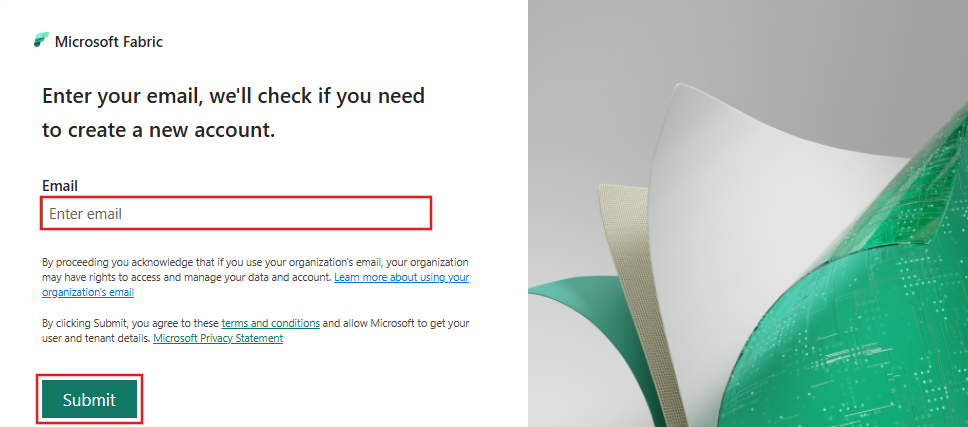
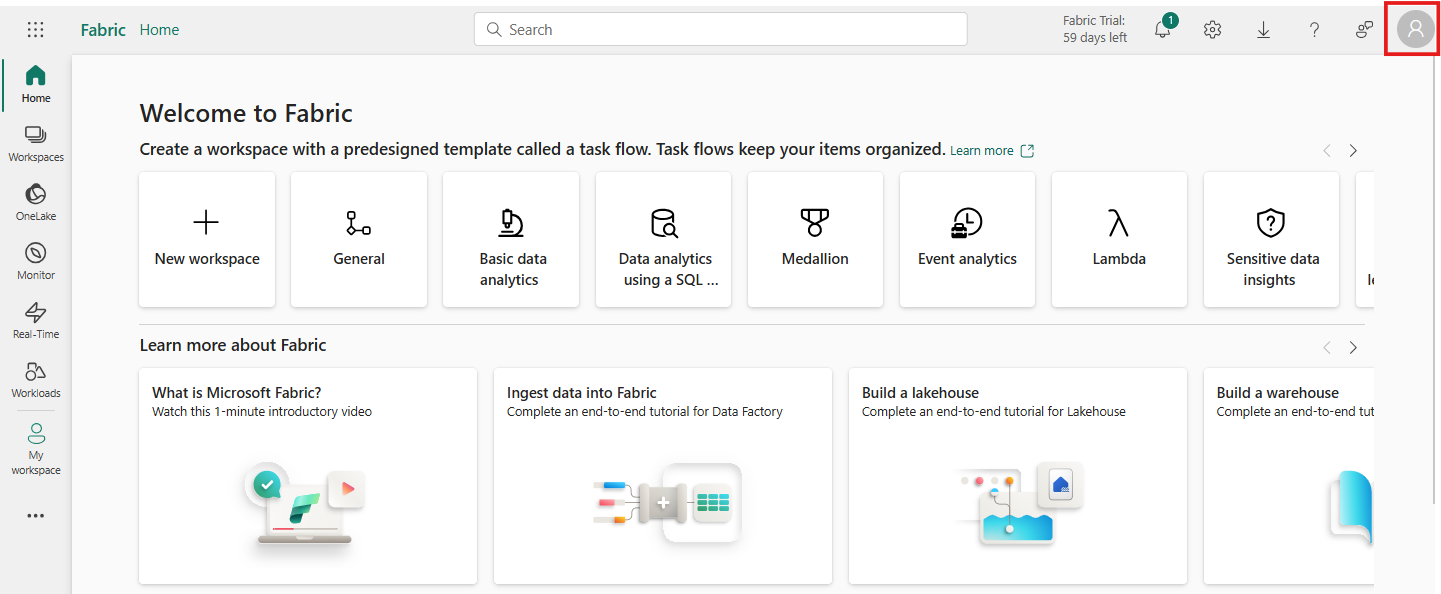
# ­Lab 1 – Build a Fabric Database for AdventureWorks for querying, reporting, and sharing the data

