": "string",
   "pointInTimeRecovery": {
      "status": "string"
   },
   "replicaSpecifications": [
      {
         "readCapacityAutoScaling": {
            "autoScalingDisabled": boolean,
            "maximumUnits": number,
            "minimumUnits": number,
            "scalingPolicy": {
               "targetTrackingScalingPolicyConfiguration": {
                  "disableScaleIn": boolean,
                  "scaleInCooldown": number,
                  "scaleOutCooldown": number,
                  "targetValue": number
               }
            }
         },
```

```
            "readCapacityUnits": number,
            "region": "string"
         }
      ],
      "tableName": "string",
      "ttl": {
         "status": "string"
      }
}
```

## Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

**addColumns**

For each column to be added to the specified table:

- `name` - The name of the column.
- `type` - An Amazon Keyspaces data type. For more information, see Data types in the *Amazon Keyspaces Developer Guide*.

Type: Array of ColumnDefinition objects

Array Members: Minimum number of 1 item.

Required: No

**autoScalingSpecification**

The optional auto scaling settings to update for a table in provisioned capacity mode. Specifies if the service can manage throughput capacity of a provisioned table automatically on your behalf. Amazon Keyspaces auto scaling helps you provision throughput capacity for variable workloads efficiently by increasing and decreasing your table's read and write capacity automatically in response to application traffic.

If auto scaling is already enabled for the table, you can use `UpdateTable` to update the minimum and maximum values or the auto scaling policy settings independently.

For more information, see Managing throughput capacity automatically with Amazon Keyspaces auto scaling in the *Amazon Keyspaces Developer Guide*.

Type: AutoScalingSpecification object

Required: No

**capacitySpecification**

Modifies the read/write throughput capacity mode for the table. The options are:

- `throughputMode:PAY_PER_REQUEST` and
- `throughputMode:PROVISIONED` - Provisioned capacity mode requires `readCapacityUnits` and `writeCapacityUnits` as input.

The default is `throughput_mode:PAY_PER_REQUEST`.

For more information, see Read/write capacity modes in the *Amazon Keyspaces Developer Guide*.

Type: CapacitySpecification object

Required: No

**cdcSpecification**

The CDC stream settings of the table.

Type: CdcSpecification object

Required: No

**clientSideTimestamps**

Enables client-side timestamps for the table. By default, the setting is disabled. You can enable client-side timestamps with the following option:

- `status: "enabled"`

Once client-side timestamps are enabled for a table, this setting cannot be disabled.

Type: ClientSideTimestamps object

Required: No

**defaultTimeToLive**

The default Time to Live setting in seconds for the table.

For more information, see Setting the default TTL value for a table in the *Amazon Keyspaces Developer Guide*.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 630720000.

Required: No

**encryptionSpecification**

Modifies the encryption settings of the table. You can choose one of the following KMS key (KMS key):

- `type:AWS_OWNED_KMS_KEY` - This key is owned by Amazon Keyspaces.
- `type:CUSTOMER_MANAGED_KMS_KEY` - This key is stored in your account and is created, owned, and managed by you. This option requires the `kms_key_identifier` of the KMS key in Amazon Resource Name (ARN) format as input.

The default is `AWS_OWNED_KMS_KEY`.

For more information, see Encryption at rest in the *Amazon Keyspaces Developer Guide*.

Type: EncryptionSpecification object

Required: No

**keyspaceName**

The name of the keyspace the specified table is stored in.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

**pointInTimeRecovery**

Modifies the `pointInTimeRecovery` settings of the table. The options are:

- `status=ENABLED`
- `status=DISABLED`

If it's not specified, the default is `status=DISABLED`.

For more information, see Point-in-time recovery in the *Amazon Keyspaces Developer Guide*.

Type: PointInTimeRecovery object

Required: No

**replicaSpecifications**

The Region specific settings of a multi-Regional table.

Type: Array of ReplicaSpecification objects

Array Members: Minimum number of 1 item.

Required: No

**tableName**

The name of the table.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

**ttl**

Modifies Time to Live custom settings for the table. The options are:

- `status:enabled`
- `status:disabled`

The default is `status:disabled`. After `ttl` is enabled, you can't disable it for the table.

For more information, see Expiring data by using Amazon Keyspaces Time to Live (TTL) in the *Amazon Keyspaces Developer Guide*.

Type: TimeToLive object

Required: No

## Response Syntax

```
{
```

```
   "resourceArn": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**resourceArn**

The Amazon Resource Name (ARN) of the modified table.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 1000.

