Introduction to Amazon DynamoDB

Lab overview

Amazon DynamoDB is a fast and flexible NoSQL database service for all applications that need consistent, single-digit millisecond latency at any scale. It is a fully managed database that supports both document and key-value data models. Its flexible data model and reliable performance make it a great fit for mobile, web, gaming, advertising technology, Internet of Things (IoT), and many other applications.

In this lab, you create a table in DynamoDB to store information about a music library. You then query the music library and, finally, delete the DynamoDB table.

Objectives

After completing this lab, you will know how to:

- Create a DynamoDB table
- Enter data into a DynamoDB table
- Query a DynamoDB table
- Delete a DynamoDB table

Task 1: Creating a New Table

In this task, you create a new table in DynamoDB named **Music**. Each table requires a primary key that is used to partition data across DynamoDB servers. A table can also have a sort key. The combination of a primary key and a sort key uniquely identifies each item in a DynamoDB table.

5. In the AWS Management Console, choose **Services**, and then choose **DynamoDB**.

6. Choose **Create table**

7. For **Table name**, enter `Music`

8. For **Partition key**, enter `Artist` and leave **String** selected.

9. For **Sort key - *optional***, enter `Song` and leave **String** selected.

10. Under **Settings** select **Customize settings** then configure the following:

    ### Read capacity

    - **Provisioned capacity units:** `10`

    ### Write capacity

    - **Provisioned capacity units:** `2`

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

- On-demand
- Provisioned (default, free-tier eligible)

The read/write capacity mode controls how you are charged for read and write throughput and how you manage capacity. You can set the read/write capacity mode when creating a table or you can change it later.

Your table will use default settings for indexes.

11. Choose **Create table**.

    The table is created in less than 1 minute.

Task 2: Adding Data

In this task, you add data to the **Music** table. A table is a collection of data on a particular topic.

Each table contains multiple items. An item is a group of attributes that is uniquely identifiable among all of the other items. Items in DynamoDB are similar in many ways to rows in other database systems. In DynamoDB, there is no limit to the number of items that you can store in a table.

Each item consists of one or more attributes. An attribute is a fundamental data element, something that does not need to be broken down any further. For example, an item in a **Music** table contains attributes such as song and artist. Attributes in DynamoDB are similar to columns in other database systems, but each item (row) can have different attributes (columns).

When you write an item to a DynamoDB table, only the primary key and sort key (if used) are required. Other than these fields, the table does not require a schema, which means that you can add attributes to one item that may be different than the attributes on other items.

12. Select **Explore items** on the left side navigation pane.
13. Click the radio button next to **Music** to select the table you created.
14. Click **Create item**.
15. Add in the following values.
    - **Artist:** `Pink Floyd`
    - **Song:** `Money`

These are the only required attributes, but you can now add additional attributes.

16. To create an additional attribute, click the **Add new attribute** button.
17. In the dropdown list, choose **String**.

    A new attribute row is added.
18. For the new attribute, replace **NewValue** with `Album` and in the **Value** column enter `The Dark Side of the Moon`.
19. Add another new attribute by choosing the **Add new attribute** button.
20. In the dropdown list, choose **Number**.
21. For the new attribute, replace **NewValue** with `Year` and in the **Value** column replace **0** with `1973`.
22. Choose **Create item** to store the new item with its four attributes.

    The item appears in the **Items returned** pane in the console.

23. Next, follow the previous steps and use the following attributes to create a second item:

Attribute Name	Attribute Type	Attribute Value
Artist	String	John Lennon
Song	String	Imagine
Album	String	Imagine
Year	Number	1971
Genre	String	Soft rock

Note that this item has an additional attribute called **Genre**. Adding this attribute is an example of each item being capable of having different attributes without having to pre-define a table schema.

24. Follow the previous steps and use the following attributes to create a third item:

Attribute Name	Attribute Type	Attribute Value
Artist	String	Psy
Song	String	Gangnam Style
Album	String	Psy 6 (Six Rules), Part 1
Year	Number	2011
LengthSeconds	Number	219

Once again, this item has a new **LengthSeconds** attribute that identifies the length of the song. The ability to include this attribute demonstrates the flexibility of a NoSQL database.

There are also faster ways to load data into DynamoDB, such as using AWS Data Pipeline, programmatically loading data, or using one of the free tools available on the internet.

Task 3: Modifying an Existing Item

You now notice that there is an error in your data. In this task, you modify an existing item.

25. From the list of items, select the row where the **Artist** is **Psy**.
26. Choose the **Actions** menu, and select **Edit item**.
27. Change the **Year Number** from **2011** to **2012**.
28. Choose **Save changes**.

The item is now updated.

Task 4: Querying the Table

There are two ways to query a DynamoDB table: **query** and **scan**.

A query operation finds items based on the Primary Key and optionally based on the Sort Key. It is fully indexed, so it runs very fast.

29. Choose **Explore items** in the left navigation pane.
30. Select **Music**.

31. If **Scan/Query items** is not already expanded, choose the arrow to expose the **Scan** and **Query** options.

32. Choose **Query**.

    Fields for the **Artist** (which is the same as partition key) and **Song** (which is the same as sort key) are now displayed.

33. Enter the following details:

    - **Artist (Partition key):** `Psy`
    - **Song (Sort key): Equal to** `Gangnam Style`

34. Choose **Run**.

The song quickly appears in the list. A query is the most efficient way to retrieve data from a DynamoDB table.

Alternatively, you can scan for an item. This option involves looking through every item in a table, so this option is less efficient and can take significant time for larger tables.

35. Choose the **Scan** option.

36. Choose the arrow to expand **Filters**.

37. Enter values for the scan filter:

    - For **Enter attribute name**, enter `Year`
    - Change **String** to **Number**.
    - **Condition:** Select **Equal to**
    - For **Enter value**, enter `1971`
    - Choose **Run**.

Only the song released in 1971 is displayed.

Task 5: Deleting an Item

In this task, you delete an item within the table.

38. Choose **Reset** then click **Run**. To load the full Music table.

39. Under **Artist**, select the check box for **Psy** to choose this item.

40. Choose the **Actions** dropdown list, and select **Delete items**.

41. On the **Delete item(s)** screen, choose **Delete**.

    This item is now deleted.