In this use-case, you are going to build a Fabric Database in Fabric Portal

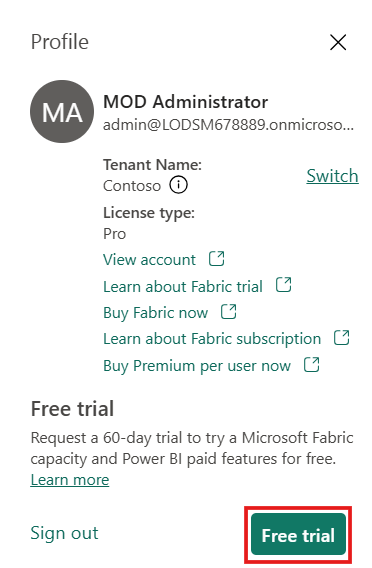
## Exercise 1 – Set up your environment

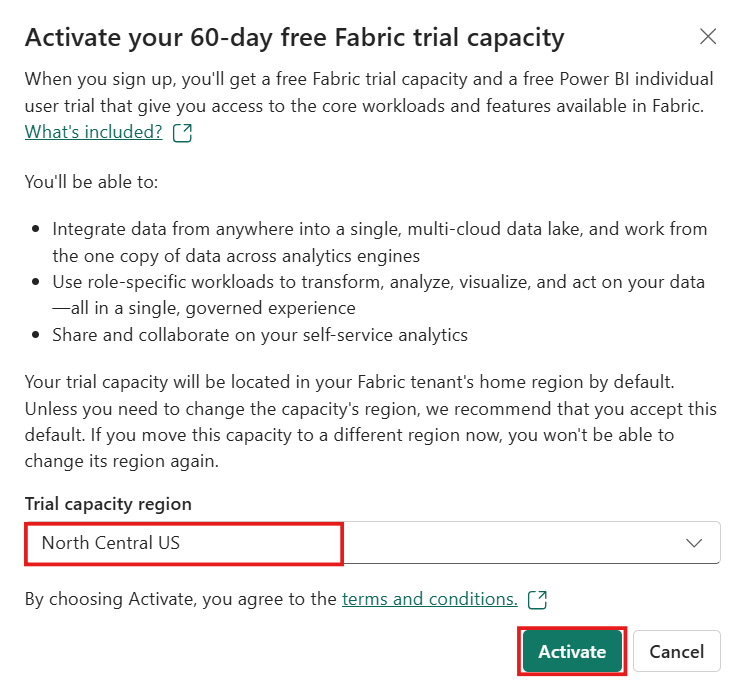
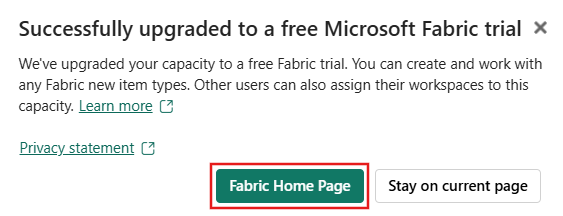
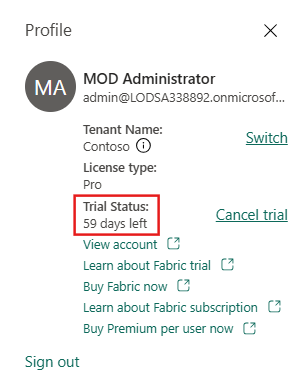
### **Task-1: Start a Fabric Capacity Trial**

Follow these steps to start your Fabric capacity trial and become the Capacity administrator of that trial.

1. Open your browser and browse the **Microsoft Fabric Trial Page** <https://app.fabric.microsoft.com/> and sign-in with your credentials.
2. On the Trial Page, Click on the **Account** **Manager**.
3. In the Account manager, select **Free trial**. If you don't see **Free trial** or **Start trial** or a **Trial status**, trials might be disabled for your tenant.

**Note:** If the Account manager already displays **Trial status**, you may already have a **Power BI trial** or a **Fabric (Free) trial** in progress. To test this out, attempt to use a Fabric feature. For more information, see [**Start using Fabric**](https://learn.microsoft.com/en-us/fabric/fundamentals/fabric-trial#other-ways-to-start-a-microsoft-fabric-trial).

****

1. If prompted, agree to the terms and select the appropriate Trial capacity region and then select **Activate**.
2. Once your trial capacity is ready, you receive a confirmation message. Select **Fabric Home Page** to begin working in Fabric. You're now the Capacity administrator for that trial capacity.
3. Open your Account manager again. Notice the heading for **Trial status**. Your Account manager keeps track of the number of days remaining in your trial.

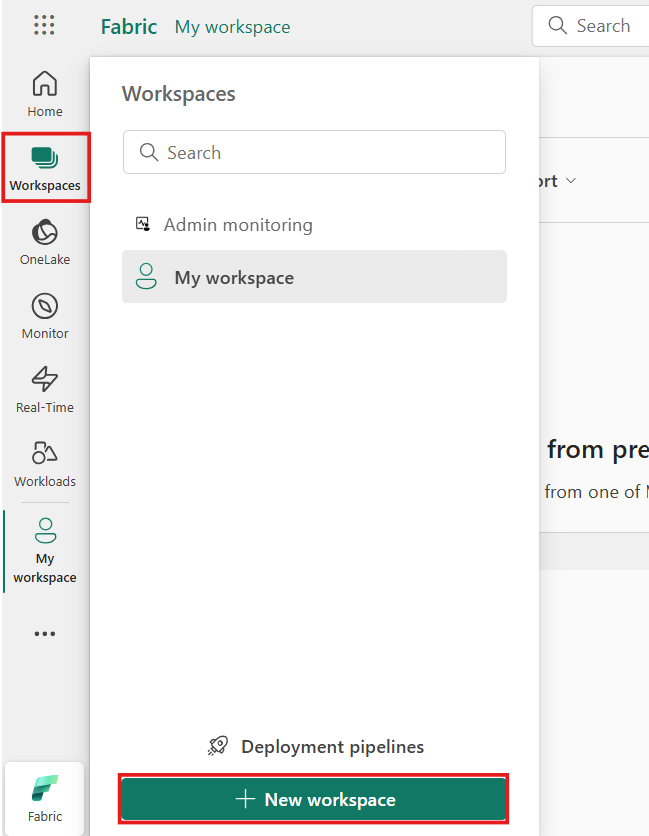
Congratulations. You now have a Fabric trial capacity that includes a Power BI individual trial (if you didn't already have a Power BI paid license) and a Fabric trial capacity.

