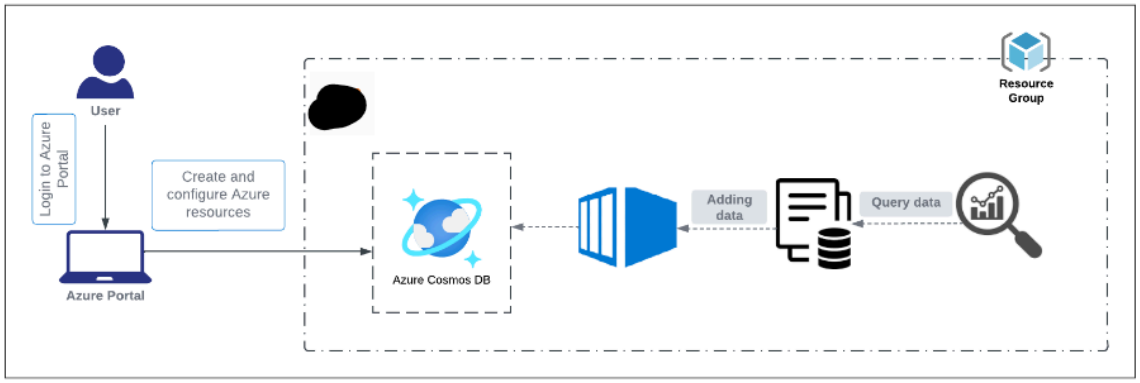
Introduction

What is Azure Cosmos DB ?

- Azure Cosmos DB is a fully managed NoSQL database that can be used for application development.
- It is highly scalable with very low latency.
- Management of Cosmos DB including updating and patching is done automatically without any manual intervention.
- With the help of Cosmos DB SQL API, one can query JSON files using SQL.
- Also, since SQL is quite popular among developers, it becomes pretty easy to query data in Cosmos DB.
- While storing the data in the Cosmos DB database, the data is split into logical partitions, thus making it easy to query the data.

Architecture Diagram



Lab Steps

Task 1: Sign in to Azure Portal

1. Go to the Azure portal by clicking on the **Open Console** button or by using URL https://portal.azure.com.

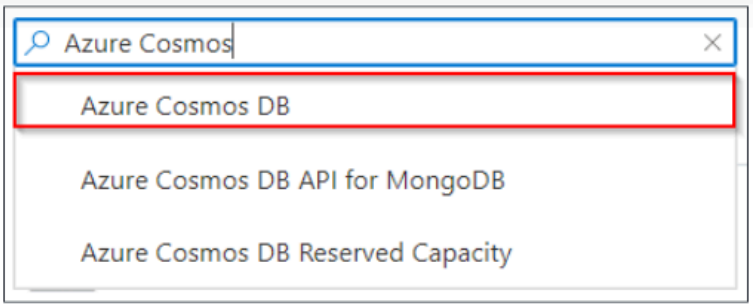
    - **Note**: It is recommended to use incognito mode to avoid Azure portal cache related issues.

2. If it automatically logs into any other azure account, please logout of it and clear cache.
3. Sign in with your given ***username*** and ***password*** on Azure portal.
4. If login is not working. Click on **End Lab** and start the lab again.

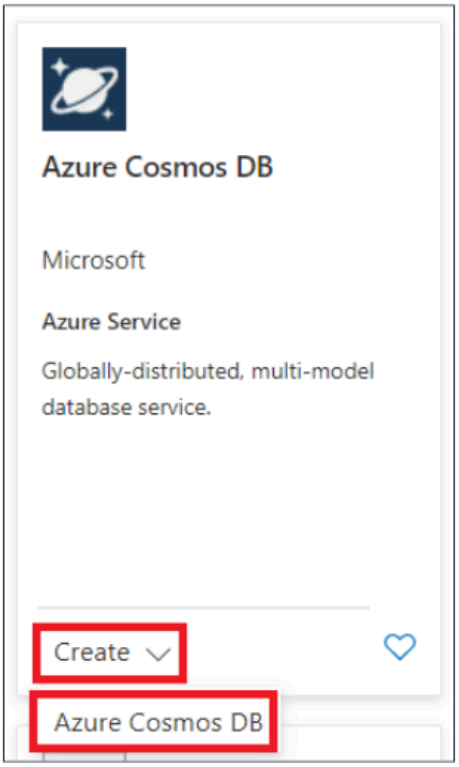
Task 2: Creating an Azure Cosmos DB Account

