**LAB01 : Load Data Into Cosmos DB with ADF**

In this lab, you will populate an Azure Cosmos DB container from an existing set of data using tools built in to Azure. After importing, you will use the Azure portal to view your imported data.

**Create Azure Cosmos DB Database and Container**

You will now create a database and container within your Azure Cosmos DB account.

1. Navigate to the [Azure Portal](https://portal.azure.com/)
2. On the left side of the portal, select the **Resource groups** link.

Une image contenant texte

Description générée automatiquement

1. In the **Resource groups** blade, locate and select the **cosmos-openhack-shared-rg** resource group.

Graphical user interface, text, application, chat or text message

Description automatically generated

1. In the **cosmos-openhack-shared-rg** blade, select the **Azure Cosmos DB** account .

Graphical user interface, application, email

Description automatically generated

1. In the **Azure Cosmos DB** blade, locate and select the **Overview** link on the left side of the blade. At the top select the **Add Container** button.

A screenshot of a computer

Description automatically generated

1. In the **Add Container** popup, perform the following actions:
   1. In the **Database id** field, select the **use existing** option and select the database  **database\_teamxx (where xx represent your user number)**.
   2. *Note: For Cosmos DB accounts using Provisioned Throughput only*: Do not check the **Provision dedicated throughput** option.

Provisioning throughput for a database allows you to share the throughput among all the containers that belong to that database. Within an Azure Cosmos DB database, you can have a set of containers which shares the throughput as well as containers, which have dedicated throughput.

* 1. In the **Container Id** field, enter the value **FoodCollection**.
  2. In the **Partition key** field, enter the value /foodGroup.

Select the **OK** button.

Graphical user interface, text, application, email

Description automatically generated

1. Wait for the creation of the new **container** and verify that the FoodCollection was created with no item for the moment.

Graphical user interface, application

Description automatically generated

**Import Lab Data Into Container**

You will use **Azure Data Factory (ADF)** to import the JSON array stored in the **nutrition.json** file from Azure Blob Storage.

To learn more about copying data to Cosmos DB with ADF, please read [ADF's documentation](https://docs.microsoft.com/azure/data-factory/connector-azure-cosmos-db)

1. On the left side of the portal, select the **Resource groups** link.

Une image contenant texte

Description générée automatiquement

1. In the **Resource groups** blade, locate and select the **cosmos-openhack-teams-rg** resource group.
2. Search for the data factory ressource named cosmos-openhack-teamxx-adf where (xx is your user number)

Une image contenant texte

Description générée automatiquement

1. Select **Open azure data Factory Studio** and you will launch ADF.

Graphical user interface, text, application, email

Description automatically generated

1. Select **Ingest**
   * We will be using ADF for a one-time copy of data from a source JSON file on Azure Blob Storage to a database in Cosmos DB’s SQL API. ADF can also be used for more frequent data transfers from Cosmos DB to other data stores

Graphical user interface, application, Word

Description automatically generated

1. Select Built-in copy task è next

Une image contenant texte

Description générée automatiquement

1. We will import data from a json file on Azure Blob Storage. In addition to Blob Storage, you can use ADF to migrate from a wide variety of sources. We will not cover migration from these sources in this tutorial

On the Source data store, select Azure Blob storage for the **Source Type** and New connection to create a linked service to the data source