### **Task-2: Create a New Fabric Workspace**

You can use an existing workspace or create a new Fabric workspace.  In workspaces, you create collections of items such as lakehouses, warehouses, and reports. You must be a member of the Admin or Member roles for the workspace to create a SQL database.

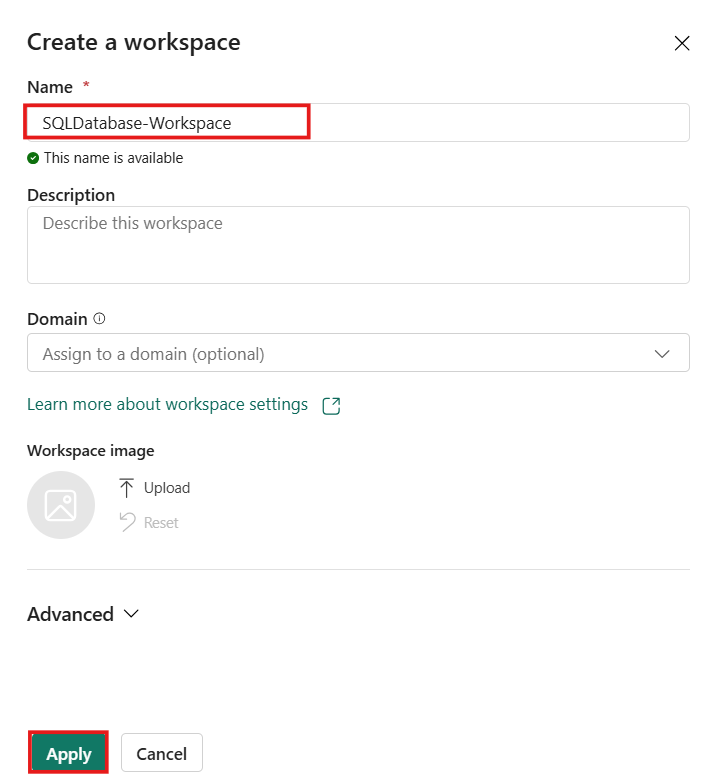
To create a workspace:

1. From left pane, select **Workspaces** > **New workspace**.



1. The Create a workspace pane opens.
   * Give the workspace a unique name (mandatory).
   * Provide a description of the workspace (optional).
   * Assign the workspace to a domain (optional).

If you are a domain contributor for the workspace, you can associate the workspace to a domain, or you can change an existing association.



1. When done, either continue to the advanced settings, or select **Apply**.

## Exercise 2 – Enable SQL Database in your Fabric

### **Task-1: Enable SQL database for your tenant**

Follow these steps to enable SQL database for your tenant.

1. Navigate to the [tenant settings](https://learn.microsoft.com/en-us/fabric/admin/tenant-settings-index) in the admin portal and in **Microsoft Fabric**, expand **Users can create Fabric items** and **SQL database (preview)**.
2. Enable the **Users can create Fabric items** and **SQL database (preview)** switches.
3. (Optional) Use the **Specific security groups** option to enable SQL database for specific users.
   * You can select **The entire organization** if you want everyone to create databases.
   * You can select **Specific security groups** to allow selected individuals in a group to create databases.
   * You can select **Except specific security groups** to block the access to create databases for individuals in that group. This option takes precedence over the previous one, if a user belongs to an included group and an excluded group, the individual would be blocked from creating databases.
   * Select **Capacity admins can enable/disable** if you want capacity admins to manage the control of who can access the SQL database workload.
4. Select **Apply**.

