CreateTable

Service: Amazon DynamoDB

The `CreateTable` operation adds a new table to your account. For each AWS account, table names must be unique within each Region. This also means you can have two tables with the same name, if you create these tables in different Regions.

`CreateTable` is an asynchronous operation. Upon receiving a `CreateTable` request, DynamoDB immediately returns a response with a `TableStatus` of `CREATING`. After the table is created, DynamoDB sets the `TableStatus` to `ACTIVE`. You can perform read and write operations only on an `ACTIVE` table.

You can optionally define secondary indexes on the new table, as part of the `CreateTable` operation. If you want to create multiple tables with secondary indexes on them, you must create the tables sequentially. Only one table with secondary indexes can be in the `CREATING` state at any given time.

You can use the `DescribeTable` action to check the table status.

Request Syntax

```
{
   "AttributeDefinitions": [
      {
         "AttributeName": "string",
         "AttributeType": "string"
      }
   ],
   "BillingMode": "string",
   "DeletionProtectionEnabled": boolean,
   "GlobalSecondaryIndexes": [
      {
         "IndexName": "string",
         "KeySchema": [
            {
               "AttributeName": "string",
               "KeyType": "string"
            }
         ],
         "OnDemandThroughput": {
            "MaxReadRequestUnits": number,
            "MaxWriteRequestUnits": number
```

KeySchema

Specifies the attributes that make up the primary key for a table or an index. The attributes in `KeySchema` must also be defined in the `AttributeDefinitions` array. For more information, see Data Model in the *Amazon DynamoDB Developer Guide*.

Each `KeySchemaElement` in the array is composed of:

- `AttributeName` - The name of this key attribute.
- `KeyType` - The role that the key attribute will assume:
  - `HASH` - partition key
  - `RANGE` - sort key

> **Note**
>
> The partition key of an item is also known as its *hash attribute*. The term "hash attribute" derives from the DynamoDB usage of an internal hash function to evenly distribute data items across partitions, based on their partition key values.
> The sort key of an item is also known as its *range attribute*. The term "range attribute" derives from the way DynamoDB stores items with the same partition key physically close together, in sorted order by the sort key value.

For a simple primary key (partition key), you must provide exactly one element with a `KeyType` of `HASH`.

For a composite primary key (partition key and sort key), you must provide exactly two elements, in this order: The first element must have a `KeyType` of `HASH`, and the second element must have a `KeyType` of `RANGE`.

For more information, see Working with Tables in the *Amazon DynamoDB Developer Guide*.

Type: Array of KeySchemaElement objects

Array Members: Minimum number of 1 item.

Required: Yes

TableName

The name of the table to create. You can also provide the Amazon Resource Name (ARN) of the table in this parameter.

Type: String

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

Required: Yes

BillingMode

Controls how you are charged for read and write throughput, and how you manage capacity. This setting can be changed later.

- PAY_PER_REQUEST - We recommend using PAY_PER_REQUEST for most DynamoDB workloads. PAY_PER_REQUEST sets the billing mode to On-demand capacity mode.
- PROVISIONED - We recommend using PROVISIONED for steady workloads with predictable growth where capacity requirements can be reliably forecasted. PROVISIONED sets the billing mode to Provisioned capacity mode.

Type: String

Valid Values: PROVISIONED | PAY_PER_REQUEST

Required: No

DeletionProtectionEnabled

Indicates whether deletion protection is to be enabled (true) or disabled (false) on the table.

Type: Boolean

Required: No

GlobalSecondaryIndexes

One or more global secondary indexes (the maximum is 20) to be created on the table. Each global secondary index in the array includes the following:

- IndexName - The name of the global secondary index. Must be unique only for this table.
- KeySchema - Specifies the key schema for the global secondary index. Each global secondary index supports up to 4 partition keys and up to 4 sort keys.
- Projection - Specifies attributes that are copied (projected) from the table into the index. These are in addition to the primary key attributes and index key attributes, which are automatically projected.