## USER STORY

# **Azure Cosmos DB: Build a .NET application using the Graph API**

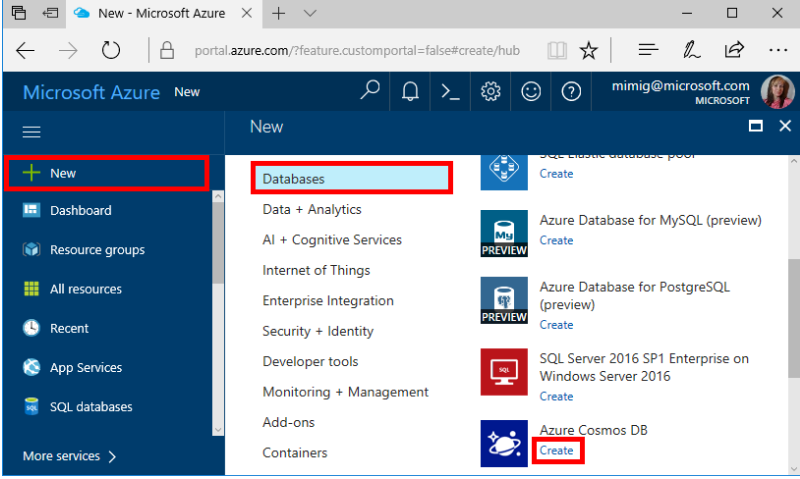
Azure Cosmos DB is Microsoft’s globally distributed multi-model database service. You can quickly create and query document, key/value, and graph databases, all of which benefit from the global distribution and horizontal scale capabilities at the core of Azure Cosmos DB.

This quick start demonstrates how to create an Azure Cosmos DB account, database, and graph (container) using the Azure portal. You then build and run a console app built on the Graph API.

## Step : 1

## Create a database account

1. In a new window, sign in to the [Azure portal](https://portal.azure.com/).
2. In the left pane, click **New**, click **Databases**, and then under **Azure Cosmos DB**, click **Create**.

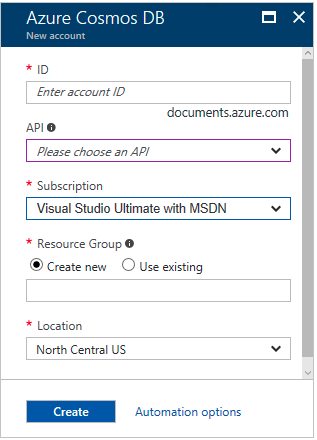


1. In the **New account** blade, specify the configuration that you want for this Azure Cosmos DB account.

With Azure Cosmos DB, you can choose one of four programming models: Gremlin (graph), MongoDB, SQL (DocumentDB), and Table (key-value), each which currently require a separate account.

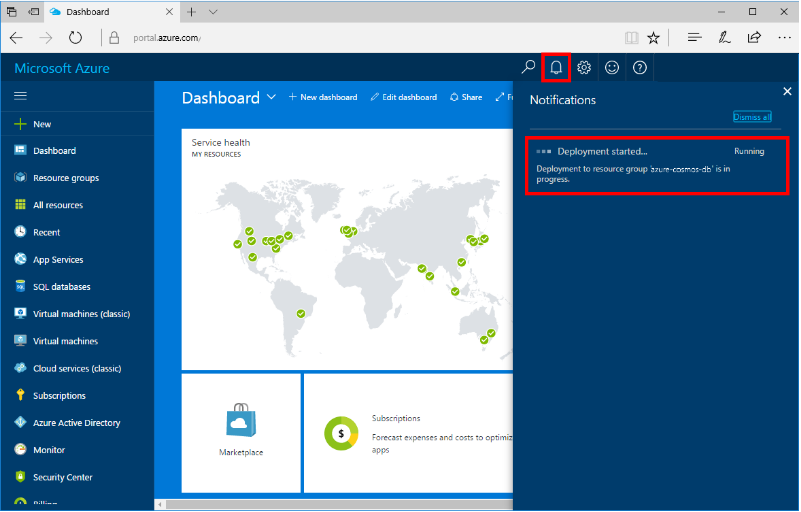
In this quick-start article, we program against the Graph API, so choose **Gremlin (graph)** as you fill out the form. If you have document data from a catalog app, key/value (table) data, or data that's migrated from a MongoDB app, realize that Azure Cosmos DB can provide a highly available, globally distributed database service platform for all your mission-critical applications.

Complete the fields on the **New account** blade, using the information in the following screenshot as a guide - your values may be different than the values in the screenshot.