### **Task-2: Enable SQL Database for a capacity**

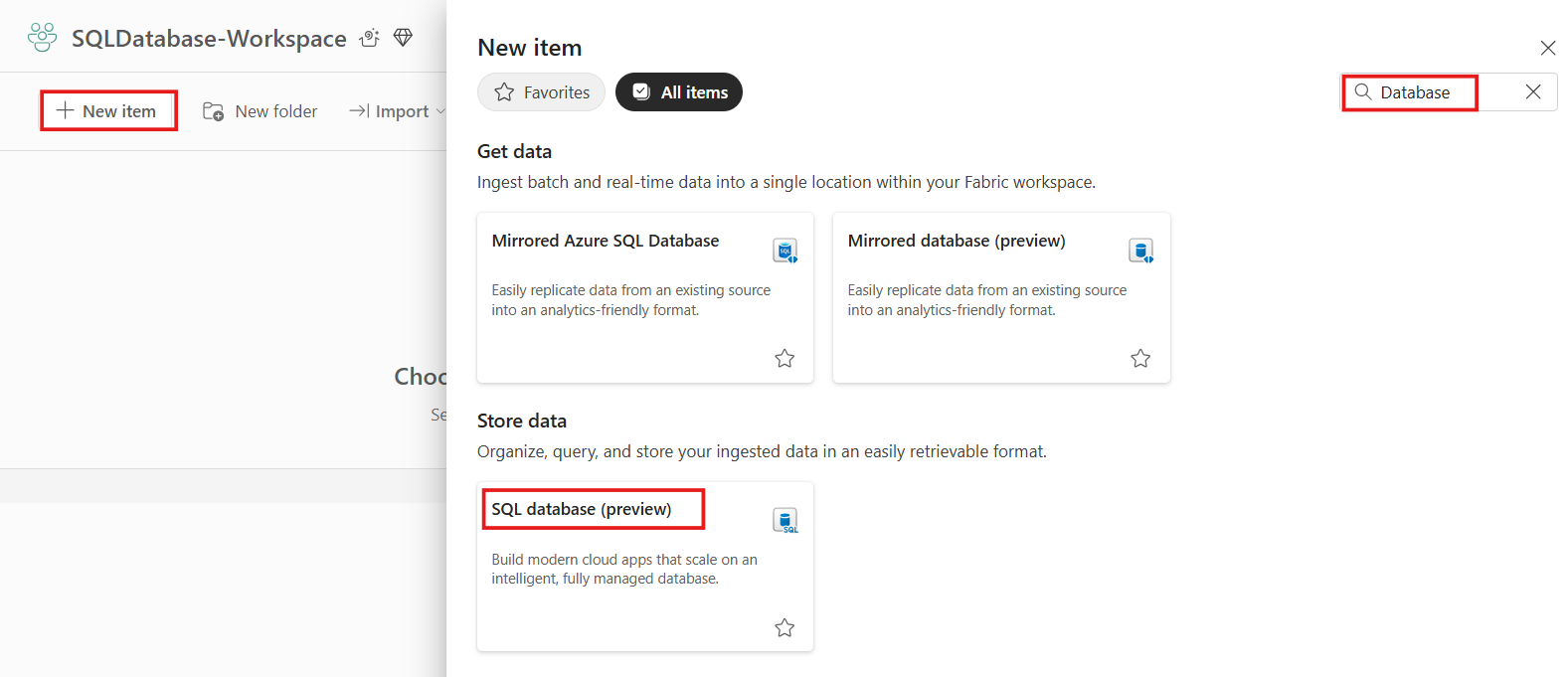
Follow these steps to enable SQL database in Microsoft Fabric for a specific capacity:

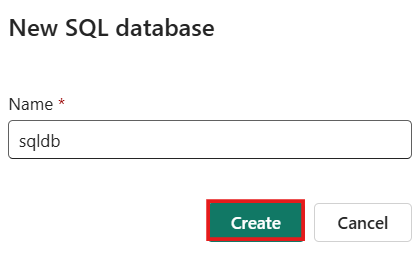
1. Navigate to the [capacity settings](https://learn.microsoft.com/en-us/fabric/admin/service-admin-portal-capacity-settings) in the admin portal.
2. Select the capacity where you want to enable SQL database.
3. Select the **Delegate tenant settings** tab.
4. Expand the **SQL database (preview)** setting.
5. Check the **Override tenant admin selection** checkbox and verify that the **SQL database (preview)** setting is enabled.
6. (Optional) Use the **Specific security groups** option to enable SQL database for specific users.
   * Select **All the users in capacity** if you want everyone to be able to create databases.
   * Select **Specific security groups** to allow selected individuals in a group to create databases.
   * You can also select **Except specific security groups** to block from creating the databases. This option takes precedence over the previous one, if a user belongs to an included group and an excluded group, the individual would be blocked from creating databases.
7. Select **Apply**.