Pattern: `arn:(aws[a-zA-Z0-9-]*):cassandra:.+.*`

## Errors

For information about the errors that are common to all actions, see Common Errors.

AccessDeniedException

You don't have sufficient access permissions to perform this action.

HTTP Status Code: 400

ConflictException

Amazon Keyspaces couldn't complete the requested action. This error may occur if you try to perform an action and the same or a different action is already in progress, or if you try to create a resource that already exists.

HTTP Status Code: 400

InternalServerException

Amazon Keyspaces was unable to fully process this request because of an internal server error.

HTTP Status Code: 500

ResourceNotFoundException

The operation tried to access a keyspace, table, or type that doesn't exist. The resource might not be specified correctly, or its status might not be `ACTIVE`.

HTTP Status Code: 400

ServiceQuotaExceededException

The operation exceeded the service quota for this resource. For more information on service quotas, see Quotas in the *Amazon Keyspaces Developer Guide*.

HTTP Status Code: 400

ValidationException

The operation failed due to an invalid or malformed request.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# Data Types

The Amazon Keyspaces API contains several data types that various actions use. This section describes each data type in detail.

> **Note**
>
> The order of each element in a data type structure is not guaranteed. Applications should not assume a particular order.

The following data types are supported:

- AutoScalingPolicy
- AutoScalingSettings
- AutoScalingSpecification
- CapacitySpecification
- CapacitySpecificationSummary
- CdcSpecification
- CdcSpecificationSummary
- ClientSideTimestamps
- ClusteringKey
- ColumnDefinition
- Comment
- EncryptionSpecification
- FieldDefinition
- KeyspaceSummary
- PartitionKey
- PointInTimeRecovery
- PointInTimeRecoverySummary
- ReplicaAutoScalingSpecification
- ReplicaSpecification
- ReplicaSpecificationSummary

- ReplicationGroupStatus
- ReplicationSpecification
- SchemaDefinition
- StaticColumn
- TableSummary
- Tag
- TargetTrackingScalingPolicyConfiguration
- TimeToLive

# AutoScalingPolicy

Amazon Keyspaces supports the `target tracking` auto scaling policy. With this policy, Amazon Keyspaces auto scaling ensures that the table's ratio of consumed to provisioned capacity stays at or near the target value that you specify. You define the target value as a percentage between 20 and 90.

## Contents

**targetTrackingScalingPolicyConfiguration**

Auto scaling scales up capacity automatically when traffic exceeds this target utilization rate, and then back down when it falls below the target. A `double` between 20 and 90.

Type: TargetTrackingScalingPolicyConfiguration object

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# AutoScalingSettings

The optional auto scaling settings for a table with provisioned throughput capacity.

To turn on auto scaling for a table in `throughputMode:PROVISIONED`, you must specify the following parameters.

Configure the minimum and maximum capacity units. The auto scaling policy ensures that capacity never goes below the minimum or above the maximum range.

- `minimumUnits`: The minimum level of throughput the table should always be ready to support. The value must be between 1 and the max throughput per second quota for your account (40,000 by default).
- `maximumUnits`: The maximum level of throughput the table should always be ready to support. The value must be between 1 and the max throughput per second quota for your account (40,000 by default).
- `scalingPolicy`: Amazon Keyspaces supports the `target tracking` scaling policy. The auto scaling target is the provisioned capacity of the table.
  - `targetTrackingScalingPolicyConfiguration`: To define the target tracking policy, you must define the target value.
    - `targetValue`: The target utilization rate of the table. Amazon Keyspaces auto scaling ensures that the ratio of consumed capacity to provisioned capacity stays at or near this value. You define `targetValue` as a percentage. A `double` between 20 and 90. (Required)
    - `disableScaleIn`: A `boolean` that specifies if `scale-in` is disabled or enabled for the table. This parameter is disabled by default. To turn on `scale-in`, set the `boolean` value to `FALSE`. This means that capacity for a table can be automatically scaled down on your behalf. (Optional)
    - `scaleInCooldown`: A cooldown period in seconds between scaling activities that lets the table stabilize before another scale in activity starts. If no value is provided, the default is 0. (Optional)
    - `scaleOutCooldown`: A cooldown period in seconds between scaling activities that lets the table stabilize before another scale out activity starts. If no value is provided, the default is 0. (Optional)