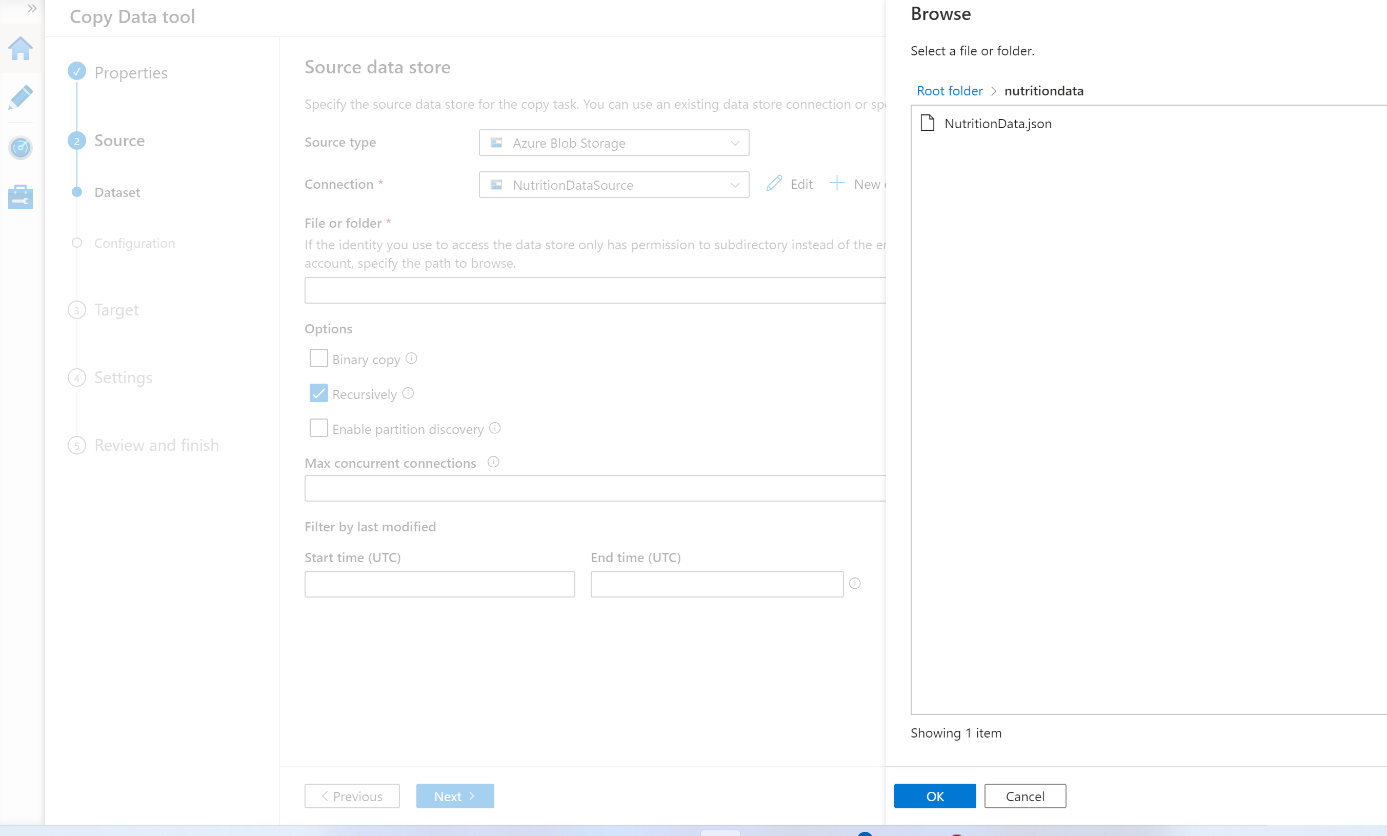
Une image contenant texte

Description générée automatiquement

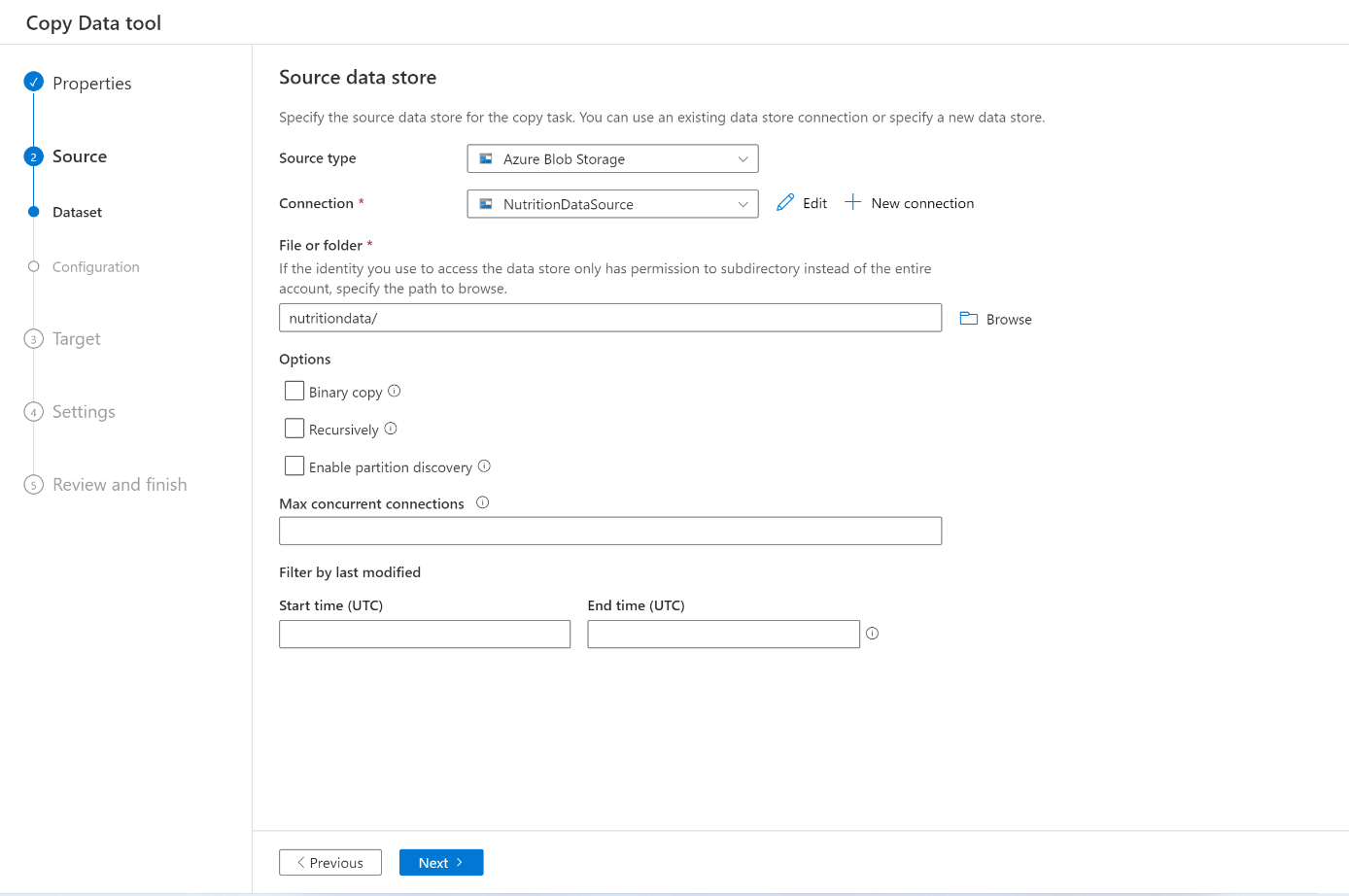
1. Name the source **NutritionDataSource** and select **SAS URI** as the Authentication method. Please use the following SAS URI for read-only access to this Blob Storage container :