1. The Azure portal menu or from the Home page, select **Create a resource**.

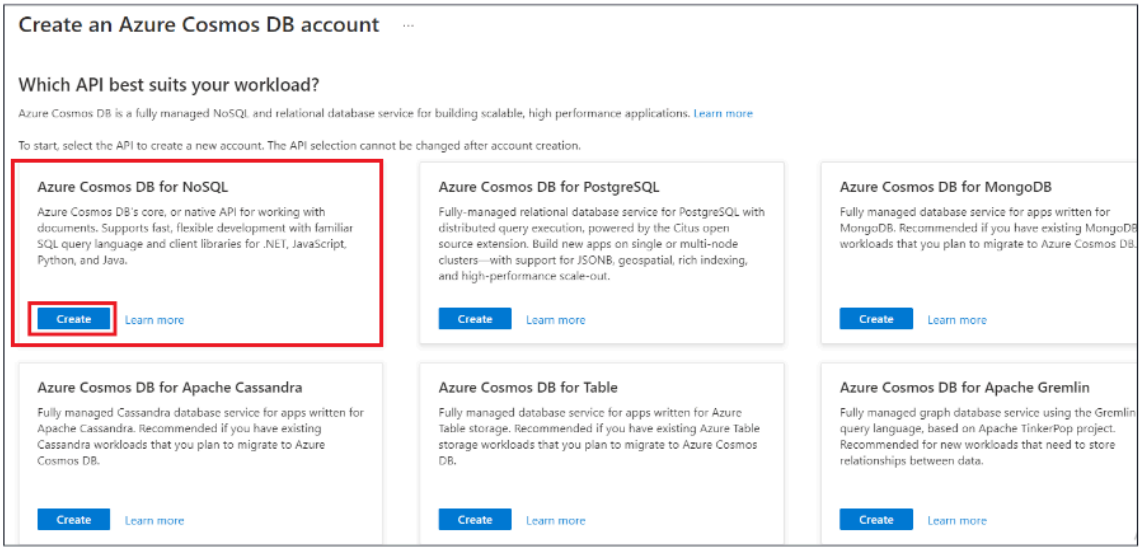
2. On the search bar, enter Azure Cosmos and then select **Azure Cosmos DB.**



