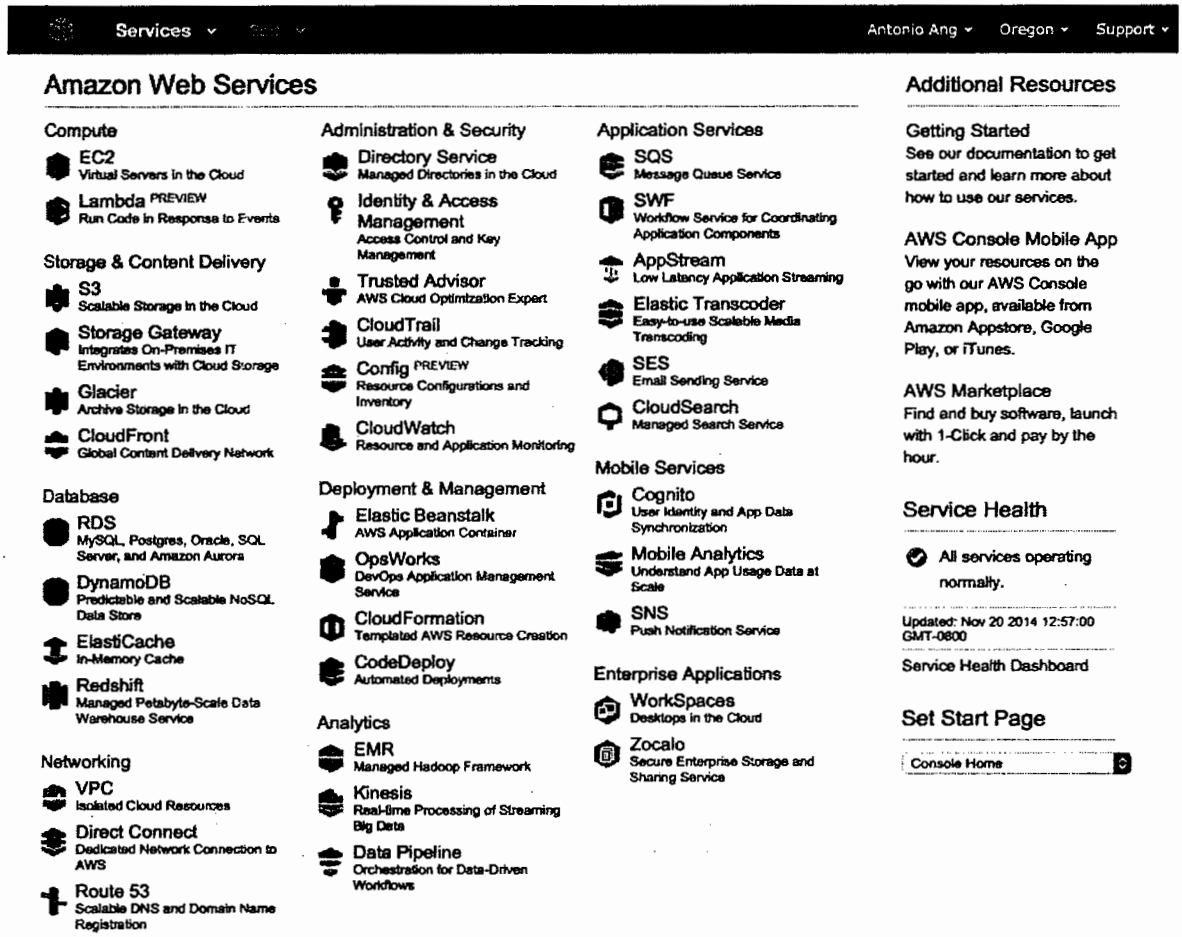
Lab 10

INTRODUCTION TO DYNAMO DB

STEP 1: Log In to the Amazon Web Service Console

This laboratory experience is about Amazon Web Services and you will use the AWS Management Console in order to complete all the lab steps.



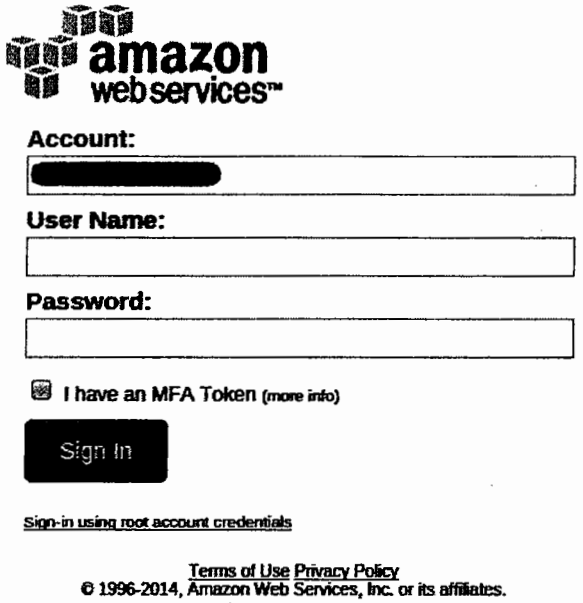
The AWS Management Console is a web control panel for managing all your AWS resources, from EC2 instances to SNS topics. The console enables cloud management for all aspects of the AWS account, including managing security credentials, or even setting up new IAM Users.

Log in to the AWS Management Console

In order to start the laboratory experience, open the Amazon Console by clicking this button:

Open AWS Console

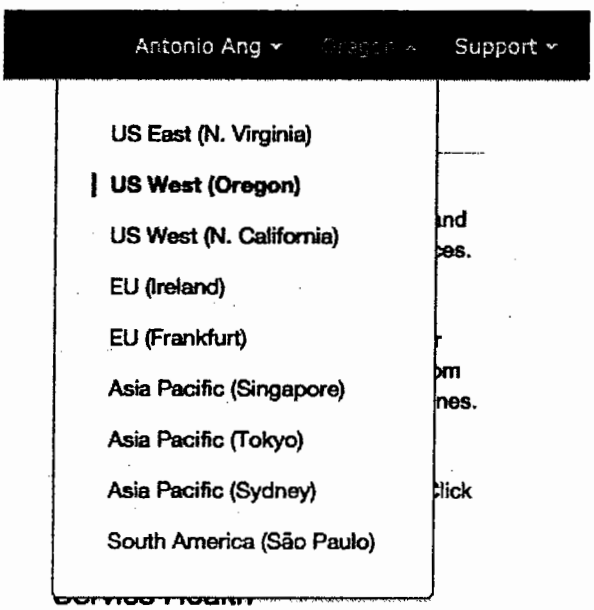
Log in with the username **xxxx** and the password **xxxx** .



Select the right AWS Region

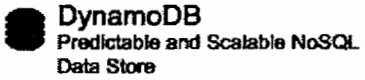
Amazon Web Services is available in different regions all over the world, and the console lets you provision resources across multiple regions. You usually choose a region that best suits your business needs to optimize your customer's experience, but you must use the region **US West (Oregon)** for this laboratory.

You can select the **US West (Oregon)** region using the upper right dropdown menu on the AWS Console page.



STEP 2: Create a DynamoDB table with a Hash Key

Creating a DynamoDB table is a simple and can be done using the AWS Management Console. Open the AWS Management console dashboard, click on the DynamoDB Database service, and you'll see the Amazon DynamoDB dashboard page.

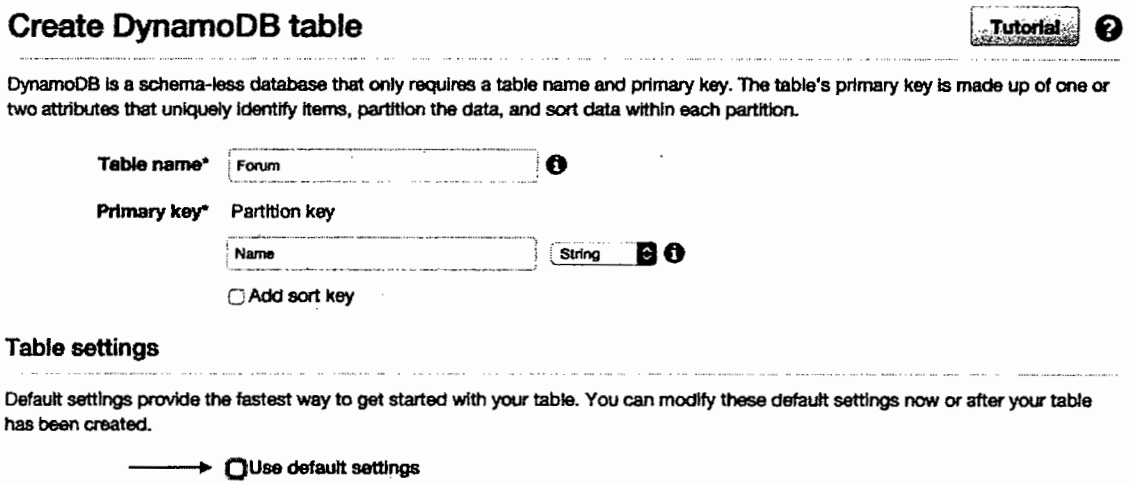


In order to create a DynamoDB table, click the **Create Table** button and the Create Table wizard will be displayed.

You must specify the table name, the primary key type and the attribute name.