For more information, see Managing throughput capacity automatically with Amazon Keyspaces auto scaling in the *Amazon Keyspaces Developer Guide*.

## Contents

**autoScalingDisabled**

This optional parameter enables auto scaling for the table if set to `false`.

Type: Boolean

Required: No

**maximumUnits**

Manage costs by specifying the maximum amount of throughput to provision. The value must be between 1 and the max throughput per second quota for your account (40,000 by default).

Type: Long

Valid Range: Minimum value of 1.

Required: No

**minimumUnits**

The minimum level of throughput the table should always be ready to support. The value must be between 1 and the max throughput per second quota for your account (40,000 by default).

Type: Long

Valid Range: Minimum value of 1.

Required: No

**scalingPolicy**

Amazon Keyspaces supports the `target tracking` auto scaling policy. With this policy, Amazon Keyspaces auto scaling ensures that the table's ratio of consumed to provisioned capacity stays at or near the target value that you specify. You define the target value as a percentage between 20 and 90.

Type: AutoScalingPolicy object

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# AutoScalingSpecification

The optional auto scaling capacity settings for a table in provisioned capacity mode.

## Contents

**readCapacityAutoScaling**

The auto scaling settings for the table's read capacity.

Type: AutoScalingSettings object

Required: No

**writeCapacityAutoScaling**

The auto scaling settings for the table's write capacity.

Type: AutoScalingSettings object

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# CapacitySpecification

Amazon Keyspaces has two read/write capacity modes for processing reads and writes on your tables:

- On-demand (default)
- Provisioned

The read/write capacity mode that you choose controls how you are charged for read and write throughput and how table throughput capacity is managed.

For more information, see Read/write capacity modes in the *Amazon Keyspaces Developer Guide*.

## Contents

**throughputMode**

The read/write throughput capacity mode for a table. The options are:

- `throughputMode:PAY_PER_REQUEST` and
- `throughputMode:PROVISIONED` - Provisioned capacity mode requires `readCapacityUnits` and `writeCapacityUnits` as input.

The default is `throughput_mode:PAY_PER_REQUEST`.

For more information, see Read/write capacity modes in the *Amazon Keyspaces Developer Guide*.

Type: String

Valid Values: `PAY_PER_REQUEST | PROVISIONED`

Required: Yes

**readCapacityUnits**

The throughput capacity specified for `read` operations defined in `read capacity units (RCUs)`.

Type: Long

Valid Range: Minimum value of 1.

Required: No

**writeCapacityUnits**

The throughput capacity specified for `write` operations defined in `write capacity units (WCUs)`.

Type: Long

Valid Range: Minimum value of 1.

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# CapacitySpecificationSummary

The read/write throughput capacity mode for a table. The options are:

- `throughputMode:PAY_PER_REQUEST` and
- `throughputMode:PROVISIONED`.

For more information, see Read/write capacity modes in the *Amazon Keyspaces Developer Guide*.

## Contents

**throughputMode**

The read/write throughput capacity mode for a table. The options are:

- `throughputMode:PAY_PER_REQUEST` and
- `throughputMode:PROVISIONED` - Provisioned capacity mode requires `readCapacityUnits` and `writeCapacityUnits` as input.

The default is `throughput_mode:PAY_PER_REQUEST`.

For more information, see Read/write capacity modes in the *Amazon Keyspaces Developer Guide*.

Type: String

Valid Values: `PAY_PER_REQUEST | PROVISIONED`

Required: Yes

**lastUpdateToPayPerRequestTimestamp**

The timestamp of the last operation that changed the provisioned throughput capacity of a table.