3. Click on **Create.**

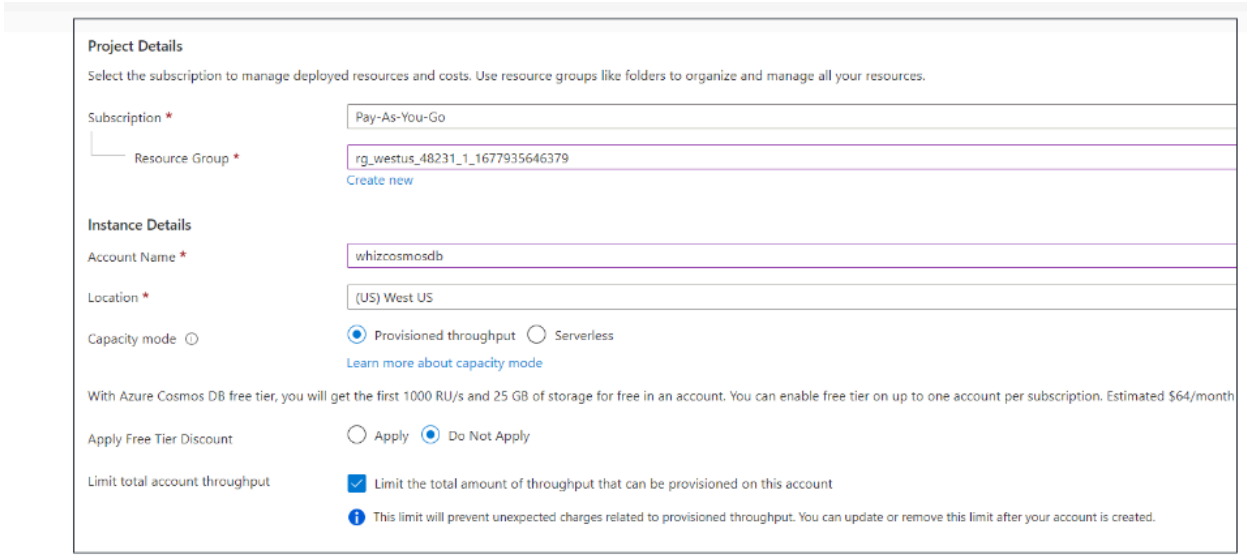


4. On the **Create an Azure Cosmos DB account** page, click on **Create** under **Azure Cosmos DB for NoSQL.**

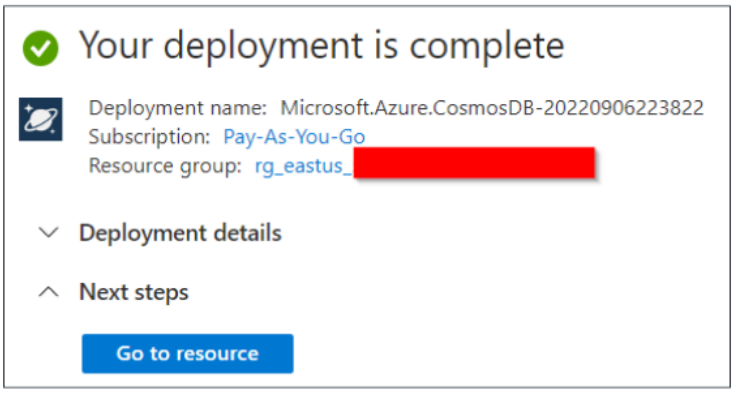


5. Now, fill out the page with the following details:

- Resource group: Select **rg_westus_XXXXX**
- Account Name: Enter ***whizcosmosdb***
- Apply Free Tier Discount: Select **Do Not Apply**