Please use the following data:

- ✓ Table Name: **Forum**
- ✓ Primary Key Type: **Hash**
- ✓ Hash Attribute Name: **Name** of type **String**

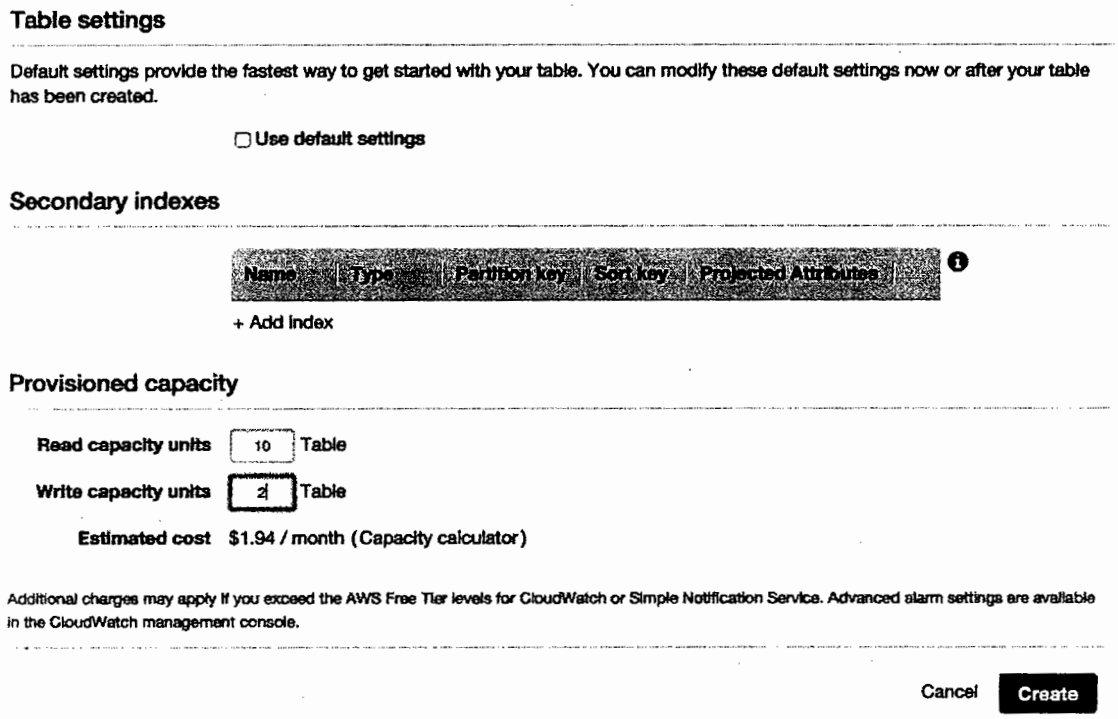


Uncheck the "Use default settings" checkbox for configuring the table **provisioned throughput capacity**.

You can specify how much provisioned throughput capacity you want to reserve for reads and writes. DynamoDB will reserve the necessary machine resources to meet your throughput needs while ensuring consistent, low-latency performance. A unit of *read capacity* represents one strongly consistent read per second (or two eventually consistent reads per second) for items as large as 4 KB. A unit of *write capacity* represents one write per second for items as large as 1 KB.

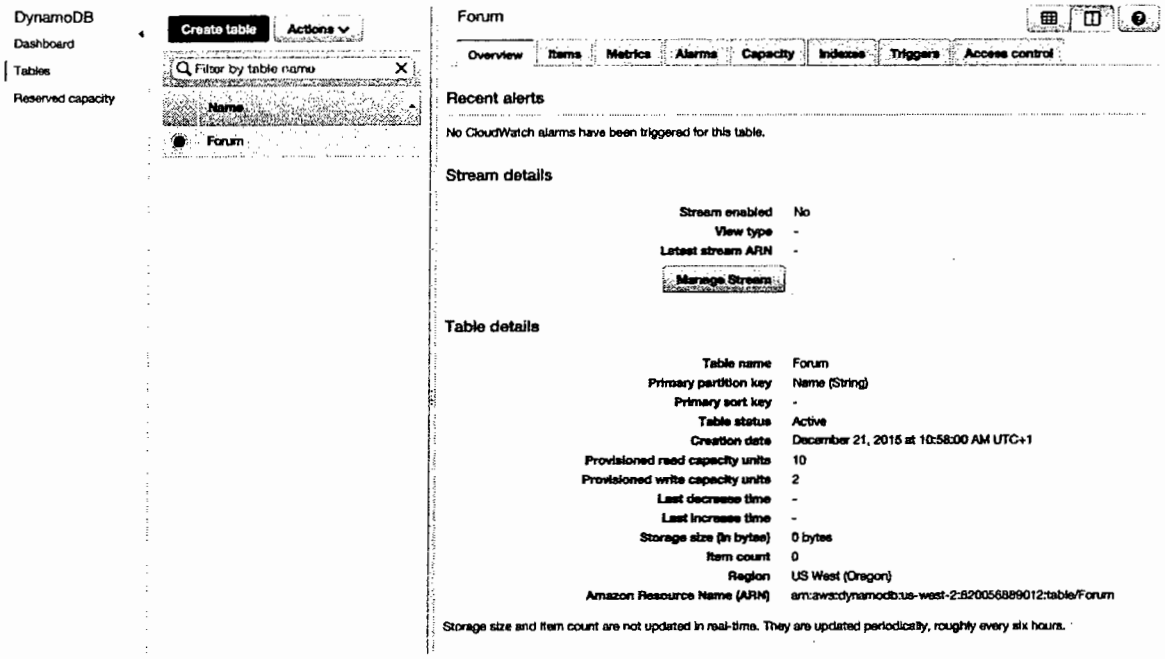
Set the following capacity units for the **Forum** table:

- ✓ Read Capacity Units: **10**
- ✓ Write Capacity Units: **2**



Click **Create** to start the table creation process.

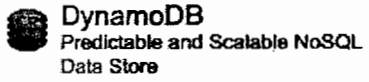
After a couple of seconds, your new table will be listed in the Amazon DynamoDB dashboard.



STEP 3: Create a DynamoDB table with local and global secondary indexes

Creating a DynamoDB table is simple and can be done using the AWS Management Console.

If you are displaying the AWS Management console dashboard, click on the DynamoDB Database service and you will see the Amazon DynamoDB dashboard page.

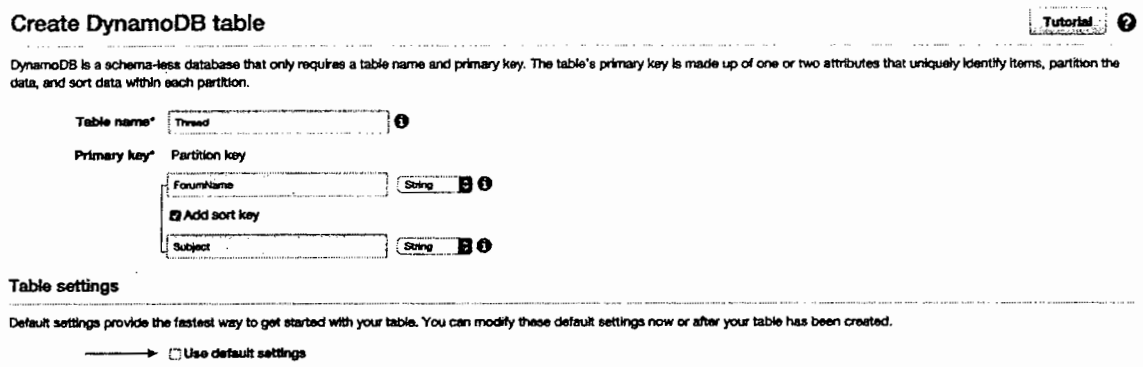


In order to create a DynamoDB table, click the **Create Table** button and the Create Table wizard will be displayed.

You need to specify the table name, the primary key type and the attribute name.

Please use the following data:

- ✓ Table Name: **Thread**
- ✓ Primary Key: **ForumName** of type **String**
- ✓ Sort Key: **Subject** of type **String**



Uncheck the "Use default settings" checkbox for configuring table **indexes** and the **provisioned throughput capacity**.

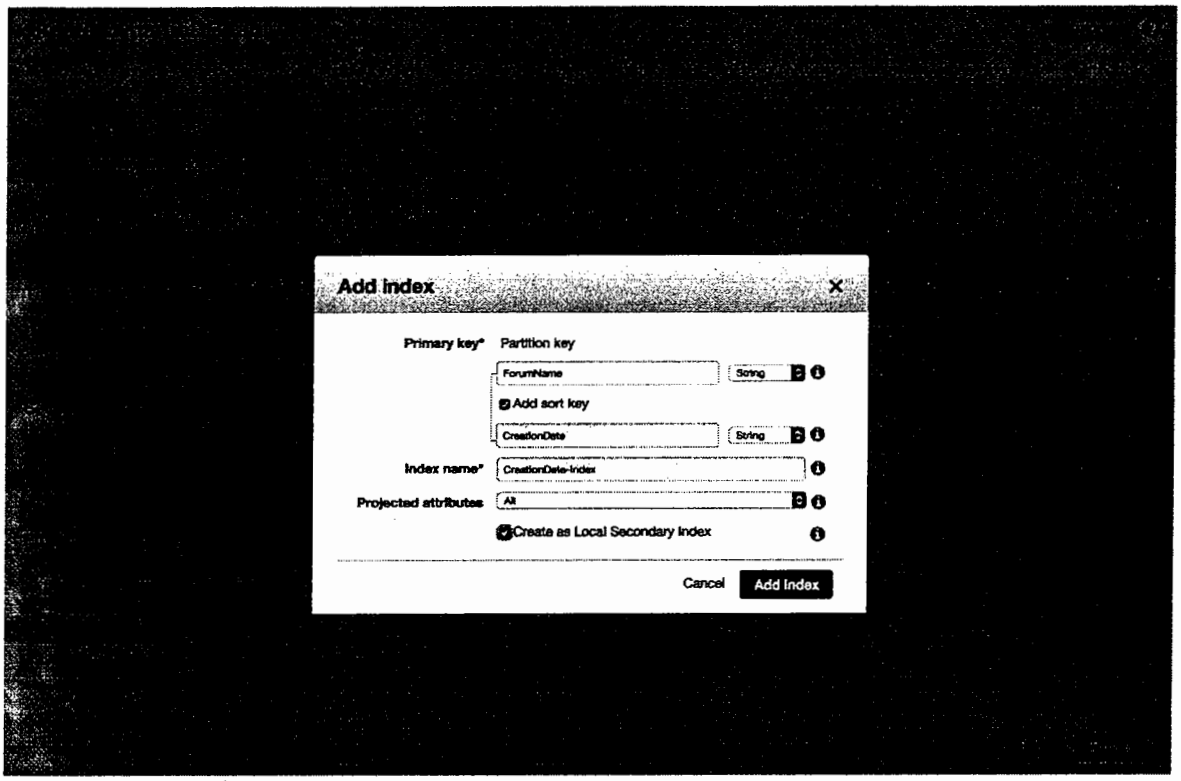
A secondary index is a data structure that contains a subset of attributes from a table, along with an alternate key to support query operations. With a secondary index, queries are no longer restricted to the table primary key. You can also retrieve the data using the alternate key defined by the secondary index. A table can have multiple secondary indexes, which gives your applications access to many different query patterns. The data in a secondary index consists of attributes that are projected, or copied from the table into the index.

