Online Lab - Comparing Database Options in Azure

Topic: Deploying Database Instances in Azure

Before we start

1. Ensure that you are logged in to your Windows 10 lab virtual machine using the following credentials:
    - Username: **Admin**
    - Password: **Pa55w.rd**
2. Review Taskbar located at the bottom of your Windows 10 desktop. The Taskbar contains the icons for the common applications you will use in the labs:
    - Microsoft Edge
    - File Explorer
    - Visual Studio Code
    - Microsoft Azure Storage Explorer
    - Bash on Ubuntu on Windows
    - Windows PowerShell

      Note: You can also find shortcuts to these applications in the **Start Menu**.

Exercise 1: Deploy a Cosmos DB database

Task 1: Open the Azure Portal

1. On the Taskbar, click the **Microsoft Edge** icon.
2. In the open browser window, navigate to the **Azure Portal** (https://portal.azure.com).
3. If prompted, authenticate with the user account account that has the owner role in the Azure subscription you will be using in this lab.

Task 2: Create a Cosmos DB database and collection

1. In the upper left corner of the Azure portal, click **Create a resource**.
2. At the top of the **New** blade, in the **Search the Marketplace** text box, type **Cosmos DB** and press **Enter**.
3. On the **Everything** blade, in the search results, click **Azure Cosmos DB**.

4. On the **Azure Cosmos DB** blade, click the **Create** button.
5. From the new **Create Azure Cosmos DB Account** blade, create a new Azure Cosmos DB account with the following settings:

    - Subscription: the name of the Azure subscription you are using in this lab.
    - Resource Group: the name of a new resource group **AADesignLab0701-RG**.
    - Account Name: any globally unique name.
    - API: **Core (SQL)**
    - Location: the name of the Azure region in which you want to deploy resources in this lab.
    - Geo-Redundancy: **Disable**
    - Multi-region Writes: **Disable**
    - Virtual Network: the name of a new virtual network az30107-vnet with the default address space, subnet name, and subnet address range

6. Wait for the provisioning to complete before you proceed to the next task

Note: The deployment should take less than 5 minutes.

7. In the hub menu in the Azure portal, click **Resource groups**.

8. On the **Resource groups** blade, click **AADesignLab0701-RG**.

9. On the **AADesignLab0701-RG** blade, click the entry representing the newly created **Azure Cosmos DB** account.

10. On the Azure Cosmos DB account blade, click the warning **Azure Portal access to the data is being blocked by your Firewall configuration**, on the **Firewall** blade, click the **All networks** option, save the change. Wait for the change to take effect, close the **Firewall** blade and return to the **Overview** section of the Azure Cosmos DB account blade.

11. At the top of the Azure Cosmos DB account blade, click the **Add Collection** button.

12. In the **Add Collection** popup, perform the following tasks:

- In the **Database id** text box, type **FinancialClubDatabase**.
- In the **Collection id** text box, type **MemberCollection**.
- In the **Storage capacity** section, select the **Fixed (10 GB)** option.
- In the **Throughput** text box, type **400**.
- Click the **OK** button.

13. On the left side of the Azure Cosmos DB account blade, click the **Keys**.

14. In the **Keys** pane, record the values in the **URI** and **PRIMARY KEY** fields. You will use these values later in this lab.

Task 3: Create and query documents in Cosmos DB

1. On the left side of the Azure Cosmos DB account blade, click **Data Explorer**.
2. In the **Data Explorer** pane, click the **MemberCollection** child node of the **FinancialClubDatabase** node.
3. Click the **New SQL Query** button at the top of the **Data Explorer** pane.
4. In the **Query 1** tab that opened, view the default query:

```
SELECT * FROM c
```

5. Click the **Execute Query** button at the top of the query editor.
6. In the left pane of the Data Explorer, expand the **MemberCollection** node.
7. Click the **Documents** child node within the **MemberCollection** node.
8. In the new **Documents** tab that opened, click the **New Document** button at the top of the tab.
9. In the **Documents** tab, replace the existing document with the following document:

```
{ "firstName": "Pennington", "lastName": "Oneal", "age": 26, "salary":
90000.00, "company": "Veraq", "isVested": false }
```