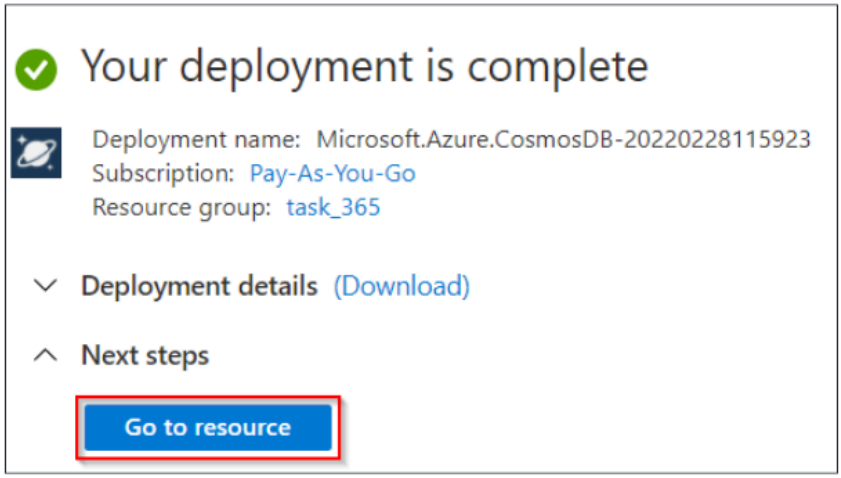


6. Keeping rest in default, click on **Review+Create** and then click on **Create.**

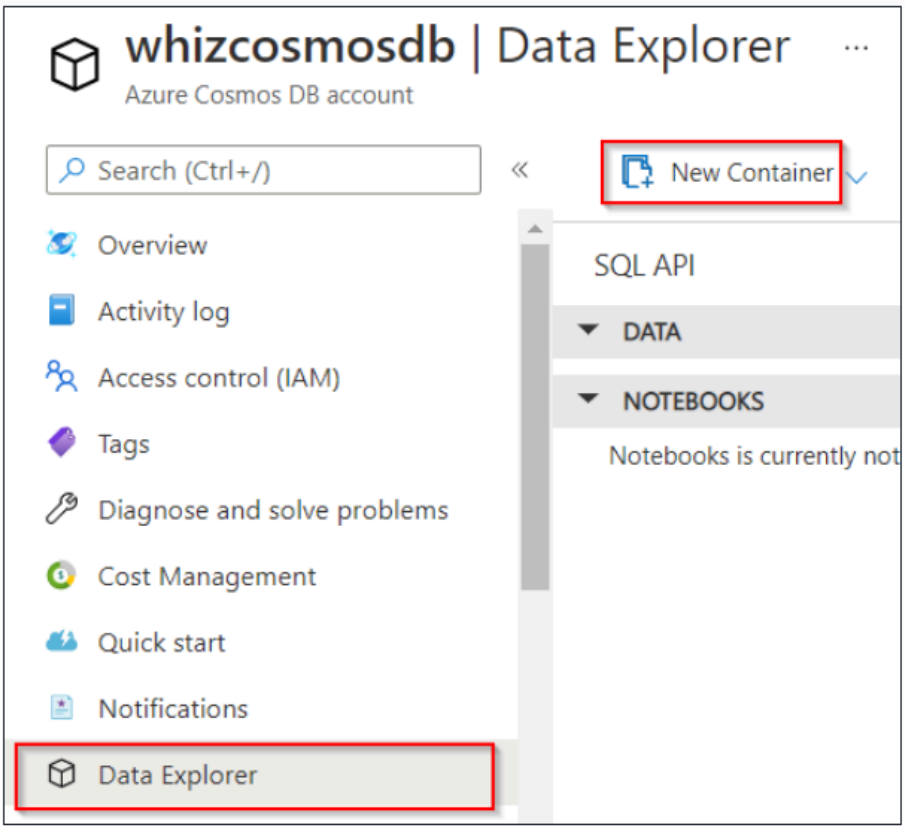


Task 3: Creating container within Cosmos DB Account

1. After successful deployment, click on **Go to Resource.**

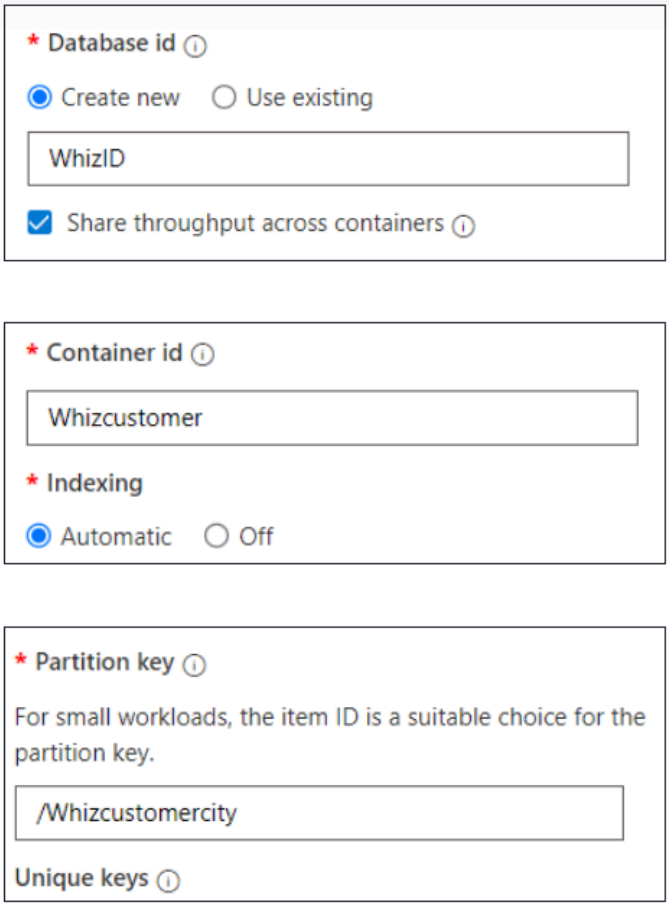


2. On the left panel, click on **Data Explorer** and then click on **New Container**.



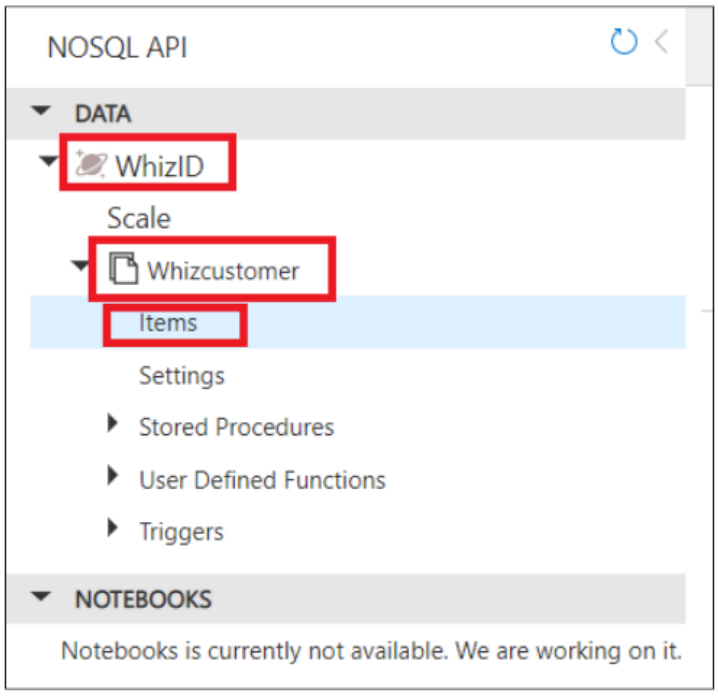