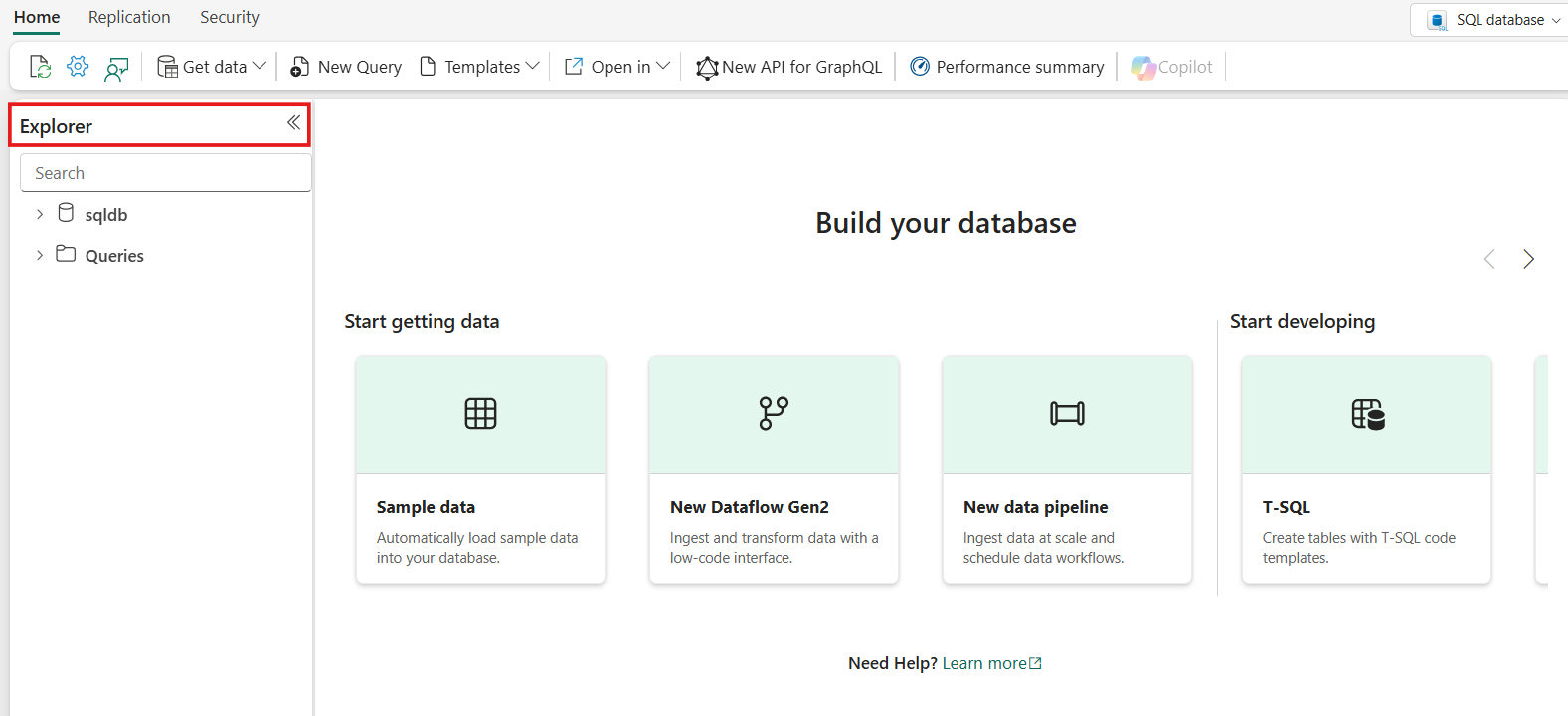
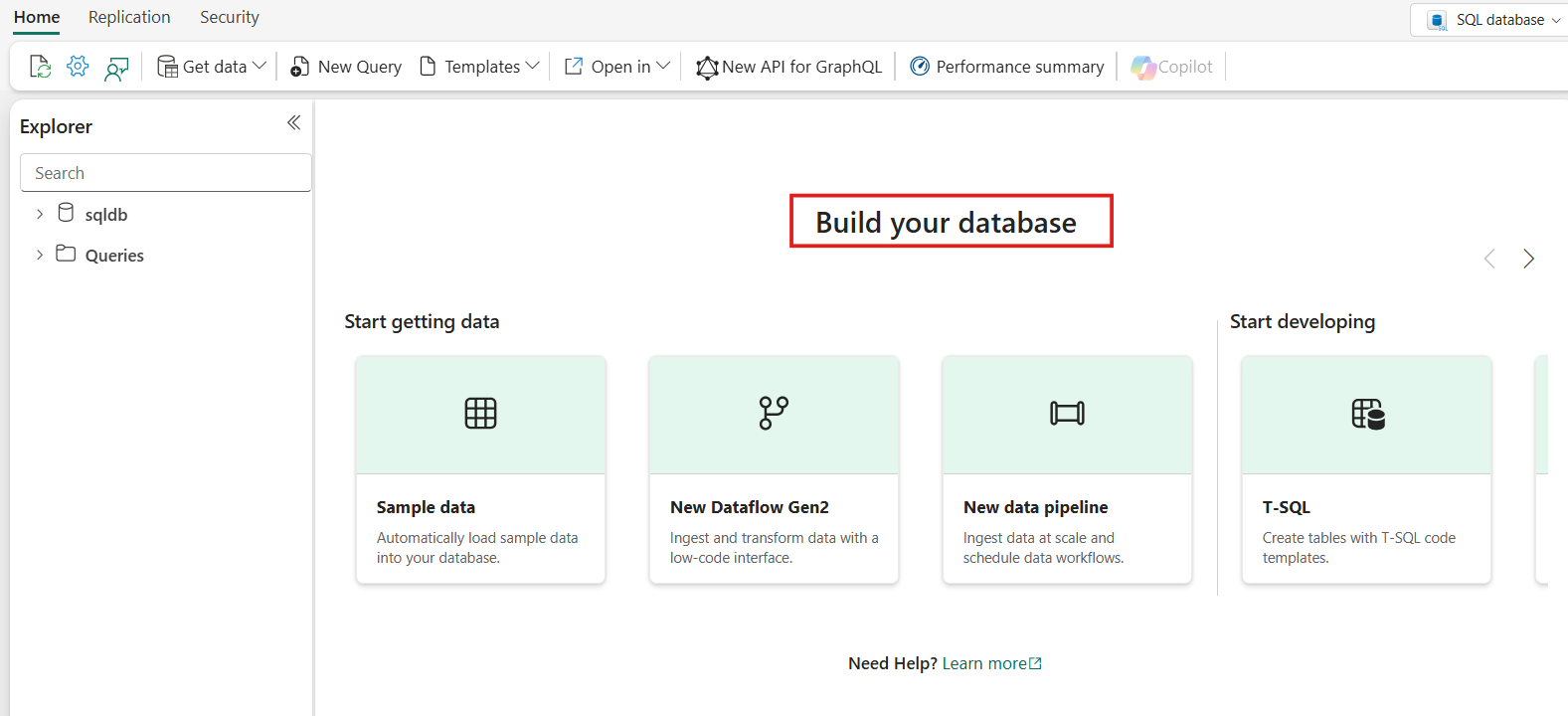
### **Task-3: Disable SQL Database**