Type: Timestamp

Required: No

**readCapacityUnits**

The throughput capacity specified for `read` operations defined in `read capacity units (RCUs)`.

Type: Long

Valid Range: Minimum value of 1.

Required: No

**writeCapacityUnits**

The throughput capacity specified for `write` operations defined in `write capacity units (WCUs)`.

Type: Long

Valid Range: Minimum value of 1.

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# CdcSpecification

The settings for the CDC stream of a table. For more information about CDC streams, see Working with change data capture (CDC) streams in Amazon Keyspaces in the *Amazon Keyspaces Developer Guide*.

## Contents

**status**

The status of the CDC stream. You can enable or disable a stream for a table.

Type: String

Valid Values: `ENABLED | ENABLING | DISABLED | DISABLING`

Required: Yes

**propagateTags**

Specifies that the stream inherits the tags from the table.

Type: String

Valid Values: `TABLE | NONE`

Required: No

**tags**

The tags (key-value pairs) that you want to apply to the stream.

Type: Array of Tag objects

Array Members: Minimum number of 1 item. Maximum number of 60 items.

Required: No

**viewType**

The view type specifies the changes Amazon Keyspaces records for each changed row in the stream. After you create the stream, you can't make changes to this selection.

The options are:

- NEW_AND_OLD_IMAGES - both versions of the row, before and after the change. This is the default.
- NEW_IMAGE - the version of the row after the change.
- OLD_IMAGE - the version of the row before the change.
- KEYS_ONLY - the partition and clustering keys of the row that was changed.

Type: String

Valid Values: NEW_IMAGE | OLD_IMAGE | KEYS_ONLY | NEW_AND_OLD_IMAGES

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# CdcSpecificationSummary

The settings of the CDC stream of the table. For more information about CDC streams, see Working with change data capture (CDC) streams in Amazon Keyspaces in the *Amazon Keyspaces Developer Guide*.

## Contents

**status**

The status of the CDC stream. Specifies if the table has a CDC stream.

Type: String

Valid Values: `ENABLED | ENABLING | DISABLED | DISABLING`

Required: Yes

**viewType**

The view type specifies the changes Amazon Keyspaces records for each changed row in the stream. This setting can't be changed, after the stream has been created.

The options are:

- `NEW_AND_OLD_IMAGES` - both versions of the row, before and after the change. This is the default.
- `NEW_IMAGE` - the version of the row after the change.
- `OLD_IMAGE` - the version of the row before the change.
- `KEYS_ONLY` - the partition and clustering keys of the row that was changed.

Type: String

Valid Values: `NEW_IMAGE | OLD_IMAGE | KEYS_ONLY | NEW_AND_OLD_IMAGES`

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ClientSideTimestamps

The client-side timestamp setting of the table.

For more information, see How it works: Amazon Keyspaces client-side timestamps in the *Amazon Keyspaces Developer Guide*.

## Contents

**status**

Shows how to enable client-side timestamps settings for the specified table.

Type: String

Valid Values: ENABLED

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ClusteringKey

The optional clustering column portion of your primary key determines how the data is clustered and sorted within each partition.

## Contents

**name**

The name(s) of the clustering column(s).

Type: String

Required: Yes

**orderBy**

Sets the ascendant (ASC) or descendant (DESC) order modifier.

Type: String

Valid Values: `ASC | DESC`

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ColumnDefinition

The names and data types of regular columns.

## Contents

**name**

The name of the column.

Type: String

Required: Yes

**type**

The data type of the column. For a list of available data types, see Data types in the *Amazon Keyspaces Developer Guide*.

Type: String

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# Comment

An optional comment that describes the table.

## Contents

**message**

An optional description of the table.

Type: String

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# EncryptionSpecification

Amazon Keyspaces encrypts and decrypts the table data at rest transparently and integrates with AWS Key Management Service for storing and managing the encryption key. You can choose one of the following AWS KMS keys (KMS keys):

- AWS owned key - This is the default encryption type. The key is owned by Amazon Keyspaces (no additional charge).
- Customer managed key - This key is stored in your account and is created, owned, and managed by you. You have full control over the customer managed key (AWS KMS charges apply).