3. Fill the form with the following information:

- Database id: Enter ***WhizID***
- Container id: Enter ***Whizcustomer***
- Partition key: Enter ***/Whizcustomercity***
- Click on **Ok.**



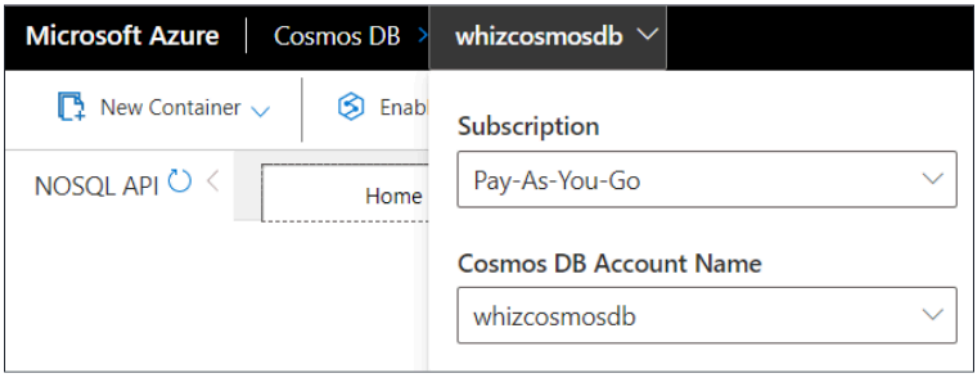
Task 4: Adding default data into database

1. After the successful creation of the container/database, click on **WhizID** (Database ID) and then click on **Whizcustomer** and then on **items**.