When you create a secondary index, you define the alternate key for the index, along with any other attributes that you want to be projected in the index. DynamoDB copies these attributes into the index, along with the primary key attributes from the table. You can then query the index just as you would query a table. Every secondary index is automatically maintained by DynamoDB. When you add, modify, or delete items, the table indexes are also updated to reflect these changes.

Click **+Add Index** and then create a **Local Secondary Index** for the **Thread** table with the following specifications:

- ✓ Primary Key: **ForumName** of type **String**
- ✓ Sort Key: **CreationDate** of type **String**
- ✓ Index Name: **CreationDate-index** table
- ✓ Projected Attributes: **All Attributes**
- ✓ Index Type: **Local Secondary Index**

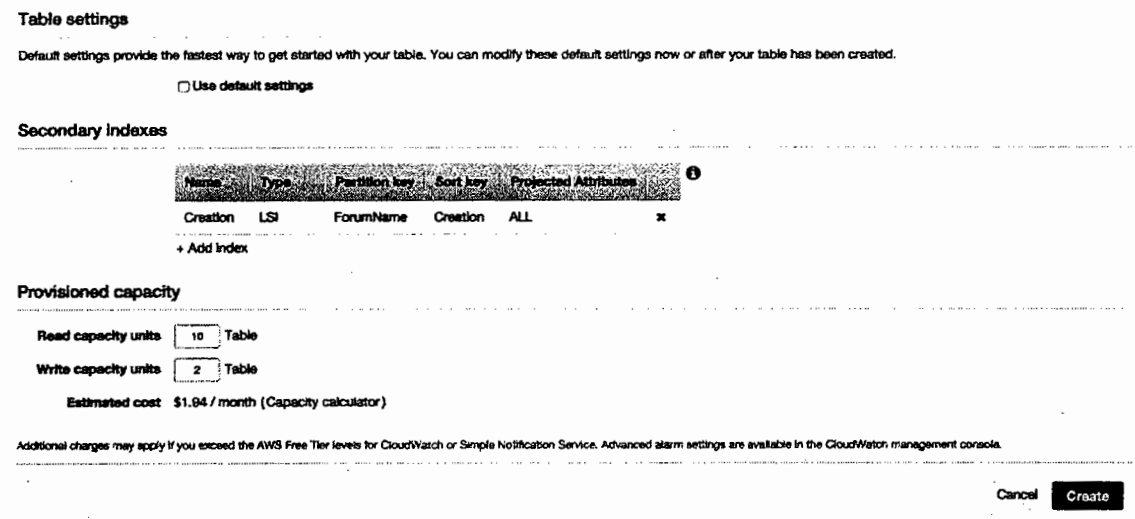
Click on the **Add Index** button and then on **Continue**.



Remember to set the **provisioned throughput capacity.** You specify how much provisioned throughput capacity you want to reserve for reads and writes. DynamoDB will reserve the necessary machine resources to meet your throughput needs while ensuring consistent, low-latency performance. A unit of *read capacity* represents one strongly consistent read per second (or two eventually consistent reads per second) for items as large as 4 KB. A unit of *write capacity* represents one write per second for items as large as 1 KB.

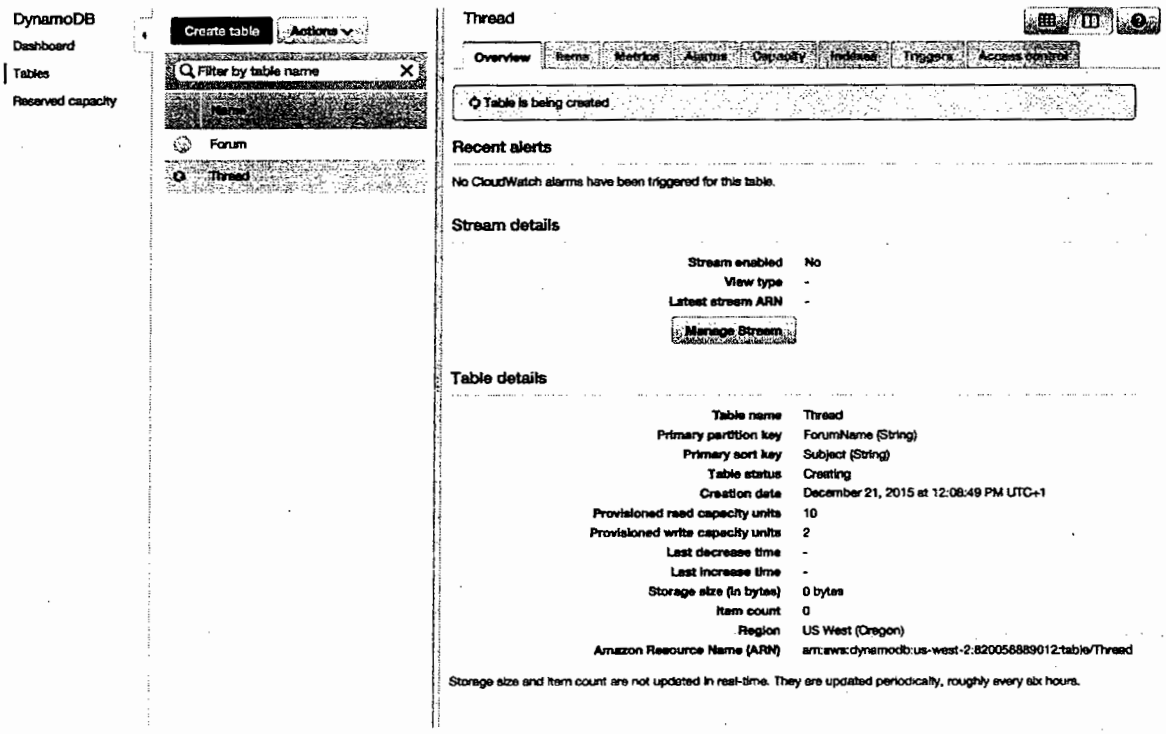
Specify the following capacity units for the **Thread** table:

- ✓ Read Capacity Units: **10**
- ✓ Write Capacity Units: **2**



Click **Create** to start its creation process.

After a few of seconds, your new table will be listed in the Amazon DynamoDB dashboard as ACTIVE state.



Create a table with Global and Local Secondary Index

A Global Secondary Index is an index with a hash and range key that can be different from those in the table. It is considered "global" because queries on the index can span all of the data in a table, across all partitions. Remember that indexes must be created at the same time you create a table. You cannot add, edit or delete any secondary index to an existing table.

Creating a Global Secondary index is similar to creating a table with a Local one. Starting from the DynamoDB dashboard, click the **Create Table** button to open the Create Table wizard.

Please use the following data for the step 1:

- ✓ Table Name: **Reply**
- ✓ Primary Key: **ID** of type **String**
- ✓ Sort key: **CreationDate** of type **String**

Uncheck the "Use default settings" checkbox for configuring table **indexes** and the **provisioned throughput capacity**.

First of all, create a **Local Secondary Index** for the **Reply** table:

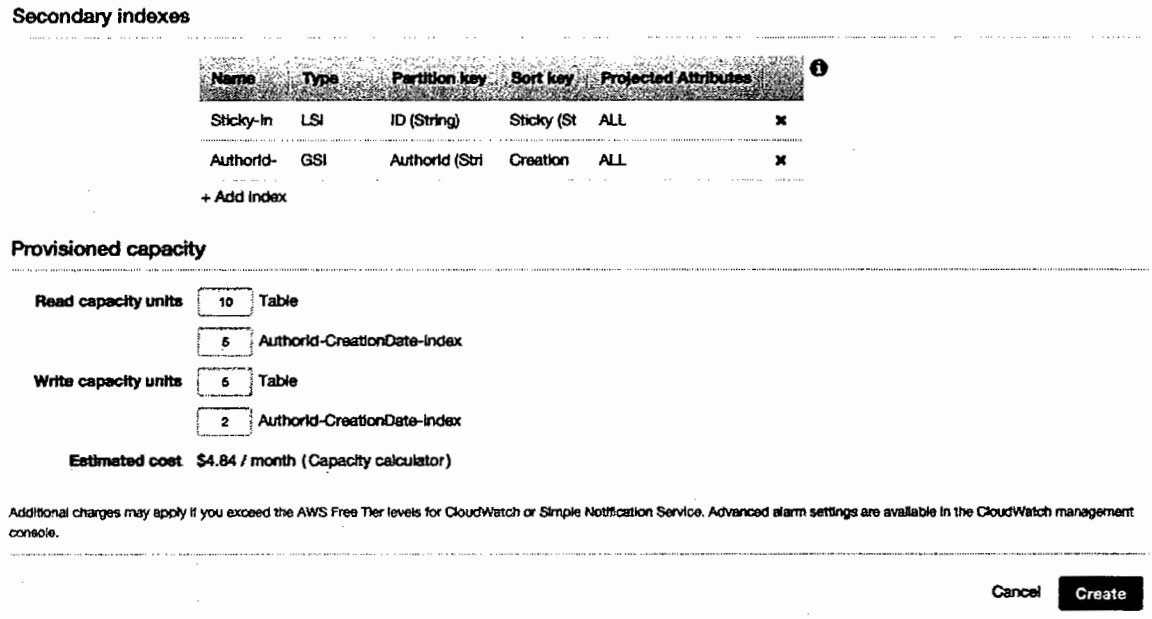
- ✓ Primary Key: **ID** of type **String**
- ✓ Sort Key: **Sticky** of type **String**
- ✓ Index Name: **Sticky-index**
- ✓ Projected Attributes: **All Attributes**
- ✓ Index type: **Local Secondary Index**

Click the **Add Index** button to add it to the Table Indexes that will be created.