For more information about encryption at rest in Amazon Keyspaces, see Encryption at rest in the *Amazon Keyspaces Developer Guide*.

For more information about AWS KMS, see AWS KMS management service concepts in the *AWS Key Management Service Developer Guide*.

## Contents

**type**

The encryption option specified for the table. You can choose one of the following KMS keys (KMS keys):

- `type:AWS_OWNED_KMS_KEY` - This key is owned by Amazon Keyspaces.
- `type:CUSTOMER_MANAGED_KMS_KEY` - This key is stored in your account and is created, owned, and managed by you. This option requires the `kms_key_identifier` of the KMS key in Amazon Resource Name (ARN) format as input.

The default is `type:AWS_OWNED_KMS_KEY`.

For more information, see Encryption at rest in the *Amazon Keyspaces Developer Guide*.

Type: String

Valid Values: `CUSTOMER_MANAGED_KMS_KEY | AWS_OWNED_KMS_KEY`

Required: Yes

**kmsKeyIdentifier**

The Amazon Resource Name (ARN) of the customer managed KMS key, for example `kms_key_identifier:ARN`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 5096.

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# FieldDefinition

A field definition consists out of a name and a type.

## Contents

**name**

The identifier.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Required: Yes

**type**

Any supported Cassandra data type, including collections and other user-defined types that are contained in the same keyspace.

For more information, see Cassandra data type support in the *Amazon Keyspaces Developer Guide*.

Type: String

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# KeyspaceSummary

Represents the properties of a keyspace.

## Contents

**keyspaceName**

The name of the keyspace.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

**replicationStrategy**

This property specifies if a keyspace is a single Region keyspace or a multi-Region keyspace. The available values are `SINGLE_REGION` or `MULTI_REGION`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 20.

Valid Values: `SINGLE_REGION | MULTI_REGION`

Required: Yes

**resourceArn**

The unique identifier of the keyspace in the format of an Amazon Resource Name (ARN).

Type: String

Length Constraints: Minimum length of 20. Maximum length of 1000.

Pattern: `arn:(aws[a-zA-Z0-9-]*):cassandra:.+.*`

Required: Yes

**replicationRegions**

If the `replicationStrategy` of the keyspace is `MULTI_REGION`, a list of replication Regions is returned.

Type: Array of strings

Array Members: Minimum number of 2 items.

Length Constraints: Minimum length of 2. Maximum length of 25.

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# PartitionKey

The partition key portion of the primary key is required and determines how Amazon Keyspaces stores the data. The partition key can be a single column, or it can be a compound value composed of two or more columns.

## Contents

**name**

The name(s) of the partition key column(s).

Type: String

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# PointInTimeRecovery

Point-in-time recovery (PITR) helps protect your Amazon Keyspaces tables from accidental write or delete operations by providing you continuous backups of your table data.

For more information, see Point-in-time recovery in the *Amazon Keyspaces Developer Guide*.

## Contents

**status**

The options are:

- `status=ENABLED`
- `status=DISABLED`

Type: String

Valid Values: `ENABLED | DISABLED`

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# PointInTimeRecoverySummary

The point-in-time recovery status of the specified table.

## Contents

**status**

Shows if point-in-time recovery is enabled or disabled for the specified table.

Type: String

Valid Values: `ENABLED | DISABLED`

Required: Yes

**earliestRestorableTimestamp**

Specifies the earliest possible restore point of the table in ISO 8601 format.

Type: Timestamp

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ReplicaAutoScalingSpecification

The auto scaling settings of a multi-Region table in the specified AWS Region.

## Contents

**autoScalingSpecification**

The auto scaling settings for a multi-Region table in the specified AWS Region.

Type: AutoScalingSpecification object

Required: No

**region**

The AWS Region.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 25.

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ReplicaSpecification

The AWS Region specific settings of a multi-Region table.

For a multi-Region table, you can configure the table's read capacity differently per AWS Region. You can do this by configuring the following parameters.