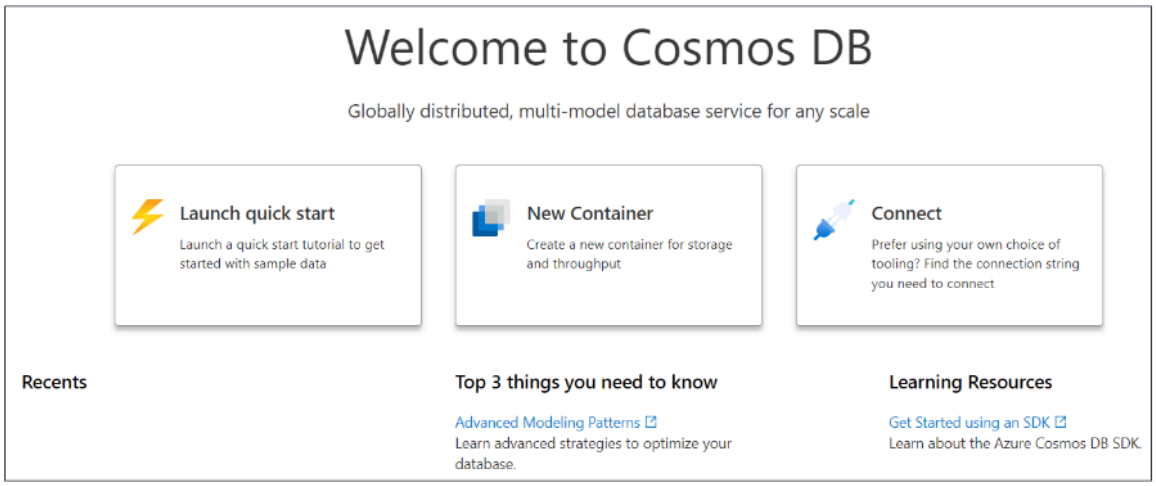


2. On the top right corner, click on the Open Full Screen option.

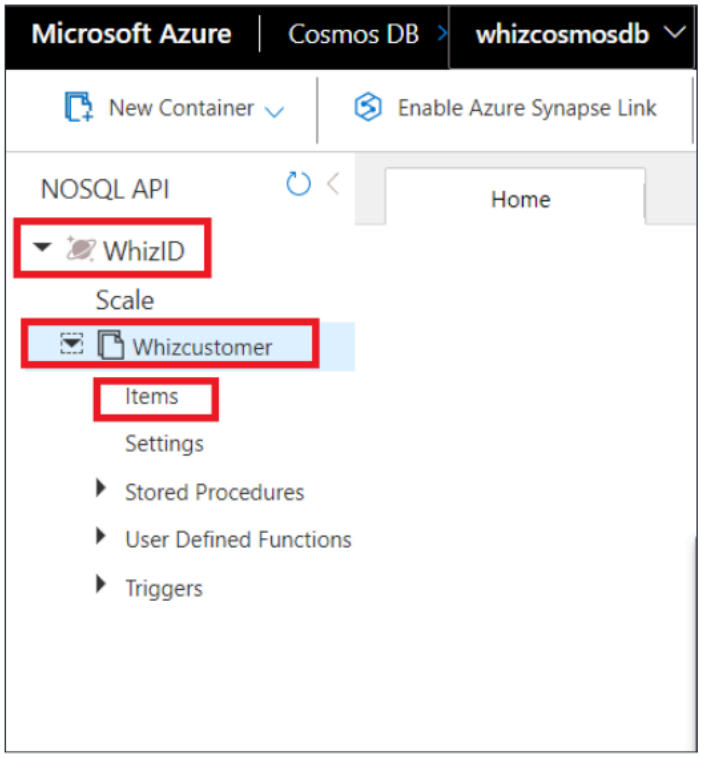
3. Click on **Open** and click on **Sign In**. Then select your **Subscription** and **Cosmos DB Account Name**.