In order to add another index, you simply have to fill in the same form again. Create a **Global Secondary Index** using the following data:

- ✓ Primary Key: **AuthorId** of type **String**
- ✓ Sort Key: **CreationDate** of type **String**
- ✓ Index Name: **AuthorId-CreationDate-index**
- ✓ Projected Attributes: **All Attributes**
- ✓ Index type: **Global Secondary Index**

Click **Add Index** and then configure the **provisioned throughput capacity**.



In 2015, Amazon added the possibility to split the R/W capacity units between the main table and the Global Secondary Index one.

Please set the following capacity units for the **Reply** table (it's named "Table") and the AuthorId-CreationDate-index table:

- ✓ Table Read Capacity Units: **10**
- ✓ AuthorId-CreationDate-index Read Capacity Units: **5**
- ✓ Write Capacity Units: **5**
- ✓ AuthorId-CreationDate-index Write Capacity Units: **2**

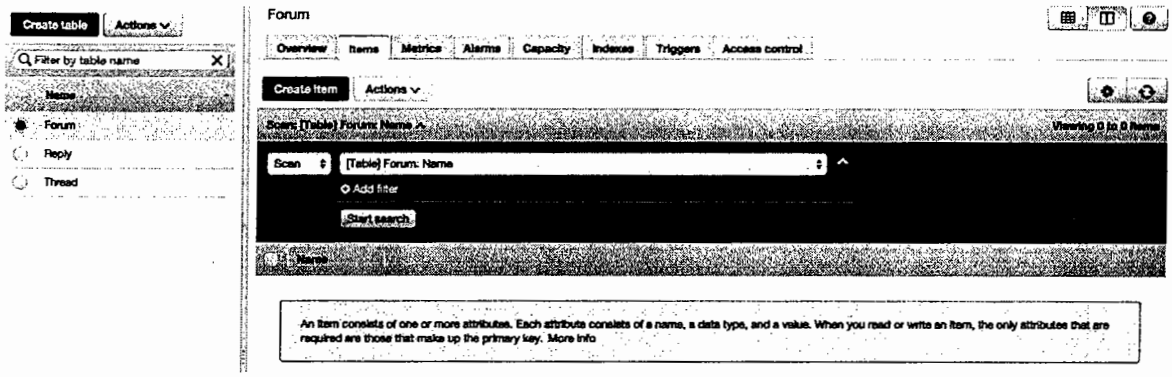
All Global Indexes are additional DynamoDB tables that are automatically syncronized when you add new items to the main table.

Click **Create** to start the creation operation.

After a few seconds, your new table will be listed in the Amazon DynamoDB dashboard as ACTIVE state.

STEP 4: Insert records into a DynamoDB table

After creating all the needed tables, you are ready to fill them with demo data. The AWS Management Console allows you to show the items stored in a DynamoDB Table by selecting it (click on Forum) and clicking on the **Items** tab.

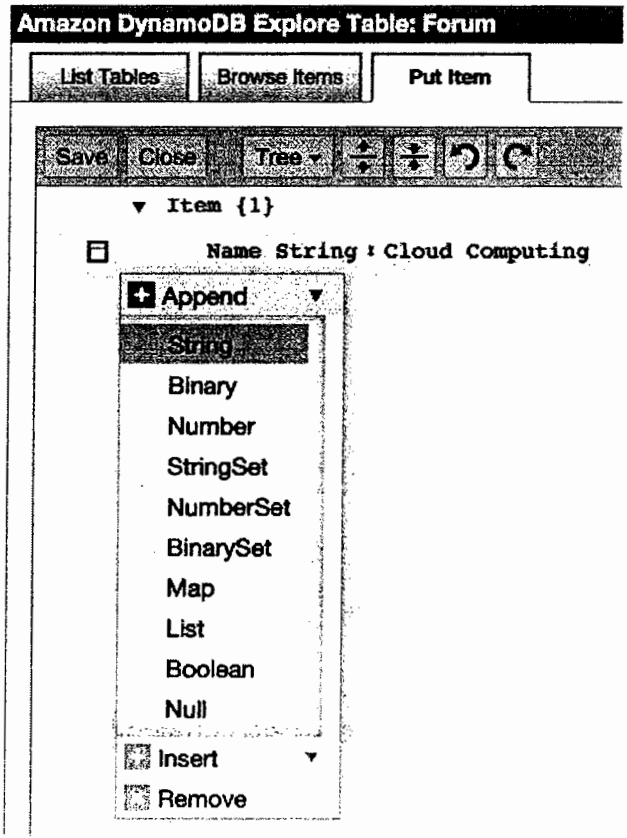


The DynamoDB console automatically executes a **Scan** query for listing all the items stored in Forum. You will see an empty page because there are no items stored.

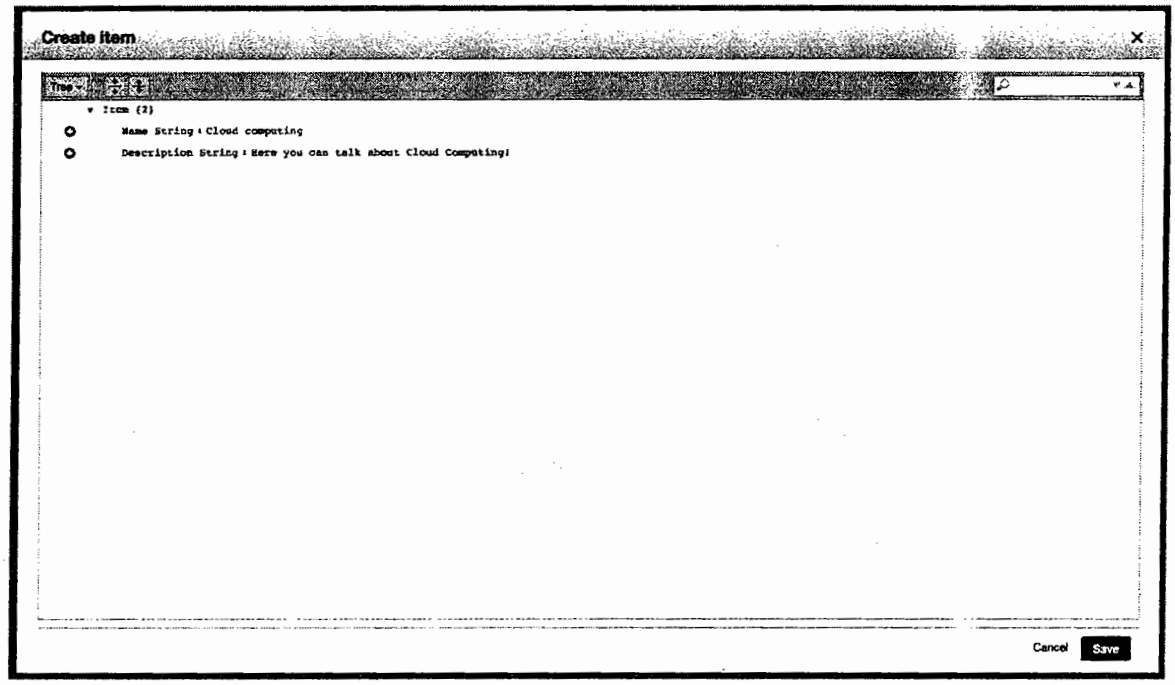
To add a new item, click **Create Item** and a tab pane will appear.

Forum only has a Hash Key named Name, so it's the only mandatory field to fill to add a new item.

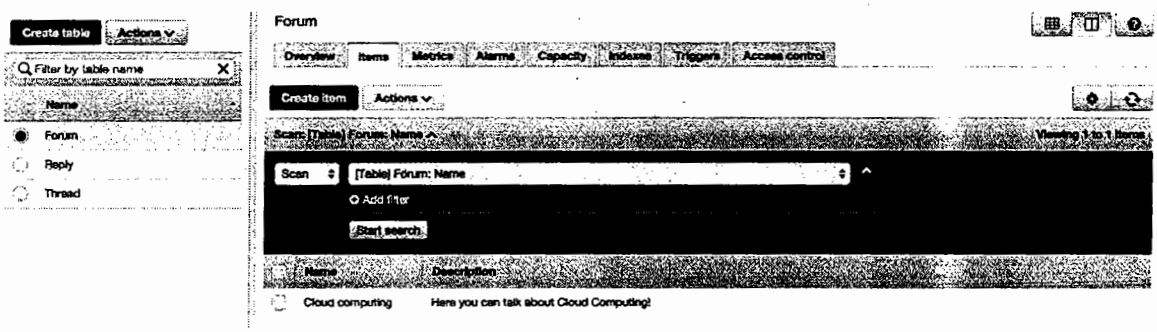
If you want to store more data, you can click on the small icon on the right of the highlighted field, select the proper field type (for eg. String) and choose a field name and value.



