COURSEWARE

AWS DynamoDB

Contents

Overview

Amazon **DynamoDB** is a NoSQL Database in the cloud, suitable for anyone needing a reliable and fully managed NoSQL solution. DynamoDB is designed to provide automated storage scaling and low latency. It is particularly useful when your application must read and store massive amounts of data and you need speed and reliability. You can scale up or scale down your tables throughput capacity, without downtime or performance degradation.

DynamoDB automatically spreads the data and traffic for your tables over a sufficient number of servers, this is done in order to handle your throughput and storage requirements whilst maintaining fast and consistent performance. All of your data is stored on solid-state disks (SSDs), and is automatically replicated across multiple `Availability Zones` in an AWS Region, this provides built-in high availability and data durability.

Table Name

This is where you define the name of the table, since this is a managed non-relational database, there is no
provisioning of infrastructure so this is part of the serverless architecture.
See below for example of how to set the table name:

```
aws dynamodb create-table --table-name [TABLE_NAME]
```

Attribute Definition

This is where you will define the name of the items or columns in relational database. See below for an example of how to define attributes:

```
aws dynamodb create-table --table-name [TABLE_NAME] --attribute-definitions
AttributeName=[ATTRIBUTE_NAME],AttributeType=S
```

Key Schema

This is defining the properties of your Attributes. There are only 2 properties that you can sit, "HASH" and
"RANGE".

See below for an example:

```
--key-schema AttributeName=MovieName,KeyType=HASH
AttributeName=MovieYear,KeyType=RANGE
```

Tutorial

Pre-requisites

- Make sure you have AWS CLI installed

- Configure the AWS CLI using the following command `aws configure`
  In this tutorial we will be creating a DynamoDB table through AWS CLI. The table we will create will be a Movie table where you enter the movie title and
  release year.

1. Make sure you run the AWS CLI commands from a machine that is configured with AWS. Run the following:

```
# creating dynamodb table
aws dynamodb create-table --table-name Movies --attribute-definitions
AttributeName=MovieName,AttributeType=S AttributeName=MovieYear,AttributeType=S
--key-schema AttributeName=MovieName,KeyType=HASH
AttributeName=MovieYear,KeyType=RANGE --provisioned-throughput
ReadCapacityUnits=5,WriteCapacityUnits=5
```

2. If that ran successfully, you should see an output in json stating the table thats going to be created. We can
   now start adding items to our table. Run the following command:

```
# creating json file to import data into table
cat << EOF > movies.json
{
    "MovieName": {"S": "Avengers Assemble"},
    "MovieYear": {"S": "2012"}
}
EOF
```

Using the movies.json file to upload data to DynamoDB

```
# Adding data to movies table
aws dynamodb put-item --table-name Movies --item file://movies.json
```

3. Lets extract information about out Table. Run the following command:

```
# Get information about table Movies
aws dynamodb describe-table --table-name Movies
```

4. To get a list of items in the table, run the following:

This command will return a single for as you need to define the specific items to get.

```
# Get items from Movies table
aws dynamodb get-item --table-name Movies --key file://movies.json
```

If you want to get all items in a table, you will need to scan the table, run the following commands:

```
# Get all items in Movies table
aws dynamodb scan --table-name Movies
```

5. Clean the AWS Environment. Delete the table using this command:

```
# Delete Movies Table
aws dynamodb delete-table --table-name Movies
```

Exercises

There are no exercises for this module.