4. You will see a page like below.



5. Now, in the new tab, click on **WhizID** (Database ID) and then click on **Whizcustomer** and then on **items**.

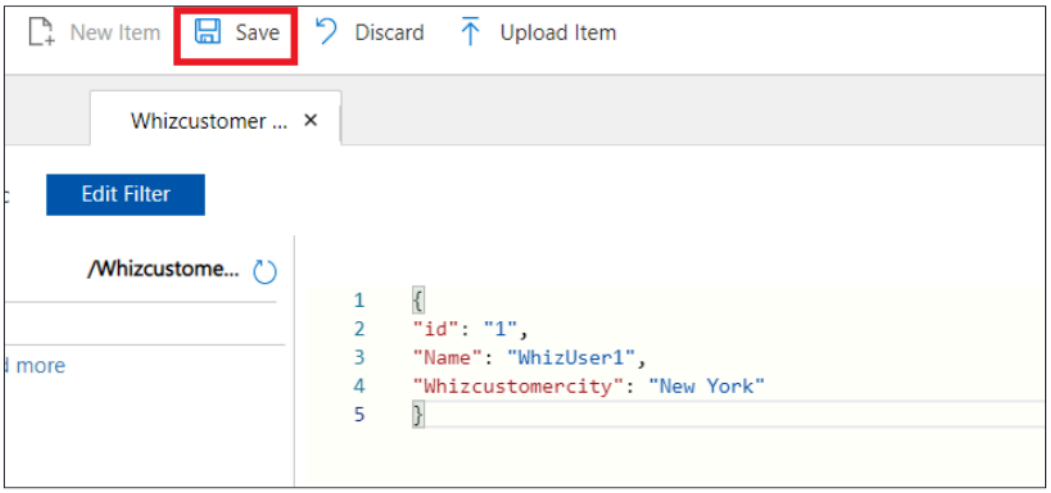


6. On the top panel, click on **New Item.**

7. Delete the text that is already displayed and then enter the following data:

```
{
    "id": "1",
    "Name": "WhizUser1",
    "Whizcustomercity": "New York"
}
```

8. Click on **Save.**



9. Again click on **New Item** and delete the data that is already displayed and then enter the following data:

```
{
    "id": "2",
    "Name": "WhizUser2",
    "Whizcustomercity": "Atlanta"
}
```

10. Click on **Save.**

Task 5: Querying the added data

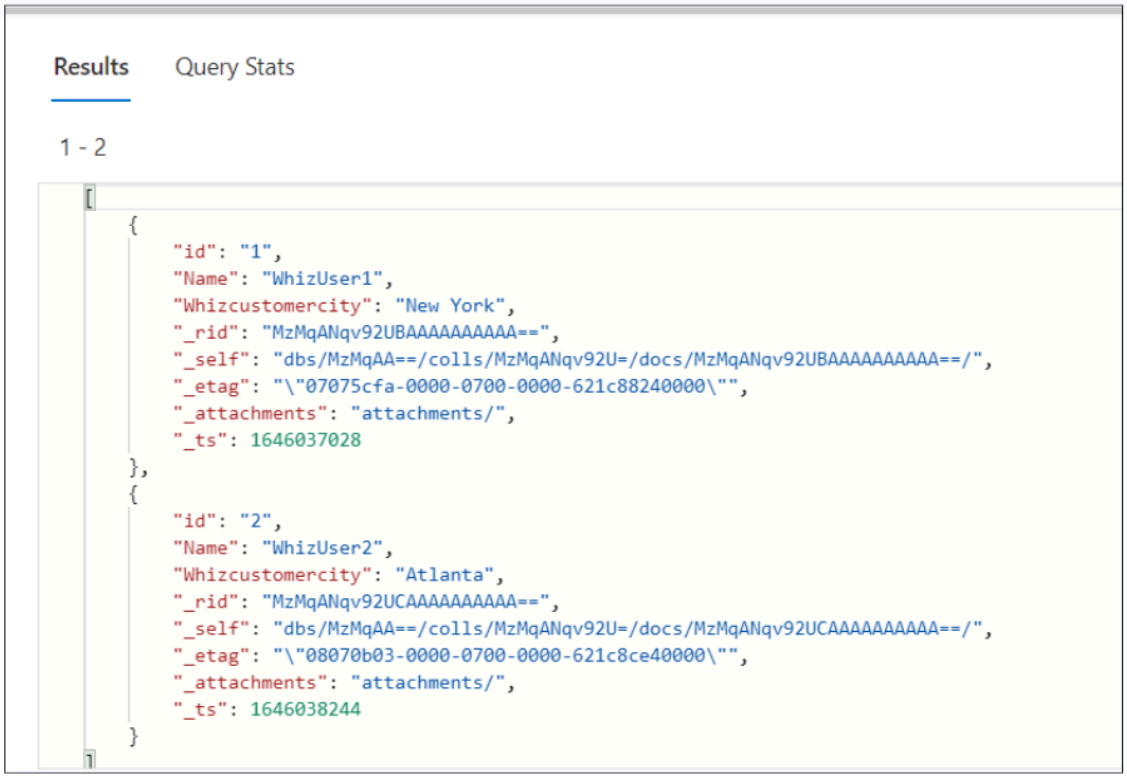
1. After the data is successfully added in the database, click on **New SQL Query** on the top panel to use the query on the saved data.