For example, you may add a **Description** field of type **String** for your Forum.



Click **Save** to store all the new items.



Please repeat the entire operation three or more times. You will use the demo data in the next lab steps.

Add rows to the Thread table

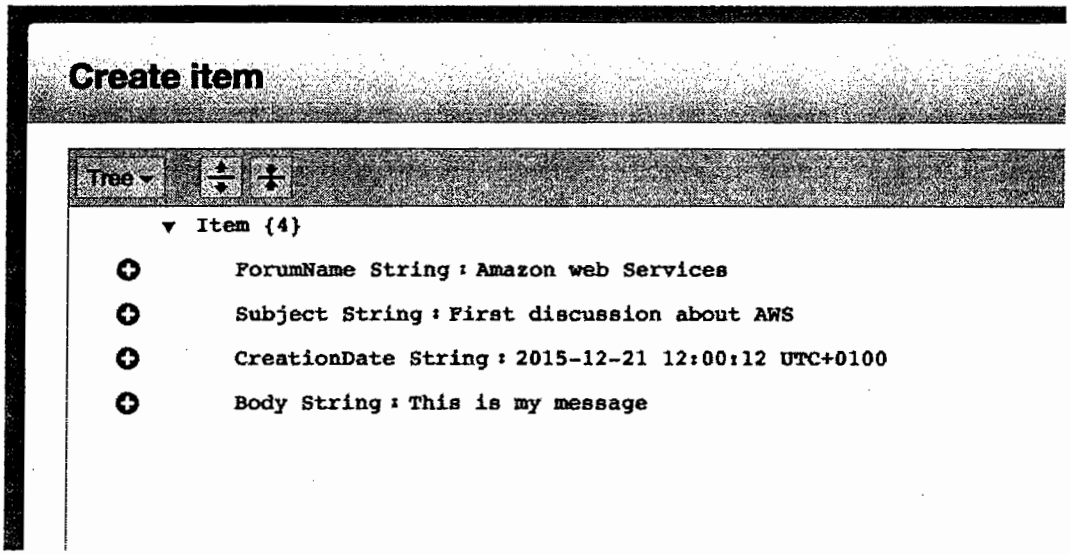
Go back to the DynamoDB dashboard, select Thread, select the **Items** tab pane and then on**Create Item**.

The Thread is a "Hash and Range" table with the CreationDate-index Local Secondary Index. For being able to save a Thread item, you have to provide:

- ✓ ForumName (the table Primary Key)
- ✓ Subject (the table Sort Key)

✓ CreationDate (the Local Secondary Index Sort Key)

You can also add other optional fields as a **Body** field where you can store Thread's first message.



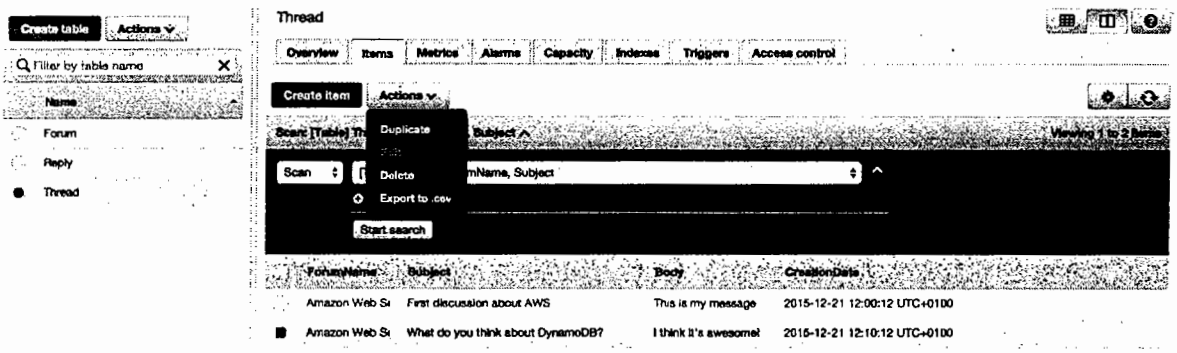
Please repeat the entire operation three or more times, you will use the demo data in the next lab steps.

STEP 5: Edit DynamoDB table rows

The AWS Management Console also allows you to edit a previously created items.

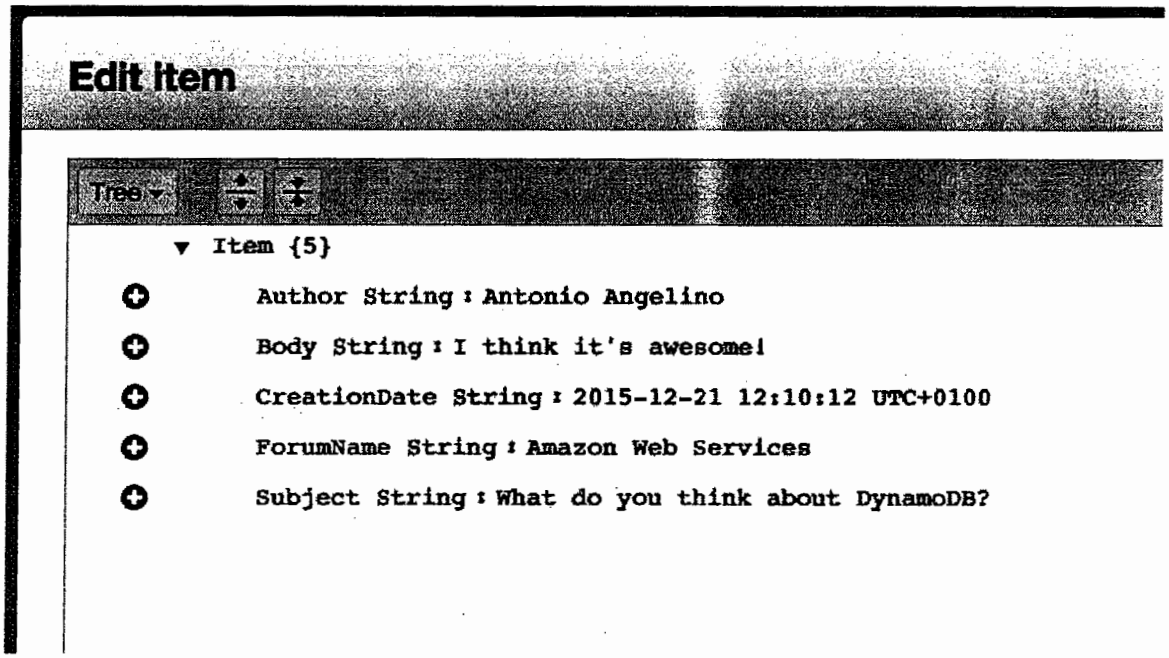
If you are not still browsing the items of an existing table, select one of them from the DynamoDB dashboard and then click on the **Items** tab pane. It lists all the existing table items.

For editing an item, select it by checking the left checkbox and then click the **Edit** button in the**Actions** drop-down menu.

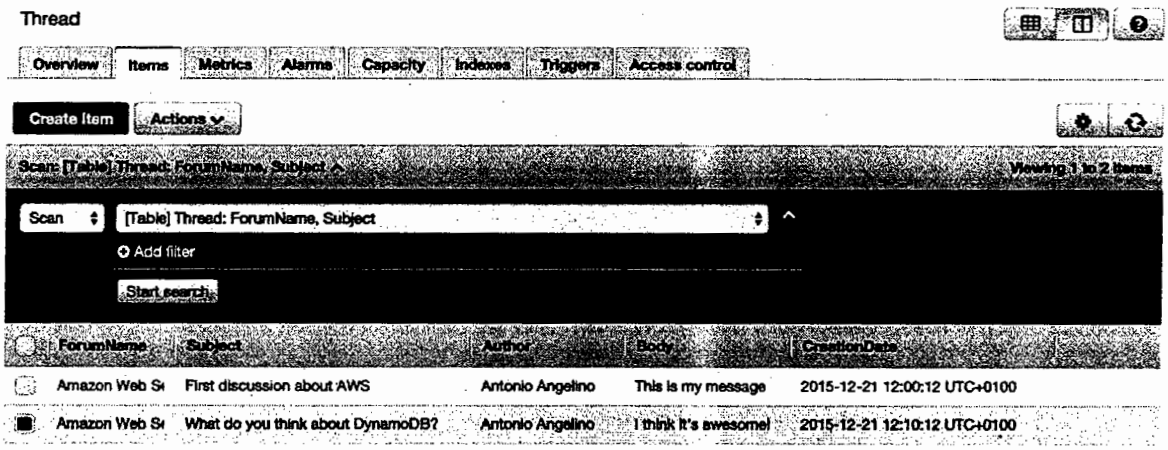


You can edit the item fields or add new data exactly as you did during the creation process.

By editing the Thread items, you may add the **Author** field or change Thread's Subject.



When you finish editing, you can modify by clicking the **Save** button.