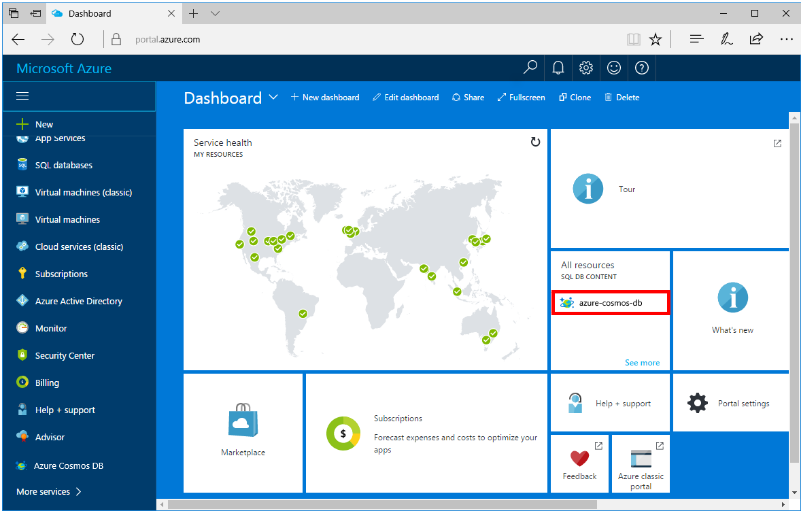


| Setting | Suggested value | Description |
| --- | --- | --- |
| ID | Unique value | A unique name that identifies this Azure Cosmos DB account. Because documents.azure.comis appended to the ID that you provide to create your URI, use a unique but identifiable ID. The ID must contain only lowercase letters, numbers, and the hyphen (-) character, and it must contain from 3 to 50 characters. |
| API | Gremlin (graph) | We program against the [Graph API](https://docs.microsoft.com/en-us/azure/cosmos-db/graph-introduction) later in this article. |
| Subscription | Your subscription | The Azure subscription that you want to use for this Azure Cosmos DB account. |
| Resource Group | The same value as ID | The new resource group name for your account. For simplicity, you can use the same name as your ID. |
| Location | The region closest to your users | The geographic location in which to host your Azure Cosmos DB account. Choose the location closest to your users to give them the fastest access to the data. |

1. Click **Create** to create the account.
2. On the top toolbar, click the **Notifications** icon  to monitor the deployment process.



1. When the Notifications window indicates the deployment succeeded, close the notification window and open the new account from the **All Resources** tile on the Dashboard.

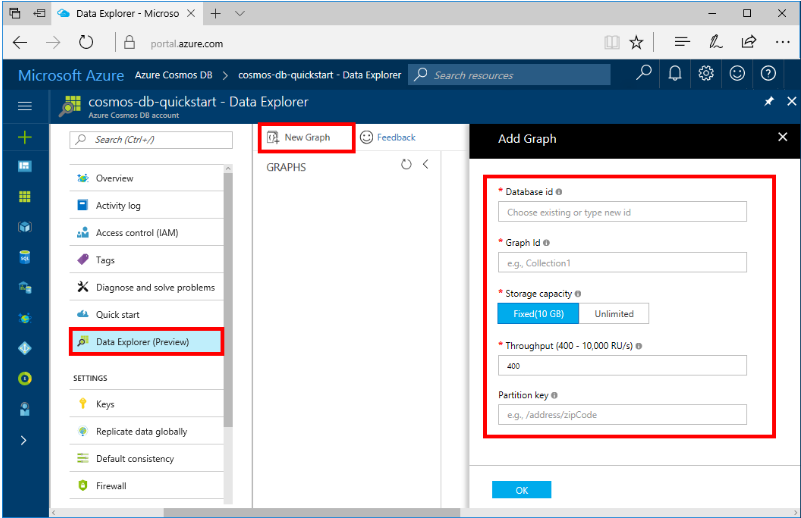


## Step : 2

## Add a graph

You can now use the Data Explorer tool in the Azure portal to create a graph database.

1. In the Azure portal, in the left navigation menu, click **Data Explorer (Preview)**.
2. In the **Data Explorer (Preview)** blade, click **New Graph**, then fill in the page using the following information.



| Setting | Suggested value | Description |
| --- | --- | --- |
| Database id | sample-database | The ID for your new database. Database names must be between 1 and 255 characters, and cannot contain / \ # ? or a trailing space. |
| Graph id | sample-graph | The ID for your new graph. Graph names have the same character requirements as database ids. |
| Storage Capacity | 10 GB | Leave the default value. This is the storage capacity of the database. |
| Throughput | 400 RUs | Leave the default value. You can scale up the throughput later if you want to reduce latency. |
| Partition key | /userid | A partition key that will distribute data evenly to each partition. Selecting the correct partition key is important in creating a performant graph, read more about it in [Designing for partitioning](https://docs.microsoft.com/en-us/azure/cosmos-db/partition-data#designing-for-partitioning). |

1. Once the form is filled out, click **OK**.

## Step : 3

## Clone the sample application

Now let's clone a Graph API app from github, set the connection string, and run it. You'll see how easy it is to work with data programmatically.

