Labs Data & AI Innovation Day

Lab 1: Azure Cosmos DB for NoSQL

This document explains how to create databases and containers that support vector search in a Cosmos DB for NoSQL account

# Connect to an existing Cosmos DB for NoSQL account

Connect to the Cosmos DB for NoSQL account that was just created:

1. In the search bar, type ‘Azure Cosmos DB for NoSQL’
2. Click on the instance that was just created ‘cosmos-nosql-2024'
3. Click on ‘Data Explorer’ in the menu
4. Close the introduction video dialog box if this is your first connection

A screenshot of a computer

Description automatically generated

Create database and container

Click on the down arrow, next to ‘New container’ and select ‘New Container’

A screenshot of a computer

Description automatically generated

Fill the New container form with the following information:

Database id: database\_teamXX

Container id: conversations

Partition key: /id

Click on the ‘OK’ button

A screenshot of a computer

Description automatically generated

Under ‘My Data’, you should see the database and container freshly created

A screenshot of a computer

Description automatically generated

Repeat the operations to create a ‘products’ container with a ‘/id’ partition key

Create a container that support vector search

In this section, we will use a Python notebook to create a container that supports vector search

## Modify environment variables

Rename the “.env template” file to “.env” and modify the variables to reflect your environment

|  |  |  |
| --- | --- | --- |
| AZURE\_COSMOSDB\_NOSQL\_ENDPOINT | The name of your Cosmos DB for NoSQL | cosmos-nosql-001documents.azure.com |
| AZURE\_COSMOSDB\_NOSQL\_KEY | Cosmos DB key | xxxx |
| AZURE\_COSMOSDB\_NOSQL\_DATABASE\_NAME | Database name | Database\_teamXX |
| AZURE\_COSMOSDB\_NOSQL\_DATABASE\_NAME | Database name | database\_<team\_name>, e.g. database\_team01 |
| AZURE\_COSMOSDB\_NOSQL\_VECTORS\_CONTAINER\_NAME | Vectors container name | vectors |

## Prepare Python virtual environment

Run these commands to create a Python virutal environment and install the required libraries:

1. Open the “lab1” folder in Visual Studio Code
2. Open a new Powershell Teminal > New Terminal
3. Type this command to create a virtual environment: python -m venv .venv
4. Activate the virtual environment with .venv\scripts\activate
5. Install the required libraries with pip install -r requirements.txt

A computer screen shot of a computer code

Description automatically generated

## Create container

1. Open the “create\_vectors\_container.ipynb” Jupyter notebook in Visual Studio Code
2. In the top-right corner, click on “Select kernel”
3. Select “Python environment” and selec the recommended environment (that should point to the .venv environment that you created in the preview section)  
     
   A screenshot of a computer program

   Description automatically generated
4. In the menu bar, click on “Run all”  
     
   A screen shot of a computer

   Description automatically generated  
     
   A screenshot of a computer program

   Description automatically generated
5. Verify in the Azure portal that the “vectors” container has been successfully created.

## A screenshot of a computer Description automatically generated