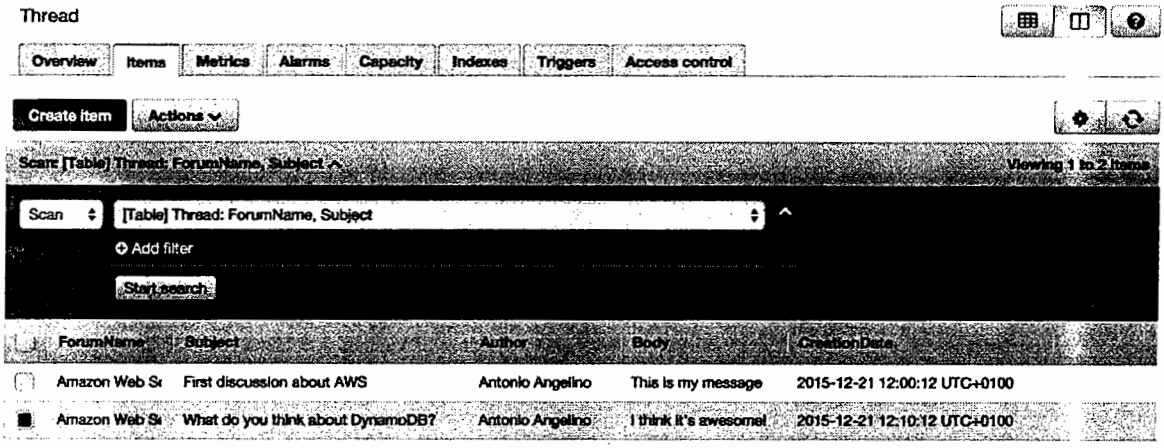


STEP 6: Query a DynamoDB table

DynamoDB provides two commands for searching data on the table: **Scan** and **Query**.

If you are not still browsing the items of an existing table, select one of them from the DynamoDB dashboard and click on the **Items** tab pane.

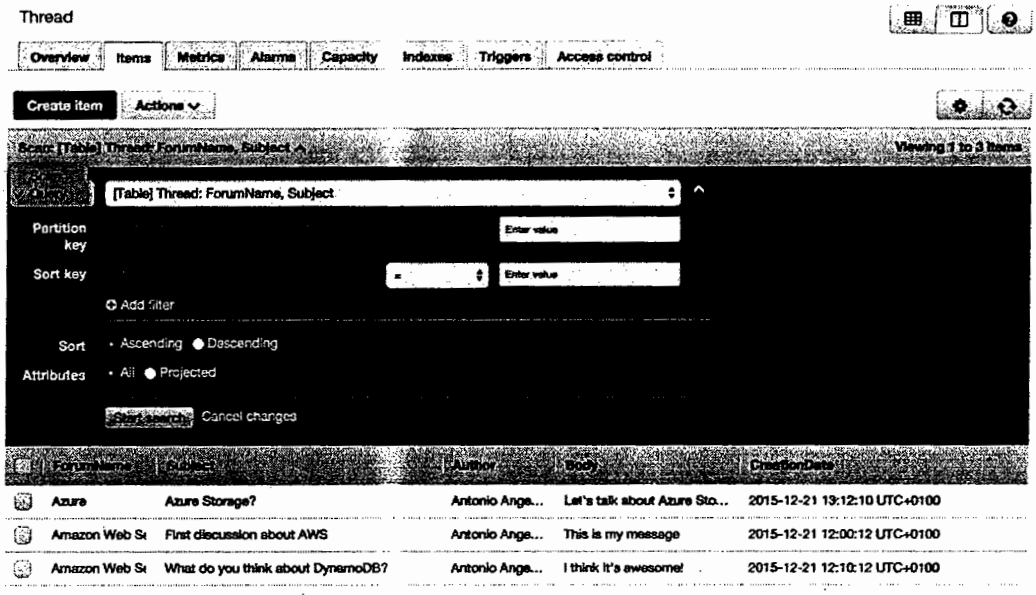
A **Scan** operation examines every item on the table and returns all the data attributes for each one of them. The Amazon DynamoDB Explore Table executes a Scan query by default.



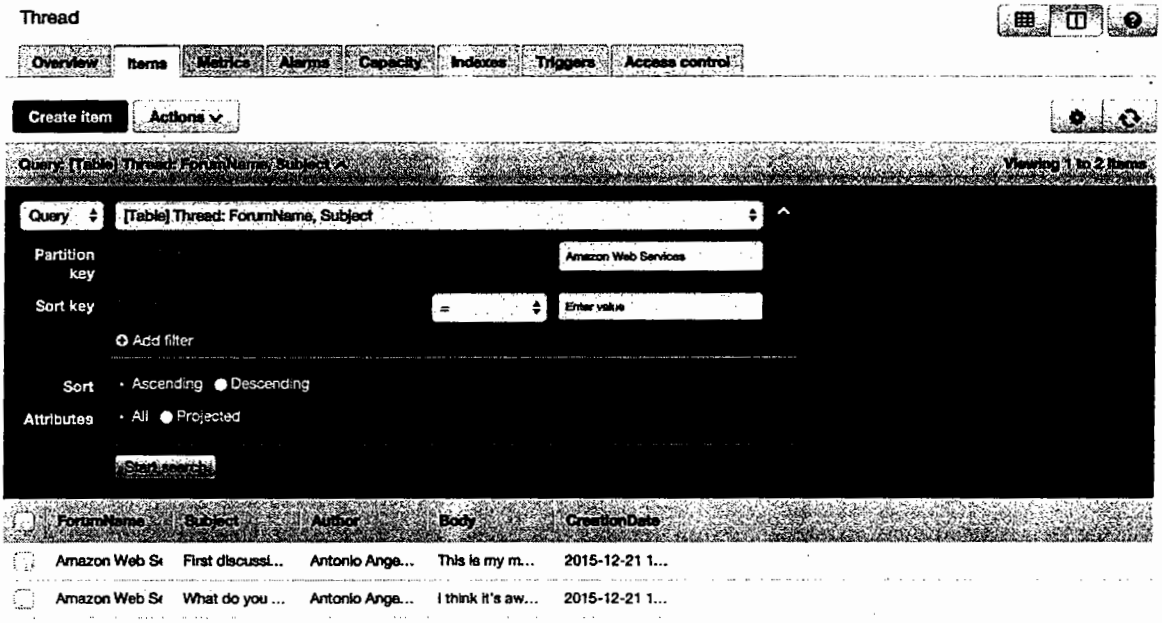
A **Query** operation finds items in a table using only primary key attribute values. You must provide a hash key attribute name and a distinct value to search for. You can optionally provide a range key attribute name and value, and use a comparison operator to refine the search results. Query supports a specific set of comparison operators for choosing key values. You must specify the hash key attribute name and value as an equality condition.

You previously created the Thread table with a Hash and Range Keys and the CreationDate-index Secondary Local Index. The Amazon DynamoDB Management Console allows you to use the primary index and the secondary one to execute a Query operation.

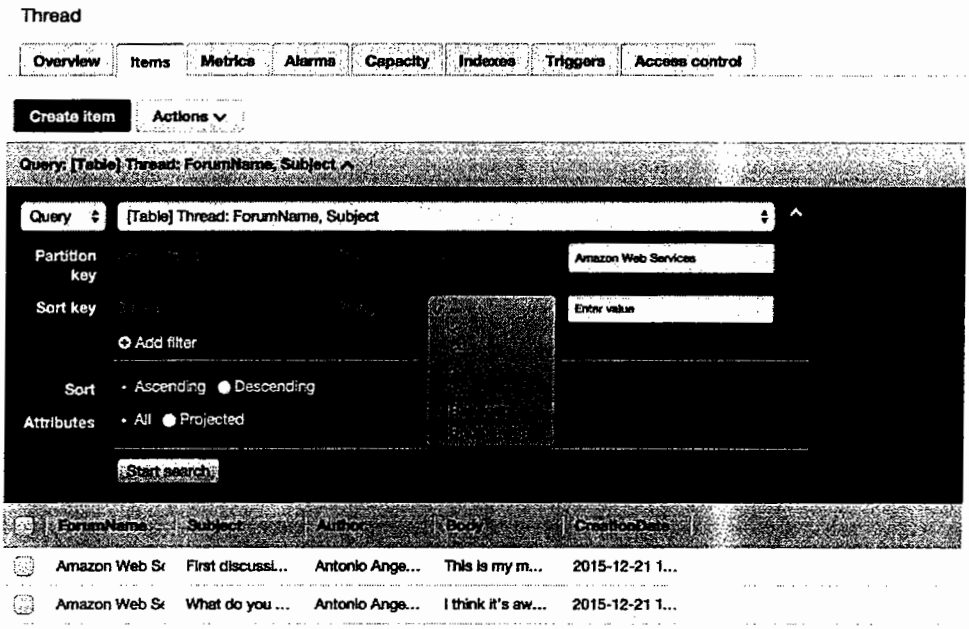
To begin querring the Thread, insert an existing ForumName value and click on the **Query**button.