https://mdwdatalake3z2bv.blob.core.windows.net/?sv=2021-06-08&ss=bfqt&srt=sco&sp=rwdlacupitfx&se=2022-07-15T21:41:03Z&st=2022-07-01T13:41:03Z&spr=https,http&sig=FO5%2FC%2B9FCJ%2F3MUtBKNMvpEFfbMR4t8SEn8xWB8IkvhI%3D

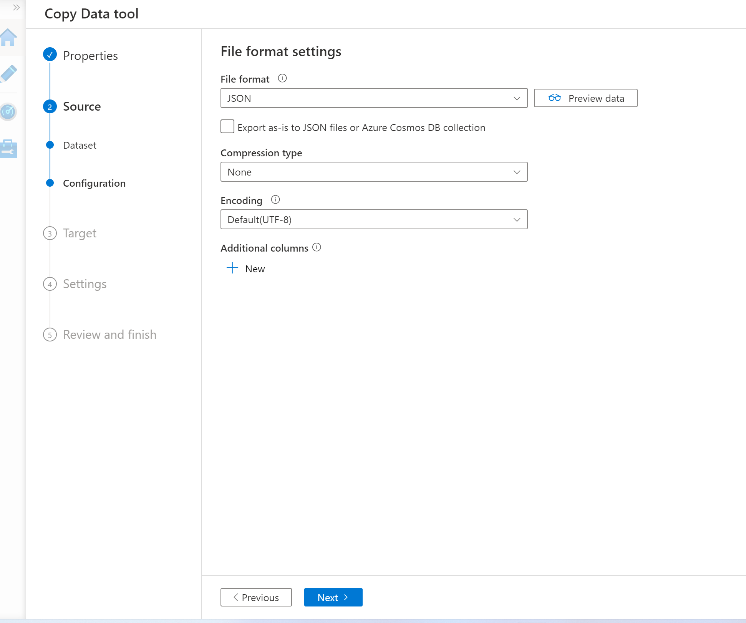
1. Select **Create**
2. Select **Next**
3. Select **Browse**, then double-click to open the **nutritiondata** folder
4. Select the **NutritionData.json** file, then select **Choose**