1. To disable SQL database, you can disable the **SQL database (preview)** admin switch.
2. If you disable SQL database for a specific capacity while it's available in your organization, your selection will only affect that capacity.
3. When SQL database not enabled in tenant settings, users who try to create a new SQL database will receive the error message "SQL database failed to create." The tenant admin switch is ignored for trial capacities.
4. To disable and disallow SQL database in Microsoft Fabric, also disable trial capacities in your tenant by turning off the "Users can try Microsoft Fabric paid features" from the Fabric Admin portal.

## Exercise 3 – Create a SQL Database in the Fabric Portal

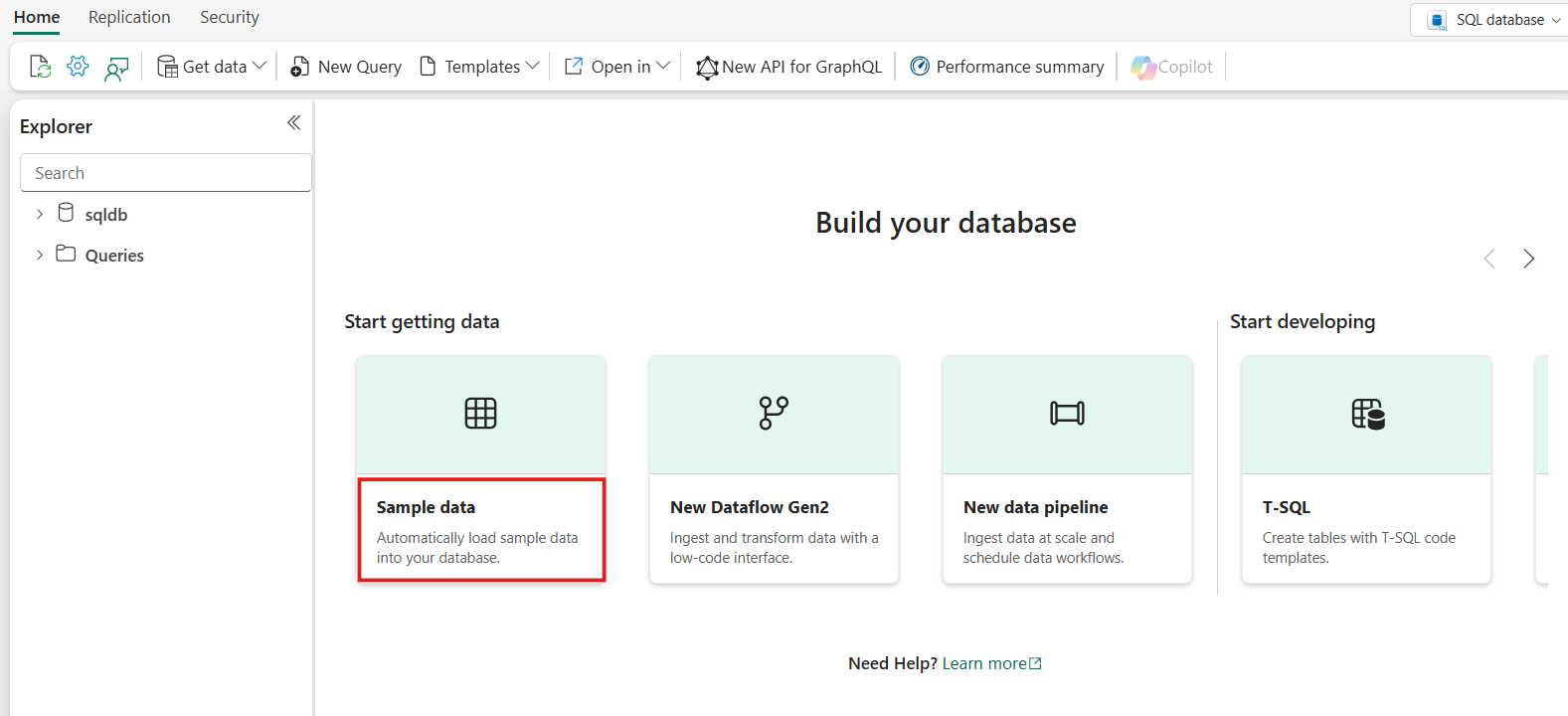
1. In the Fabric Portal, select **+ New Item**, and search for **SQL Databases (preview)** tile.
2. Provide a name for the **New Database**. Select **Create**.