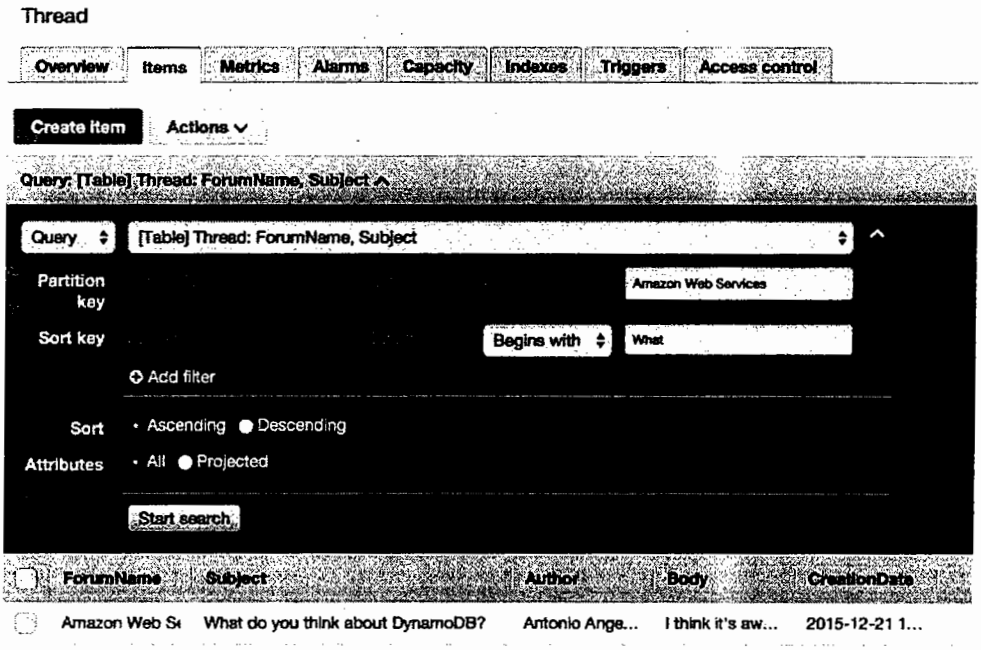
Query results are always sorted by the range key (Subject). If the data type of the range key is Number, the results are returned in numeric order; otherwise, the results are returned in order of ASCII character code values. By default, the sort order is ascending. To reverse the order, you can use the **Sort** radio button.



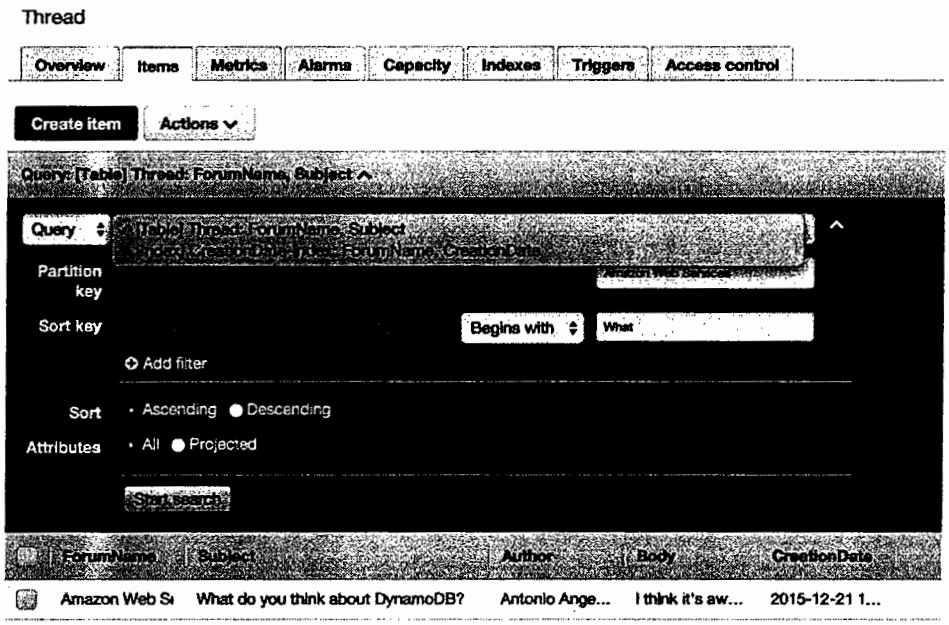
You can optionally specify a second condition, referring to the range key attribute. This condition allows you to choose from several conditional operators: equal to, less than, greater than, between, begins with and so on.



You can find all the table items with a specific ForumName and the Subject that *begins with* a specific word in a few seconds.

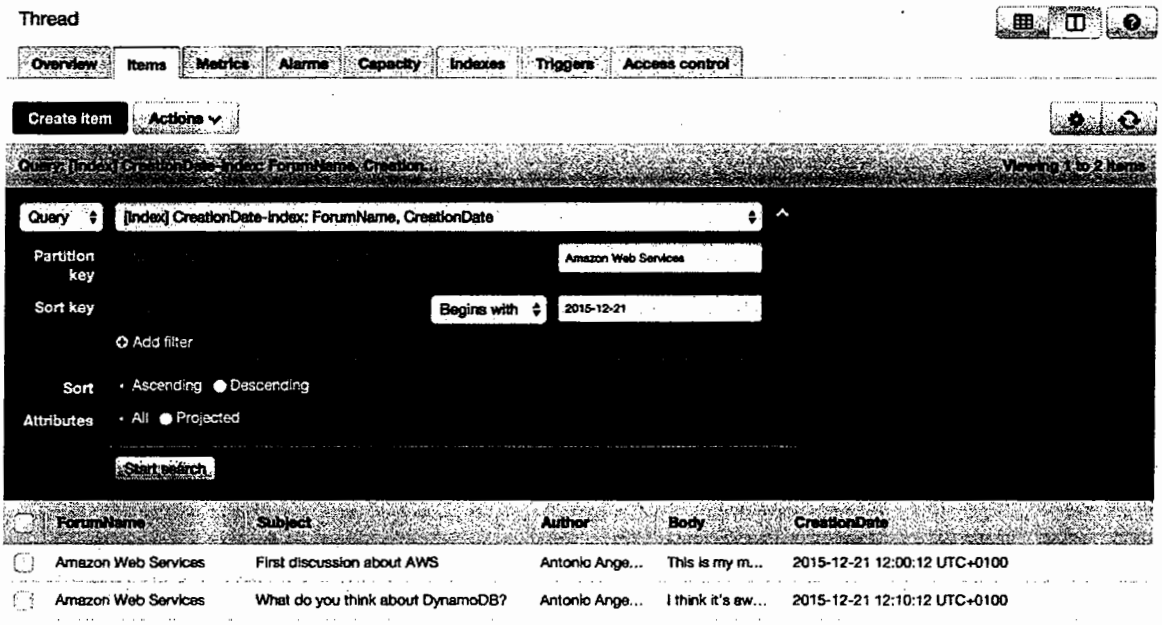


In order to use a Local or Global Secondary Index, simply switch the Index Name using the provided select box.



Fill the Hash and Range Key with the proper data and then hit the **Query** button.

You can also decide not to display all the attributes by selecting the **Projected** radio button and choosing which attributes should appear in the result set.