10. Click the **Save** button at the top of the **Documents** tab.
11. In the **Documents** tab, click the **New Document** button at the top of the tab.
12. In the **Documents** tab, replace the existing document with the following document:

```
{ "firstName": "Suzanne", "lastName": "Oneal", "company": "Veraq" }
```

13. Click the **Save** button at the top of the **Documents** tab.
14. Switch back to the **Query 1** tab, re-run the default query `SELECT * FROM c` by clicking the **Execute Query** button at the top of the query editor, and review the results.
15. In the query editor, replace the default query with the following query:

```
SELECT c.id, c.firstName, c.lastName, c.isVested, c.company FROM c
WHERE IS_DEFINED(c.isVested)
```

16. Click the **Execute Query** button at the top of the query editor and review the results.
17. In the query editor, replace the existing query with the following query:

```
SELECT c.id, c.firstName, c.lastName, c.age FROM c WHERE c.age > 20
```

18. Click the **Execute Query** button at the top of the query editor and review the results.
19. In the query editor, replace the existing query with the following query:

```
SELECT VALUE c.id FROM c
```

20. Click the **Execute Query** button at the top of the query editor and review the results.
21. In the query editor, replace the existing query with the following query:

```
SELECT VALUE { "badgeNumber": SUBSTRING(c.id, 0, 8), "company":
c.company, "fullName": CONCAT(c.firstName, " ", c.lastName) } FROM c
```

22. Click the **Execute Query** button at the top of the query editor and review the results.

Review: In this exercise, you created a new Cosmos DB account, database, and collection, added sample documents to the collection, and run sample queries targeting these documents.

Exercise 2: Deploy Application using Cosmos DB

Task 1: Deploy API App code using Azure Resource Manager templates and GitHub

1. In the upper left corner of the Azure portal, click **Create a resource**.
2. At the top of the **New** blade, in the **Search the Marketplace** text box, type **Template Deployment** and press **Enter**.
3. On the **Everything** blade, in the search results, click **Template Deployment**.
4. On the **Template deployment** blade, click the **Create** button.
5. On the **Custom deployment** blade, click the **Build your own template in the editor** link.
6. On the **Edit template** blade, click the **Load file** link.
7. In the **Open** file dialog that appears, navigate to the **F:\Mod07\Labfiles\Starter** folder.
8. Select the **api.json** file.
9. Click the **Open** button.

10. Back on the **Edit template** blade, click the **Save** button to persist the template.
11. Back on the **Custom deployment** blade, perform the following tasks:
    - Leave the **Subscription** drop-down list entry set to its default value.
    - In the **Resource group** section, select the **Use existing** option and then, in the drop-down list, select **AADesignLab0701-RG**.
    - In the **Terms and Conditions** section, click the **I agree to the terms and conditions stated above** checkbox.
    - Click the **Purchase** button.
12. Wait for the deployment to complete before you proceed to the next task.

    Note: Deployment from source control can take up to 10 minutes.

Task 2: Validate API App

1. In the hub menu in the Azure portal, click **Resource groups**.
2. On the **Resource groups** blade, click **AADesignLab0701-RG**.
3. On the **AADesignLab0701-RG** blade, click the entry representing the newly created App Service API app.
4. On the API app blade, click **Application Settings**.
5. On the Application Settings blade, scroll down to the **Application settings** section and perform the following tasks:
    - Set the value of the **CosmosDB:AuthorizationKey** setting to the value of the **PRIMARY KEY** setting of the **Cosmos DB** account you created earlier in this lab.
    - Update the value of the **CosmosDB:EndpointUrl** setting to the value of the **URI** setting of the **Cosmos DB** instance you created earlier in this lab.
    - Click the **Save** button at the top of the pane.
6. On the left-side of the API app blade, click **Overview**.
7. Click the **Restart** button at the top of the blade and, when prompted to confirm, click **Yes**.
8. Click the **Browse** button at the top of the blade. This will open a new browser tab displaying the **Swagger UI** homepage.