1. Open a git terminal window, such as git bash, and cd to a working directory.
2. Run the following command to clone the sample repository.

bashCopy

git clone https://github.com/Azure-Samples/azure-cosmos-db-graph-dotnet-getting-started.git

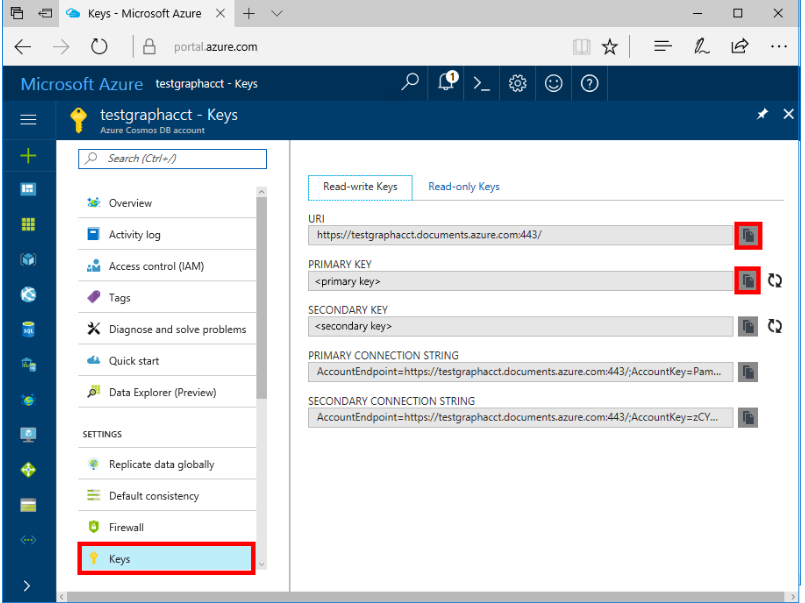
1. Then open Visual Studio and open the solution file.

## Step : 4

## Update your connection string

Now go back to the Azure portal to get your connection string information and copy it into the app.

1. In Visual Studio 2017, open the App.config file.
2. In the Azure portal, in your Azure Cosmos DB account, click **Keys** in the left navigation.



1. Copy your **URI** value from the portal and make it the value of the Endpoint key in App.config. You can use the copy button as shown in the preceding screenshot to copy the value.

<add key="Endpoint" value="https://FILLME.documents.azure.com:443" />

1. Copy your **PRIMARY KEY** value from the portal, and make it the value of the AuthKey key in App.config, then save your changes.

<add key="AuthKey" value="FILLME" />

You've now updated your app with all the info it needs to communicate with Azure Cosmos DB.

## Step : 5

## Run the console app

1. In Visual Studio, right-click on the **GraphGetStarted** project in **Solution Explorer** and then click **Manage NuGet Packages**.
2. In the NuGet **Browse** box, type Microsoft.Azure.Graphs and check the **Includes prerelease** box.
3. From the results, install the **Microsoft.Azure.Graphs** library. This installs the Azure Cosmos DB graph extension library package and all dependencies.

If you get a message about reviewing changes to the solution, click **OK**. If you get a message about license acceptance, click **I accept**.

1. Click CTRL + F5 to run the application.

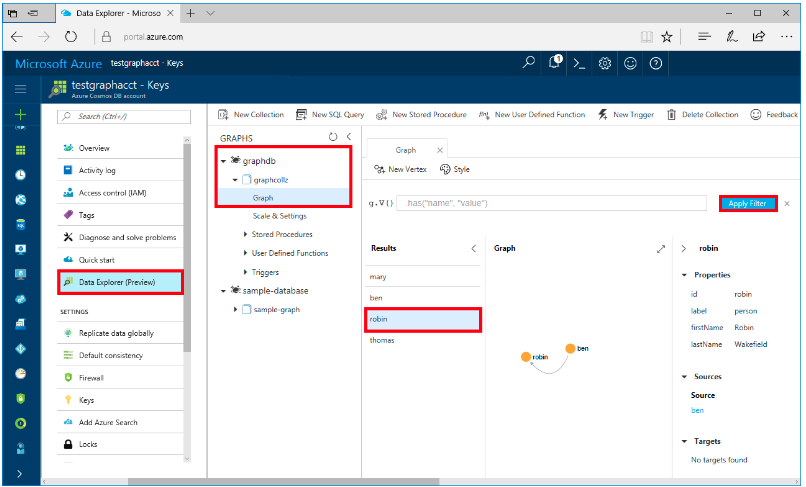
The console window displays the vertexes and edges being added to the graph. When the script completes, press ENTER twice to close the console window.

## Step : 6

## Browse using the Data Explorer

You can now go back to Data Explorer in the Azure portal and browse and query your new graph data.

1. In Data Explorer, the new database appears in the Graphs pane. Expand **graphdb**, **graphcollz**, and then click **Graph**.
2. Click the **Apply Filter** button to use the default query to view all the verticies in the graph. The data generated by the sample app is displayed in the Graphs pane.



You can zoom in and out of the graph, you can expand the graph display space, add additional verticies, and move verticies on the display surface.