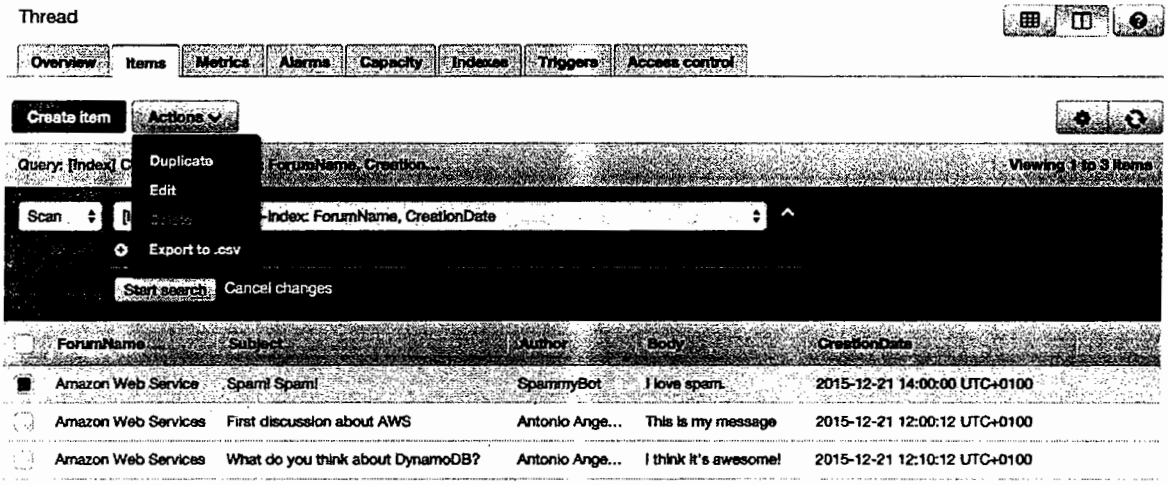


STEP 7: Delete a DynamoDB table item

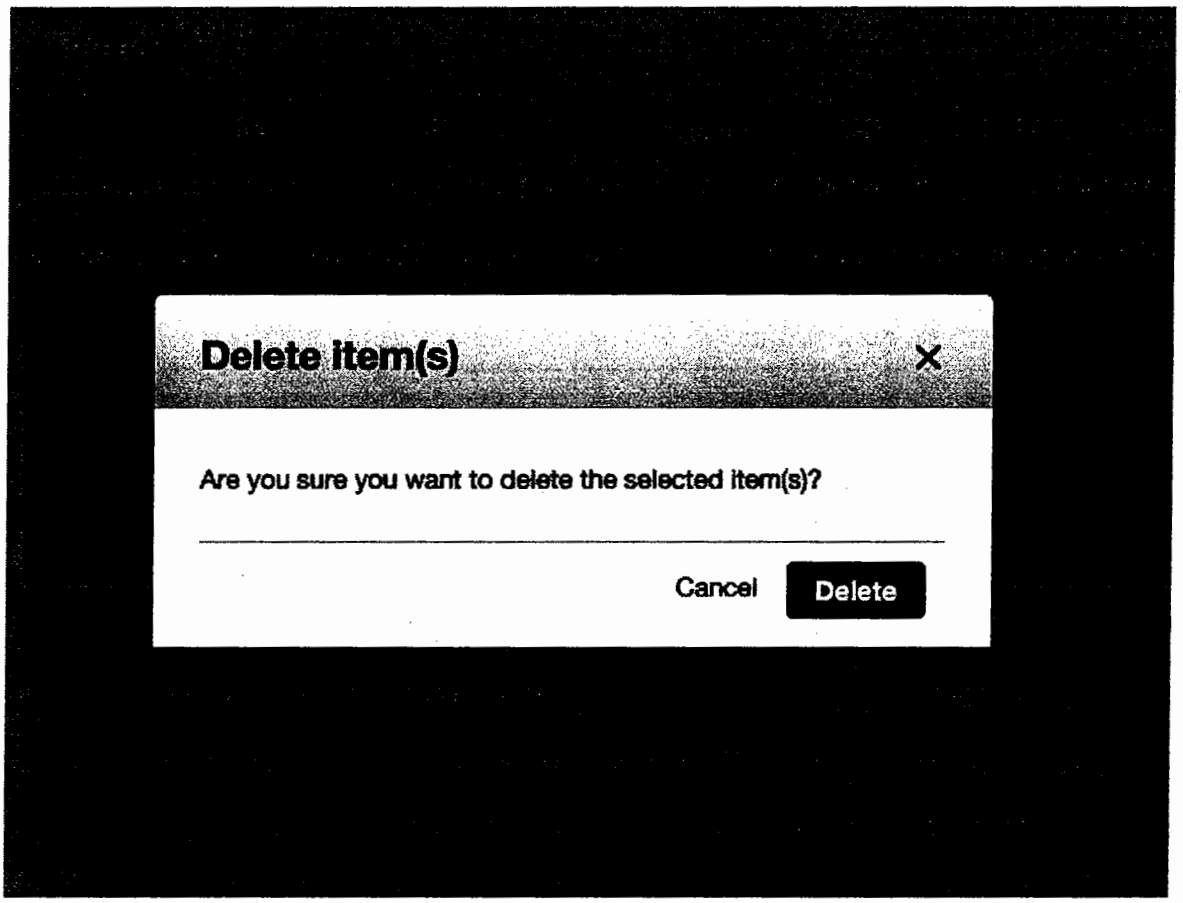
Deleting a DynamoDB table item is task ancan be done using the **Items** tab pane of a DynamoDB Table.

If you are not still browsing the items of an existing table, select one of them from the DynamoDB dashboard and then click on the **Items** tab pane.

For deleting an item, select it by checking the left checkbox and then select the **Delete action**from the **Action** drop-down menu.



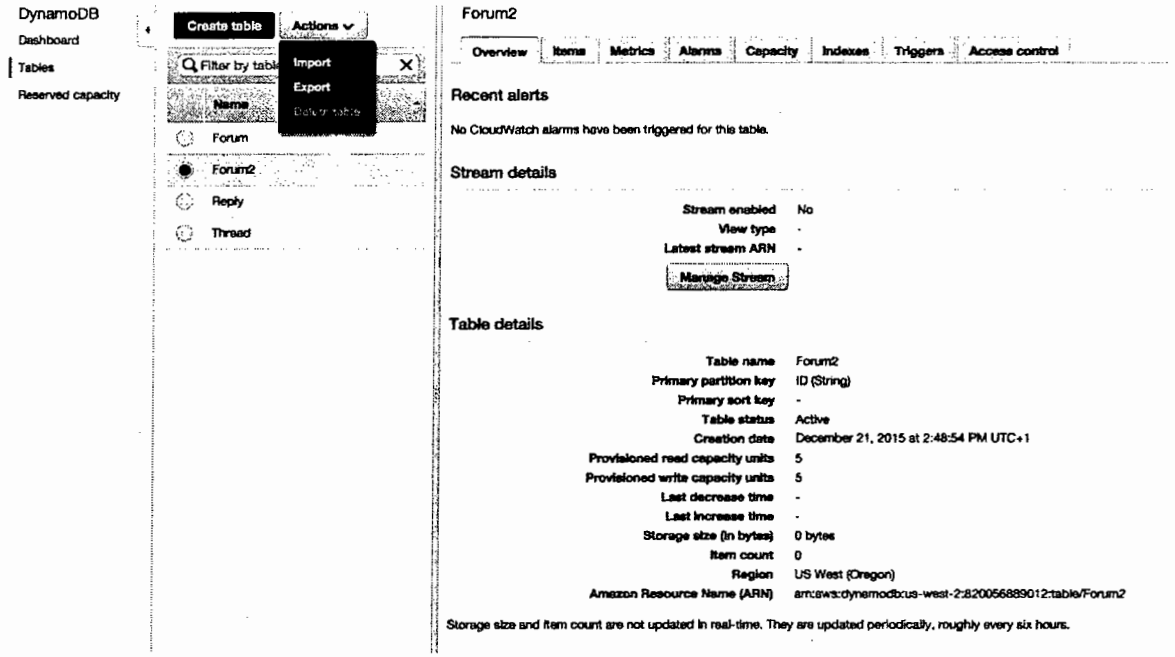
The AWS Management Console will ask you to confirm the operation. Click **Delete** and the selected item will be permanently deleted.



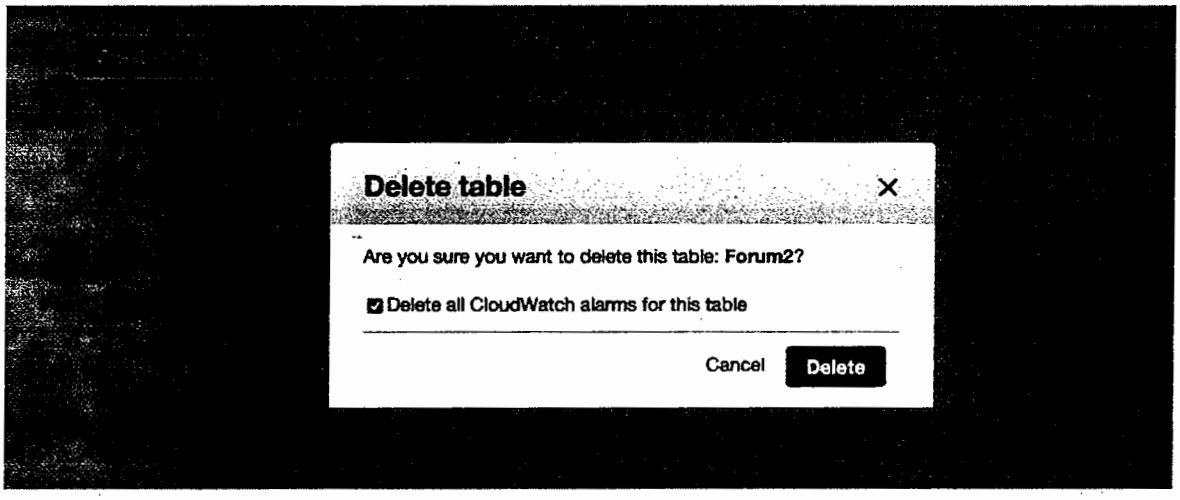
STEP 8: Delete a DynamoDB table

You can delete a DynamoDB table any time from the DynamoDB dashboard in the AWS Management Console.

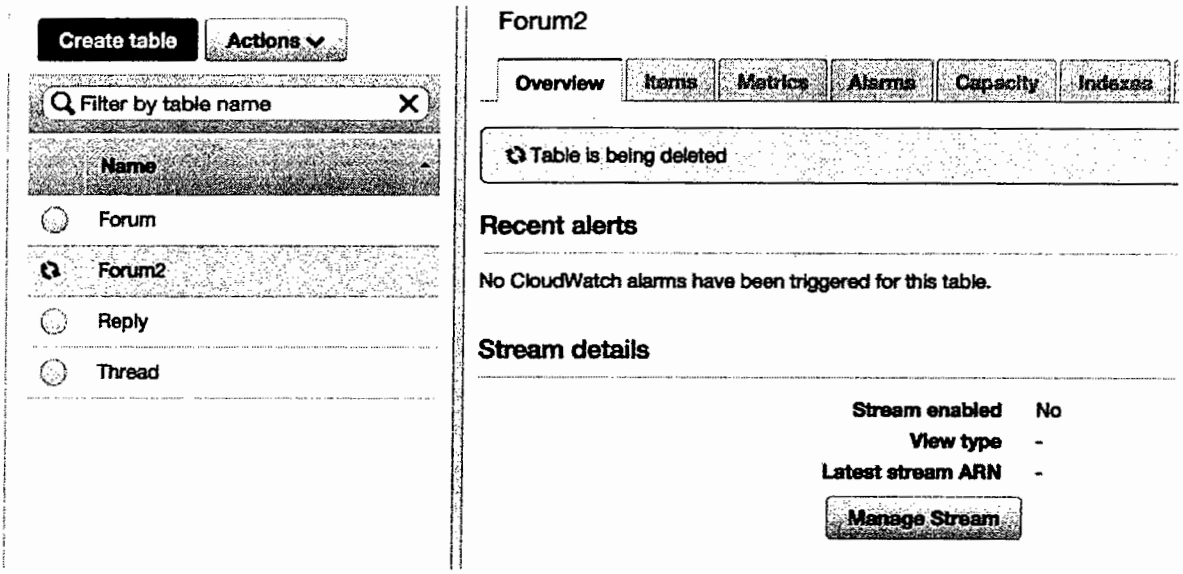
For deleting an entire table, click on it, then select the **Delete table** action from the**Actions** drop-down menu.



During the table deletion process, you can choose to delete the table and all its related services (CloudWatch alarms and import/export pipelines) or just some of them.



After clicking the delete button, the table changes its status to "deleting" and it will shortly disappear from the DynamoDB table list.


Create table
Actions
Filter by table name
Name
Forum
Forum2
Reply
Thread
Forum2
Overview
Items
Metrics
Alarms
Capacity
Indexes
Table is being deleted
Recent alerts
No CloudWatch alarms have been triggered for this table.
Stream details
Stream enabled
No
View type
-
Latest stream ARN
-
Manage Stream