    Note: If you click the **Browse** button before the API app has fully restarted, you may not be able to follow the remaining steps in this task. If this happens, refresh your browser until the API app is running again.

9. On the **Swagger UI** homepage, click **GET/Documents**.
10. Click the **Try it out!** button.
11. Review the results of the request.
12. Back on the **Swagger UI** homepage, click **POST/Populate**.

13. In the **Parameters** section, in the **Value** field for the **options** parameter, paste in the following JSON content:

    ```
    { "quantity": 50 }
    ```

14. In the **Response Messages** section, click the **Try it out!** button.
15. Review the results of the request.
16. Back on the **Swagger UI** homepage, click **GET/Documents**.
17. Locate the **Response Content Type** section. Click the **Try it out!** button.
18. Review the results of the request.
19. Close the new browser tab and return to the browser tab displaying the Azure portal.

Review: In this exercise, you created a new API App that uses the .NET Core DocumentDB SDK to connect to Azure Cosmos DB collection and manage its documents.

Exercise 3: Connect Cosmos DB to Azure Search

Task 1: Create Azure Search Instance

1. In the upper left corner of the Azure portal, click **Create a resource**.
2. At the top of the **New** blade, in the **Search the Marketplace** text box, type **Search** and press **Enter**.
3. On the **Everything** blade, in the search results, click **Azure Search**.
4. On the **Azure Search** blade, click the **Create** button.
5. On the **New Search Service** blade, perform the following tasks:
    - In the **URL** text box, enter a globally unique name. Record its value. You will use it later in this lab.
    - Leave the **Subscription** drop-down list entry set to its default value.
    - In the **Resource group** section, select the **Use existing** option and then, in the drop-down list, select **AADesignLab0701-RG**.
    - In the **Location** drop-down list, select the Azure region matching or near the location where you deployed Cosmos DB resource earlier in this labb
    - Click **Pricing tier**.
    - On the **Choose your pricing tier** blade, click **Free** and then click the **Select** button.
    - Click the **Create** button.
6. Wait for the provisioning to complete before you proceed to the next step.
7. In the hub menu in the Azure portal, click **Resource groups**.
8. On the **Resource groups** blade, click **AADesignLab0701-RG**.

9. On the **AADesignLab0701-RG** blade, click the entry representing the newly created Azure Search instance.
10. On the Search service blade, click **Keys**.
11. In the **Keys** pane, record the value of **PRIMARY ADMIN KEY**. You will use it later in this lab.

Task 2: Index Cosmos DB Data in Azure Search

1. In the hub menu in the Azure portal, click **Resource groups**.
2. On the **Resource groups** blade, click **AADesignLab0701-RG**.
3. On the **AADesignLab0701-RG** blade, click the entry representing the Azure Cosmos DB account you created earlier in this lab.
4. On the Azure Cosmos DB account blade, click **Add Azure Search**.
5. On the **Import data** blade, click **Search service** and, on the **Search services** blade, click the newly created Azure Search Service instance.

   Note: You will be presented with two blades: the **Data Source** blade with the **CosmosDB** option already selected and with the **New data source** blade.

6. On the **New data source** blade, perform the following tasks:
   - In the **Name** text box, type **cosmosdata**.
   - In the **Connection string** text box, accept the default entry.
   - In the **Database** drop-down list, select the **FinancialClubDatabase** entry.
   - in the **Collection** drop-down list, select the **MemberCollection** entry.
   - In the **Query** field, enter the following SQL query:

     SELECT c.id, c.firstName, c.lastName, c.age, c.salary, c.company, c.isVested, c.*ts FROM c WHERE c.*ts >= @HighWaterMark ORDER BY c._ts

   - Ensure that the **Query results ordered by _ts** checkbox is selected.
   - Click the **OK** button.
7. On the **Cognitive Search** blade, click the **OK** button.
8. On the **Index** blade, perform the following tasks:
   - In the **Index name** text box, type **memberindex**.
   - In the **Key** drop-down list, ensure that the **id** entry is selected.
   - For the **id** field in the table, ensure that the **RETRIEVABLE**, **FILTERABLE**, and **SORTABLE** checkboxes are selected.
   - For the **firstName** field in the table, ensure that the **RETRIEVABLE**, **SORTABLE**, and **SEARCHABLE** options are selected.