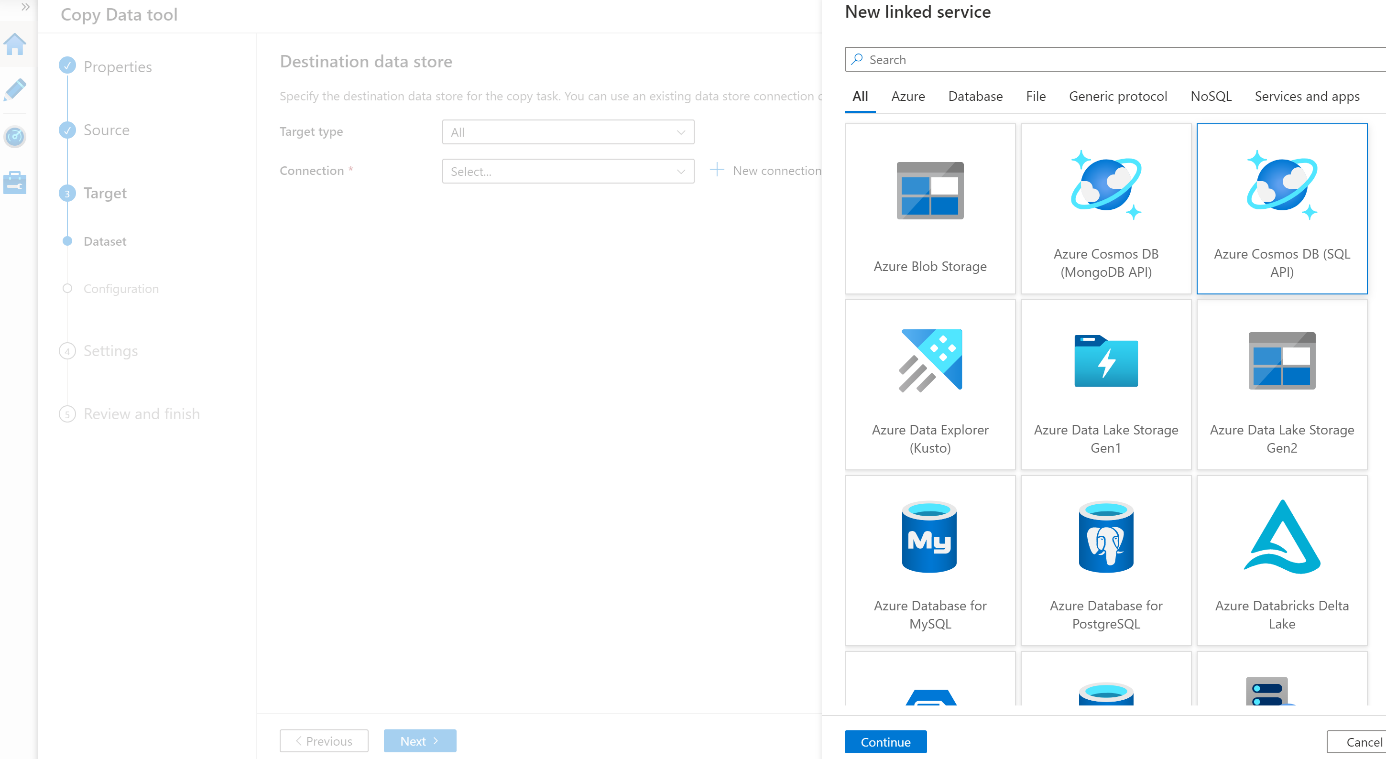
1. Un-check **Copy file recursively** or **Binary Copy** if they are checked. Also ensure that other fields are empty. Click **Next**



1. Select the file format as **JSON format**. Then select **Next**.



1. You have now successfully connected the Blob Storage container with the nutrition.json file as the source.
2. For the **Destination data store** add the Cosmos DB target data store by selecting **Create new connection** and selecting **Azure Cosmos DB (SQL API)**.



1. Name the linked service **NutritionTargetCosmos** and select your Azure subscription and Cosmos DB account. You should also select the Cosmos DB **ImportDatabase** that you created earlier.

Une image contenant texte

Description générée automatiquement

1. Select your newly created **targetcosmosdb** connection as the Destination data store.
2. Select your **FoodCollection** container from the drop-down menu. You will map your Blob storage file to the correct Cosmos DB container. Select **Next** to continue.

Une image contenant texte

Description générée automatiquement

1. Give a name to the copy task and select **next**.



1. Select **Next** to begin deployment After deployment is complete, select **Monitor**.

Graphical user interface, text, application, email

Description automatically generated

1. After a few minutes, refresh the page and the status for the ImportNutrition pipeline should be listed as **Succeeded**.

Graphical user interface, text, application, email

Description automatically generated

1. Once the import process has completed, close the ADF. You will now proceed to validate your imported data.

**Validate Imported Data**

The Azure Cosmos DB Data Explorer allows you to view documents and run queries directly within the Azure Portal. In this exercise, you will use the Data Explorer to view the data stored in our container.

You will validate that the data was successfully imported into your container using the **Items** view in the **Data Explorer**.

1. Return to the **Azure Portal** ([http://portal.azure.com](http://portal.azure.com/)).
2. In the **Resource groups** blade, locate and select the **cosmos-openhack-shared-rg**  resource group.
3. In the **Azure Cosmos DB** blade, locate and select the **Data Explorer** link on the left side of the blade.
4. In the **Data Explorer** section, expand the **database\_teamxx** database node and then expand the **FoodCollection** container node.

Graphical user interface, application

Description automatically generated

1. Within the **FoodCollection** node, select the **Items** link to view a subset of the various documents in the container. Select a few of the documents and observe the properties and structure of the documents.

Graphical user interface

Description automatically generated