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 and supports both document and key-value data models. Its flexible data model and reliable performance make it a great fit for mobile, web, gaming, ad-tech, Internet of Things (IoT), and many other applications.

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

Topics covered

In this lab, you will:

- Create an Amazon DynamoDB table
- Enter data into an Amazon DynamoDB table
- Query an Amazon DynamoDB table
- Delete an Amazon DynamoDB table

Duration

This lab requires approximately **35 minutes** to complete.

Accessing the AWS Management Console

1. At the upper-right corner of these instructions, choose **Start Lab**

Troubleshooting tip: If you get an **Access Denied** error, close the error box, and choose **Start Lab** again.

2. The lab status can be interpreted as follows:
    - A red circle next to **AWS** at the upper-left corner of this page indicates the lab has not been started.
    - A yellow circle next to **AWS** at the upper-left corner of this page indicates the lab is starting.
    - A green circle next to **AWS** at the upper-left corner of this page indicates the lab is ready.

    Please wait for the lab to be ready before proceeding.

3. At the top of these instructions, choose the green circle next to **AWS**

    This option will open the AWS Management Console in a new browser tab. The system will automatically sign you in.

    Tip: If a new browser tab does not open, a banner or icon at the top of your browser will indicate that your browser is preventing the site from opening pop-up windows. Choose the banner or icon, and choose **Allow pop-ups**.

4. Arrange the AWS Management Console tab so that it displays along side these instructions. Ideally, you should be able to see both browser tabs at the same time so that you can follow the lab steps.

    Do not change the lab Region unless specifically instructed to do so.

Task 1: Create a new table

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

5. In the AWS Management Console, choose the **Services** menu. Under **Database**, choose **DynamoDB**.
6. Choose **Create table**.

7. For the **Table name**, enter `Music`
8. For the **Partition key**, enter `Artist` and leave **String** selected in the dropdown list.
9. For **Sort key - *optional***, enter `Song` and leave **String** selected.

   Your table will use the default settings for indexes and provisioned capacity.

10. Scroll down, and choose **Create table**.

    The table will be created in less than 1 minute. Wait for the **Music** table to be **Active** before moving on to the next task.

Task 2: Add data

In this task, you will 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 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 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 partition key and sort key (if used) are required. Other than these fields, the table does not require a schema. This means that you can add attributes to one item that may be different than the attributes for other items.

11. Choose the **Music** table.
12. Choose **Actions**, and then choose **Create item**.
13. For the **Artist** value, enter `Pink Floyd`
14. For the **Song** value, enter `Money`

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

15. To create an additional attribute, choose **Add new attribute**.
16. In the dropdown list, select **String**.

    A new attribute row will be added.

17. For the new attribute, enter the following:
    - **FIELD**: `Album`
    - **VALUE**: `The Dark Side of the Moon`
18. To add another new attribute, choose **Add new attribute**.
19. In the dropdown list, choose **Number**.

    A new number attribute will be added.

20. For the new attribute, enter the following:
    - **FIELD**: `Year`
    - **VALUE**: `1973`
21. Choose **Create item**.

    The item has now been added to the **Music** table.

22. Similarly, to create a second item, use the following attributes:

Attribute Name	Attribute Type	
Artist	String	John L
Song	String	Imagi
Album	String	Imagi
Year	Number	1971
Genre	String	Soft ro

23. This item has an additional attribute called **Genre**. This is an example of each item being capable of having different attributes without having to pre-define a table schema.
24. To create a third item, use the following attributes:

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

25. Once again, this item has a new **LengthSeconds** attribute identifying the length of the song. This demonstrates the flexibility of a NoSQL database.
26. There are also faster ways to load data into DynamoDB, such as using AWS Command Line Interface, programmatically loading data, or using one of the free tools available on the internet.

Task 3: Modify an existing item

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

24. In the DynamoDB dashboard, under **Tables**, choose **Explore Items**.
25. Choose the **Music** button.
26. Choose **Psy**.
27. Change the **Year** from **2011** to **2012**.
28. Choose **Save changes**.

    The item is now updated.

Task 4: Query 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 the sort key. It is fully indexed, so it runs very fast.

29. Expand **Scan/Query items**, and choose **Query**.

    Fields for the Artist (Partition key) and Song (Sort key) are now displayed.

30. Enter the following details:
    - **Artist (Partition key):** `Psy`
    - **Song (Sort key):** `Gangnam Style`
31. Choose **Run**.

    The song quickly appears in the list. You might need to scroll down to see this result.

    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 it is less efficient and can take significant time for larger tables.

32. Scroll up on the page, and choose **Scan**.
33. Expand **Filters**, and enter the following values:
    - For **Attribute name**, enter `Year`
    - For **Type**, choose **Number**.
    - For **Value**, enter `1971`
34. Choose **Run**

    Only the song released in 1971 is displayed.

Task 5: Delete the table

In this task, you will delete the **Music** table, which will also delete all the data in the table.

35. In the DynamoDB dashboard, under **Tables**, choose **Update settings**.
36. Choose the **Music** table if it is not already selected.
37. Choose **Actions**, and then choose **Delete table**.
38. On the confirmation panel, enter `delete` and choose **Delete table**.

    The table will be deleted.

Conclusion

Congratulations! You now have successfully:

- Created an Amazon DynamoDB table
- Entered data into an Amazon DynamoDB table
- Queried an Amazon DynamoDB table
- Deleted an Amazon DynamoDB table

For more information about DynamoDB, see the DynamoDB documentation.

Lab complete

39. Choose **End Lab** at the top of this page, and then select **Yes** to confirm that you want to end the lab.
40. An **Ended AWS Lab Successfully** message is briefly displayed indicating that the lab has ended.

For more information about AWS Training and Certification, see AWS Training and Certification.

Your feedback is welcome and appreciated. If you would like to share any suggestions or corrections, please provide the details in our AWS Training and Certification Contact Form.