    - For the **lastName** field in the table, ensure that the **RETRIEVABLE**, **SORTABLE**, and **SEARCHABLE** checkboxes are selected.
    - For the **age** field in the table, ensure that the **RETRIEVABLE**, **FILTERABLE**, **SORTABLE**, and **FACETABLE** checkboxes are selected.
    - For the **salary** field in the table, ensure that the **RETRIEVABLE**, **FILTERABLE**, **SORTABLE**, and **FACETABLE** checkboxes are selected.
    - For the **company** field in the table, ensure that the **RETRIEVABLE**, **FACETABLE**, and **SEARCHABLE** checkboxes are selected.
    - For the **isVested** field in the table, ensure that the **RETRIEVABLE**, **FILTERABLE**, **SORTABLE**, **FACETABLE** checkboxes are selected.
    - Click the **OK** button.
9. On the **Create an Indexer** blade, perform the following tasks:
    - In the **Name** text box, type **cosmosmemberindexer**.
    - In the **Schedule** section, select the **Custom** option.
    - In the **Interval (minutes)** text box, type **5**.
    - In the **Start time (UTC)** field, specify the current date and accept the default value of the time entry.
    - Click the **OK** button.
10. Back on the **Import data** blade, click the **OK** button.

Task 3: Validate API App

1. In the hub menu in the Azure portal, click **Resource groups**.
2. On the **Resource groups** blade, click **AADesignLab0701-RG**.
3. On the **AADesignLab0701-RG** blade, click the entry representing the App Service API app you created earlier in this lab.
4. On the API app blade, click **Application settings**.
5. On the Application settings blade, scroll down to the **Application settings** section and perform the following tasks:
    - Set the value of the **Search:AccountName** setting to the name of the Azure Search instance you created earlier in this lab.
    - Set the value of the **Search:QueryKey** setting to the value of the **PRIMARY KEY** of the Azure Search instance you created earlier in this lab.
    - Click the **Save** button at the top of the blade.
6. On the API app blade, click **Overview**.
7. Click the **Restart** button at the top of the blade and, when prompted to confirm, click **Yes**.
8. Click the **Browse** button at the top of the blade. This will open a new browser tab displaying the **Swagger UI** homepage.

> **Note**: If you click the **Browse** button before the API app has fully restarted, you may not be able to follow the remaining steps in this task. If this happens, refresh your browser until the API app is running again.

9. On the **Swagger UI** homepage, click **Cosmos DB API v.1.0.0** at the top of the page and select the **Cosmos DB API v.2.0.0** option from the drop-down list.
10. Click **GET/Documents/search**.
11. In the **Parameters** section, in the **Value** text box of the **query** parameter, type the following text:

    ```
    Oneal
    ```

12. In the **Response Messages** section, click the **Try it out!** button.
13. Review the results of the request.
14. In the **Parameters** section, in the **Value** text box of the **query** parameter, type the following text:

    ```
    penn*
    ```

15. In the **Response Messages** section, click the **Try it out!** button.
16. Review the results of the request.
17. Close the new browser tab and return to the browser tab displaying the Azure portal.

Review: In this exercise, you created an Azure Search instance that uses an indexer to index the documents in Azure Cosmos DB.

Exercise 4: Remove lab resources

Task 1: Delete the resource group

1. In the hub menu in the Azure portal, click **Resource groups**.
2. On the **Resource groups** blade, click **AADesignLab0701-RG**.
3. On the **AADesignLab0701-RG** blade, click **Delete resource group**.
4. In the **Are you sure you want to delete "AADesignLab0701-RG"?** pane, in the **TYPE THE RESOURCE GROUP NAME** text box, type **AADesignLab0701-RG** and click **Delete**.

Review: In this exercise, you removed the resources used in this lab.