- `region`: The Region where these settings are applied. (Required)
- `readCapacityUnits`: The provisioned read capacity units. (Optional)
- `readCapacityAutoScaling`: The read capacity auto scaling settings for the table. (Optional)

## Contents

**region**

The AWS Region.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 25.

Required: Yes

**readCapacityAutoScaling**

The read capacity auto scaling settings for the multi-Region table in the specified AWS Region.

Type: AutoScalingSettings object

Required: No

**readCapacityUnits**

The provisioned read capacity units for the multi-Region table in the specified AWS Region.

Type: Long

Valid Range: Minimum value of 1.

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ReplicaSpecificationSummary

The Region-specific settings of a multi-Region table in the specified AWS Region.

If the multi-Region table is using provisioned capacity and has optional auto scaling policies configured, note that the Region specific summary returns both read and write capacity settings. But only Region specific read capacity settings can be configured for a multi-Region table. In a multi-Region table, your write capacity units will be synced across all AWS Regions to ensure that there is enough capacity to replicate write events across Regions.

## Contents

**capacitySpecification**

The read/write throughput capacity mode for a table. The options are:

- `throughputMode:PAY_PER_REQUEST` and
- `throughputMode:PROVISIONED`.

For more information, see Read/write capacity modes in the *Amazon Keyspaces Developer Guide*.

Type: CapacitySpecificationSummary object

Required: No

**region**

The AWS Region.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 25.

Required: No

**status**

The status of the multi-Region table in the specified AWS Region.

Type: String

Valid Values: `ACTIVE | CREATING | UPDATING | DELETING | DELETED | RESTORING | INACCESSIBLE_ENCRYPTION_CREDENTIALS`

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ReplicationGroupStatus

This shows the summary status of the keyspace after a new AWS Region was added.

## Contents

**keyspaceStatus**

The status of the keyspace.

Type: String

Valid Values: `ACTIVE | CREATING | UPDATING | DELETING`

Required: Yes

**region**

The name of the Region that was added to the keyspace.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 25.

Required: Yes

**tablesReplicationProgress**

This shows the replication progress of tables in the keyspace. The value is expressed as a percentage of the newly replicated tables with status `Active` compared to the total number of tables in the keyspace.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 7.

Pattern: `[0-9]{1,3}(?:[.][0-9]{1,2})?%`

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ReplicationSpecification

The replication specification of the keyspace includes:

- `regionList` - the AWS Regions where the keyspace is replicated in.
- `replicationStrategy` - the required value is `SINGLE_REGION` or `MULTI_REGION`.

## Contents

**replicationStrategy**

The `replicationStrategy` of a keyspace, the required value is `SINGLE_REGION` or `MULTI_REGION`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 20.

Valid Values: `SINGLE_REGION | MULTI_REGION`

Required: Yes

**regionList**

The `regionList` contains the AWS Regions where the keyspace is replicated in.

Type: Array of strings

Array Members: Minimum number of 2 items.

Length Constraints: Minimum length of 2. Maximum length of 25.

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2

- AWS SDK for Ruby V3

# SchemaDefinition

Describes the schema of the table.

## Contents

**allColumns**

The regular columns of the table.

Type: Array of ColumnDefinition objects

Array Members: Minimum number of 1 item.

Required: Yes

**partitionKeys**

The columns that are part of the partition key of the table .

Type: Array of PartitionKey objects

Array Members: Minimum number of 1 item.

Required: Yes

**clusteringKeys**

The columns that are part of the clustering key of the table.

Type: Array of ClusteringKey objects

Required: No

**staticColumns**

The columns that have been defined as `STATIC`. Static columns store values that are shared by all rows in the same partition.

Type: Array of StaticColumn objects

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# StaticColumn

The static columns of the table. Static columns store values that are shared by all rows in the same partition.

## Contents

**name**

The name of the static column.

Type: String

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# TableSummary

Returns the name of the specified table, the keyspace it is stored in, and the unique identifier in the format of an Amazon Resource Name (ARN).

## Contents

**keyspaceName**

The name of the keyspace that the table is stored in.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

**resourceArn**

The unique identifier of the table in the format of an Amazon Resource Name (ARN).

Type: String

Length Constraints: Minimum length of 20. Maximum length of 1000.