2. Enter the following query to display all the records stored in the database:

```
SELECT * FROM c
```

3. Click on **Execute Query** on the top panel.

4. All the saved data would be displayed on the screen.

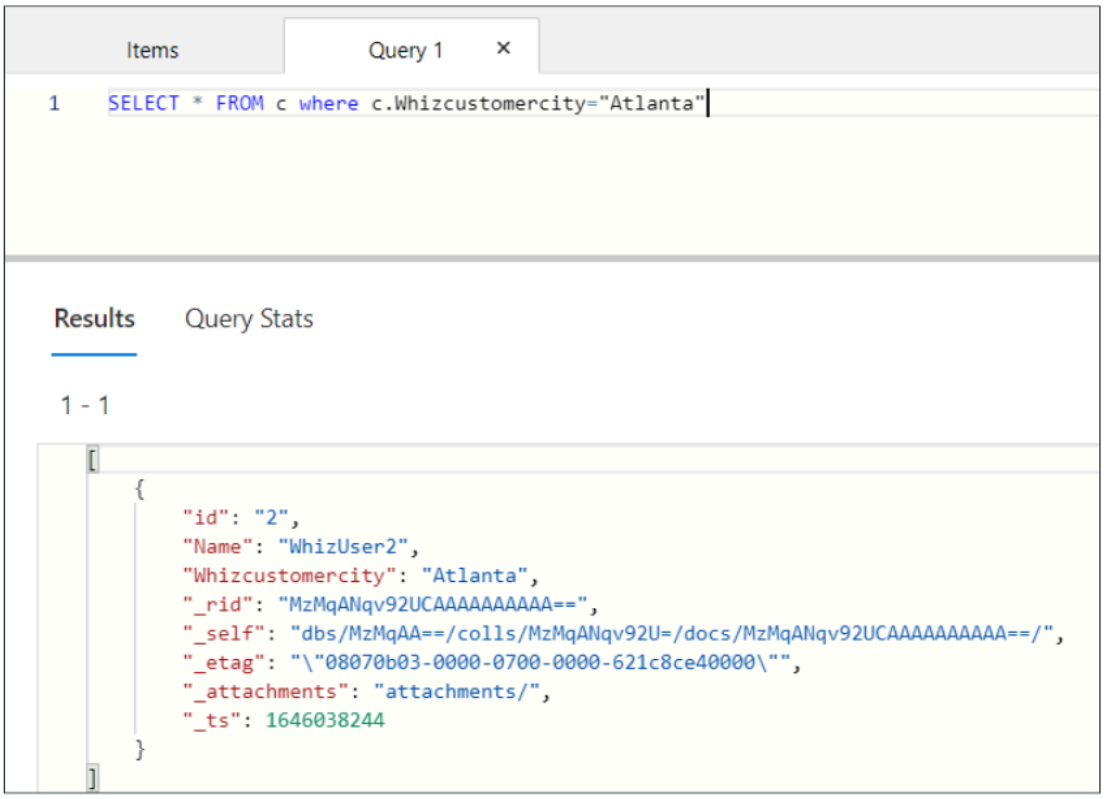


5. To display records where Whizcustomercity="Atlanta", enter the following command:

```
SELECT * FROM c where c.Whizcustomercity="Atlanta"
```

6. Click on **Execute Query** on the top panel.

7. All the data that satisfies the condition will be displayed.



Completion and Conclusions

1. You have successfully logged into Azure Portal.
2. You have successfully created an Azure Cosmos DB Account.
3. You have successfully created a container within Cosmos DB Account
4. You have successfully added default data into database
5. You have successfully queried the added data.
6. You have successfully tested the validation.
7. You have successfully deleted the resources.