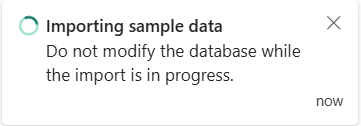
1. When the new database is provisioned, on the **Home** page for the database, notice the **Explorer** pane showing database objects.
2. Under **Build your database**, three useful tiles can help you get your newly created database up and running.
   * **Sample data** option lets you import a sample data into your **Empty** database.
   * **T-SQL** option gives you a web-editor that can be used to write T-SQL to create database object like schema, tables, views, and more. For users who are looking for code snippets to create objects, they can look for available samples in **Templates** drop down list at the top of the menu.
   * **Connection strings** option shows the SQL database connection string that is required when you want to connect using SQL Server Management Studio, the mssql extension with Visual Studio Code, or other external tools.
3. **Consider adding your new database to source control.**

## Exercise 4 - Load AdventureWorks sample data in your SQL database in Microsoft Fabric

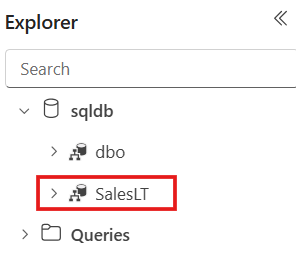
1. Once the new database is created, open the database's home page. Select **Sample Data**.

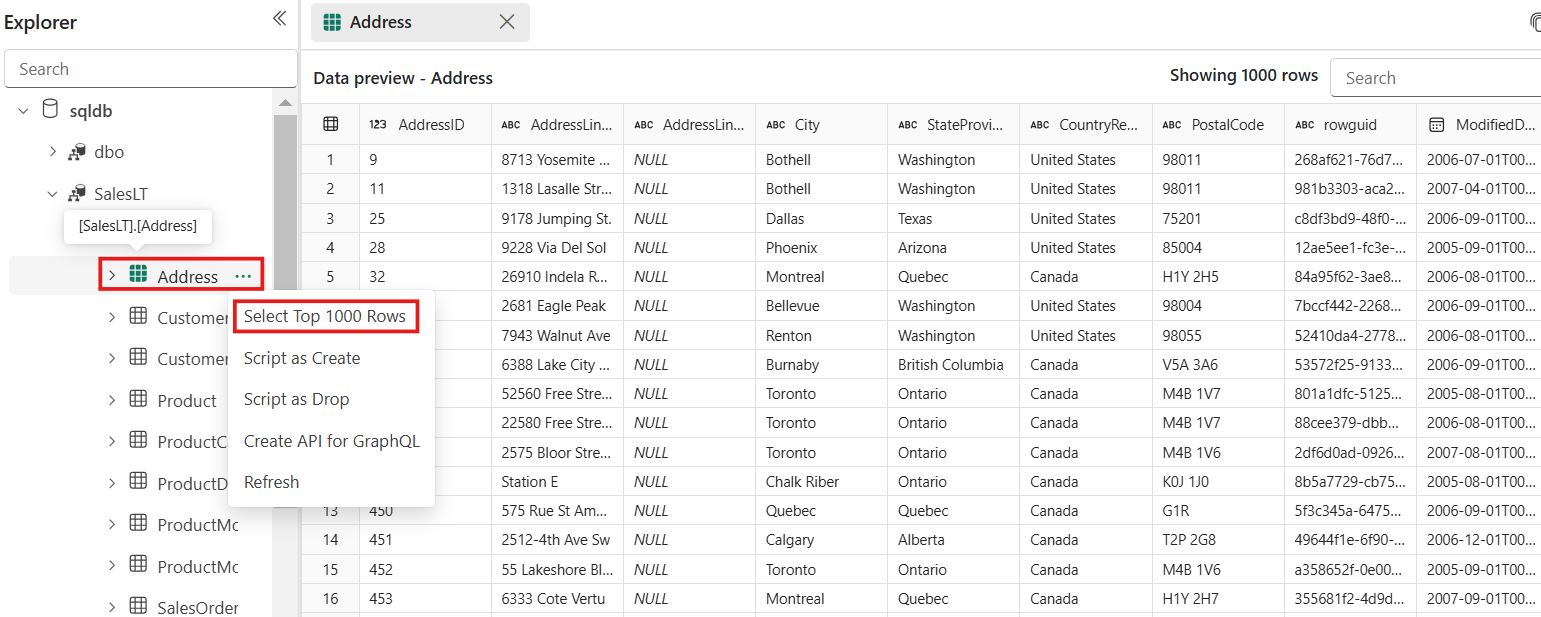


1. You'll see a **Loading Sample Data** notification.
   * Don't modify the database while the import is in process.



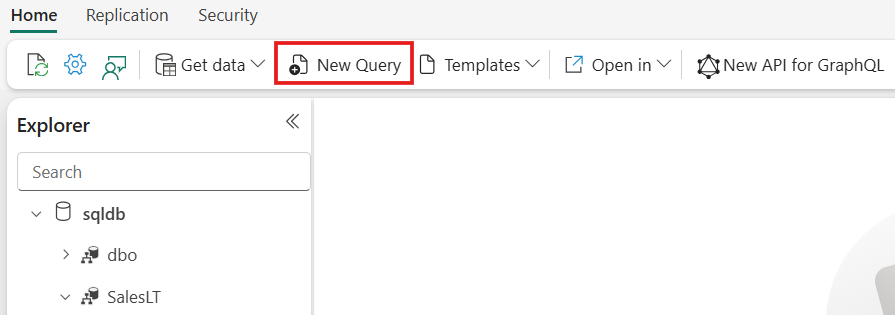
1. Once complete, there's a notification. The object explorer also refreshes to show the new **SalesLT** schema. You're now ready to get started with the AdventureWorksLT sample database.