Pattern: `arn:(aws[a-zA-Z0-9-]*):cassandra:.+.*`

Required: Yes

**tableName**

The name of the table.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 48.

Pattern: `[a-zA-Z0-9][a-zA-Z0-9_]{0,47}`

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# Tag

Describes a tag. A tag is a key-value pair. You can add up to 50 tags to a single Amazon Keyspaces resource.

AWS-assigned tag names and values are automatically assigned the `aws:` prefix, which the user cannot assign. AWS-assigned tag names do not count towards the tag limit of 50. User-assigned tag names have the prefix `user:` in the Cost Allocation Report. You cannot backdate the application of a tag.

For more information, see Adding tags and labels to Amazon Keyspaces resources in the *Amazon Keyspaces Developer Guide*.

## Contents

**key**

The key of the tag. Tag keys are case sensitive. Each Amazon Keyspaces resource can only have up to one tag with the same key. If you try to add an existing tag (same key), the existing tag value will be updated to the new value.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Required: Yes

**value**

The value of the tag. Tag values are case-sensitive and can be null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# TargetTrackingScalingPolicyConfiguration

The auto scaling policy that scales a table based on the ratio of consumed to provisioned capacity.

## Contents

**targetValue**

Specifies the target value for the target tracking auto scaling policy.

Amazon Keyspaces auto scaling scales up capacity automatically when traffic exceeds this target utilization rate, and then back down when it falls below the target. This ensures that the ratio of consumed capacity to provisioned capacity stays at or near this value. You define `targetValue` as a percentage. A `double` between 20 and 90.

Type: Double

Required: Yes

**disableScaleIn**

Specifies if `scale-in` is enabled.

When auto scaling automatically decreases capacity for a table, the table *scales in*. When scaling policies are set, they can't scale in the table lower than its minimum capacity.

Type: Boolean

Required: No

**scaleInCooldown**

Specifies a `scale-in` cool down period.

A cooldown period in seconds between scaling activities that lets the table stabilize before another scaling activity starts.

Type: Integer

Required: No

**scaleOutCooldown**

Specifies a scale out cool down period.

A cooldown period in seconds between scaling activities that lets the table stabilize before another scaling activity starts.

Type: Integer

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# TimeToLive

Enable custom Time to Live (TTL) settings for rows and columns without setting a TTL default for the specified table.

For more information, see Enabling TTL on tables in the *Amazon Keyspaces Developer Guide*.

## Contents

**status**

Shows how to enable custom Time to Live (TTL) settings for the specified table.

Type: String

Valid Values: ENABLED

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# Service-specific Errors

The Amazon Keyspaces API contains service-specific exceptions that various actions return. This section describes each exception in detail.

The following service-specific exceptions are returned:

- AccessDeniedException
- ConflictException
- InternalServerException
- ResourceNotFoundException
- ServiceQuotaExceededException
- ValidationException

# AccessDeniedException

You don't have sufficient access permissions to perform this action.

HTTP Status Code returned: 400

## Contents

**message**

You don't have the required permissions to perform this operation. Verify your IAM permissions and try again.

Type: String

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for Ruby V3

# ConflictException

Amazon Keyspaces couldn't complete the requested action. This error may occur if you try to perform an action and the same or a different action is already in progress, or if you try to create a resource that already exists.

HTTP Status Code returned: 400

## Contents

**message**

The requested operation conflicts with the current state of the resource or another concurrent operation.

Type: String

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for Ruby V3

# InternalServerException

Amazon Keyspaces was unable to fully process this request because of an internal server error.

HTTP Status Code returned: 500

## Contents

**message**

An internal service error occurred. Retry your request. If the problem persists, contact AWS Support.

Type: String

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for Ruby V3

# ResourceNotFoundException

The operation tried to access a keyspace, table, or type that doesn't exist. The resource might not be specified correctly, or its status might not be `ACTIVE`.

HTTP Status Code returned: 400

## Contents

**message**

The specified resource was not found. Verify the resource identifier and ensure the resource exists and is in an ACTIVE state.

Type: String

Required: No

**resourceArn**

The unique identifier in the format of Amazon Resource Name (ARN) for the resource couldn't be found.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 1000.

Pattern: `arn:(aws[a-zA-Z0-9-]*):cassandra:.+.*`

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for Ruby V3

# ServiceQuotaExceededException

The operation exceeded the service quota for this resource. For more information on service quotas, see Quotas in the *Amazon Keyspaces Developer Guide*.

HTTP Status Code returned: 400

## Contents

**message**

The requested operation would exceed the service quota for this resource. Review the service quotas and adjust your request accordingly.

Type: String

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for Ruby V3

# ValidationException

The operation failed due to an invalid or malformed request.

HTTP Status Code returned: 400

## Contents

**message**

The request parameters are invalid or malformed. Review the API documentation and correct the request format.

Type: String

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for Ruby V3

# Common Parameters

The following list contains the parameters that all actions use for signing Signature Version 4 requests with a query string. Any action-specific parameters are listed in the topic for that action. For more information about Signature Version 4, see Signing AWS API requests in the *IAM User Guide*.

**Action**

The action to be performed.

Type: string

Required: Yes

**Version**

The API version that the request is written for, expressed in the format YYYY-MM-DD.

Type: string

Required: Yes

**X-Amz-Algorithm**

The hash algorithm that you used to create the request signature.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Valid Values: `AWS4-HMAC-SHA256`

Required: Conditional

**X-Amz-Credential**

The credential scope value, which is a string that includes your access key, the date, the region you are targeting, the service you are requesting, and a termination string ("aws4_request"). The value is expressed in the following format: *access_key*/*YYYYMMDD*/*region*/*service*/aws4_request.

For more information, see Create a signed AWS API request in the *IAM User Guide*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

**X-Amz-Date**

The date that is used to create the signature. The format must be ISO 8601 basic format (YYYYMMDD'T'HHMMSS'Z'). For example, the following date time is a valid X-Amz-Date value: `20120325T120000Z`.

Condition: X-Amz-Date is optional for all requests; it can be used to override the date used for signing requests. If the Date header is specified in the ISO 8601 basic format, X-Amz-Date is not required. When X-Amz-Date is used, it always overrides the value of the Date header. For more information, see Elements of an AWS API request signature in the *IAM User Guide*.

Type: string

Required: Conditional

**X-Amz-Security-Token**

The temporary security token that was obtained through a call to AWS Security Token Service (AWS STS). For a list of services that support temporary security credentials from AWS STS, see AWS services that work with IAM in the *IAM User Guide*.

Condition: If you're using temporary security credentials from AWS STS, you must include the security token.

Type: string

Required: Conditional

**X-Amz-Signature**

Specifies the hex-encoded signature that was calculated from the string to sign and the derived signing key.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

**X-Amz-SignedHeaders**

Specifies all the HTTP headers that were included as part of the canonical request. For more information about specifying signed headers, see Create a signed AWS API request in the *IAM User Guide*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

# Common Errors

This section lists the errors common to the API actions of all AWS services. For errors specific to an API action for this service, see the topic for that API action.

**AccessDeniedException**

You do not have sufficient access to perform this action.

HTTP Status Code: 400

**IncompleteSignature**

The request signature does not conform to AWS standards.

HTTP Status Code: 400

**InternalFailure**

The request processing has failed because of an unknown error, exception or failure.

HTTP Status Code: 500

**InvalidAction**

The action or operation requested is invalid. Verify that the action is typed correctly.

HTTP Status Code: 400

**InvalidClientTokenId**

The X.509 certificate or AWS access key ID provided does not exist in our records.

HTTP Status Code: 403

**NotAuthorized**

You do not have permission to perform this action.

HTTP Status Code: 400

**OptInRequired**

The AWS access key ID needs a subscription for the service.

HTTP Status Code: 403

**RequestExpired**

The request reached the service more than 15 minutes after the date stamp on the request or more than 15 minutes after the request expiration date (such as for pre-signed URLs), or the date stamp on the request is more than 15 minutes in the future.

HTTP Status Code: 400

**ServiceUnavailable**

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

**ThrottlingException**

The request was denied due to request throttling.

HTTP Status Code: 400

**ValidationError**

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400