1. Expand the SalesLT schema in the **Object Explorer** to see the objects that were created. Select on any of the tables to quickly view the data.
   * For more options, like select top 100 rows or to script an object out, right-click or select the context menu (...) of the object name.

## Exercise 5 – Create a table in SQL Database in Fabric

### **Task-1: Create a table with T-SQL Queries**

1. Open your SQL database.
2. Select the **New Query** button in the main ribbon.
3. Create the definition of your table in T-SQL with the help of this sample:

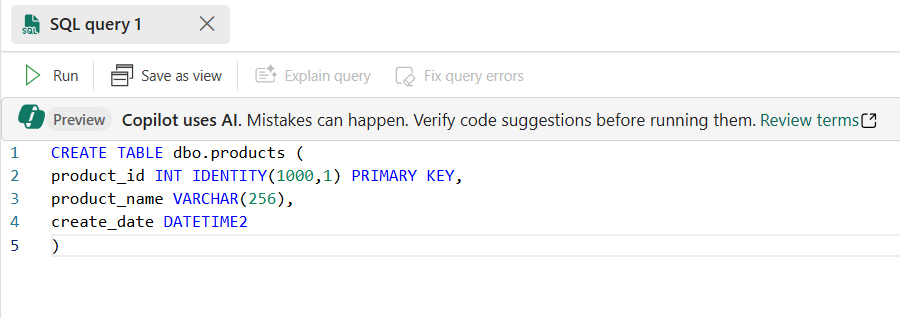
**SQL**

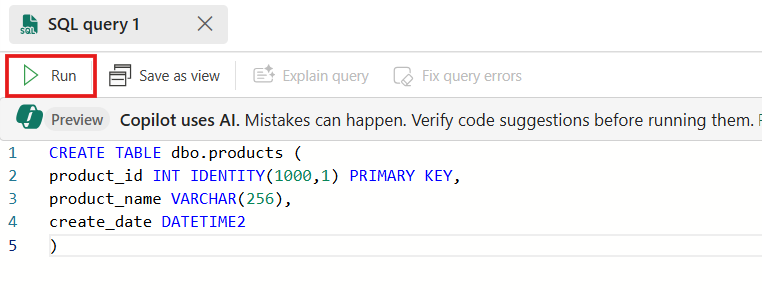
CREATE TABLE dbo.products (

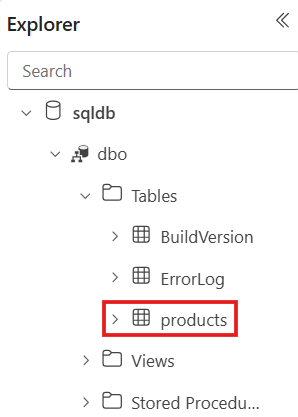
product\_id INT IDENTITY(1000,1) PRIMARY KEY,

product\_name VARCHAR(256),

create\_date DATETIME2



1. Once you have a table design you like, select **Run** in the toolbar of the query window.
2. If the **Object Explorer** is already expanded to show tables, it will automatically refresh to show the new table upon create. If not, expand the tree to see the new table.



### **Task-2: Creating a table with Copilot**

1. Open your SQL database.
2. Select the **New Query** button in the main ribbon.
3. Type in the following text as a T-SQL comment into the query window and press **Tab** on your keyboard:

**SQL**

--create a new table that to store information about products with some typical columns and a monotonistically increasing primary key called ProductID

1. After a few seconds, Copilot will generate a suggested T-SQL script based on the prompt.
2. Press the **Tab** key again to accept Copilot's suggestion. It should look something like this:

**SQL**

--create a new table that to store information about products with some typical columns and a monotonistically increasing primary key called ProductID

CREATE TABLE [dbo].[ProductInformation] (

-- Primary Key for the ProductInformation table

[ProductID] INT IDENTITY(1,1) PRIMARY KEY,

-- Name of the product

[ProductName] VARCHAR(100) NOT NULL,

-- Description of the product

[Description] VARCHAR(MAX),

-- Brand of the product

[Brand] VARCHAR(50),

-- List price of the product

[ListPrice] DECIMAL(10, 2),

-- Sale price of the product

[SalePrice] DECIMAL(10, 2),

-- Item number of the product

[ItemNumber] VARCHAR(20),

-- Global Trade Item Number of the product

[GTIN] VARCHAR(20),

-- Package size of the product

[PackageSize] VARCHAR(50),

-- Category of the product

[Category] VARCHAR(50),

-- Postal code related to the product

[PostalCode] VARCHAR(10),

-- Availability of the product

[Available] BIT,

-- Embedding data of the product

[Embedding] VARBINARY(MAX),

-- Timestamp when the product was created

[CreateDate] DATETIME

);

1. Review and edit Copilot's suggested T-SQL to better fit your needs.
2. Once you have a table design you like, select **Run** in the toolbar of the query window.
3. If the **Object Explorer** is already expanded to show tables, it will automatically refresh to show the new table upon create. If not, expand the tree to see the new table.

## Exercise 6 – Query your SQL database in Fabric

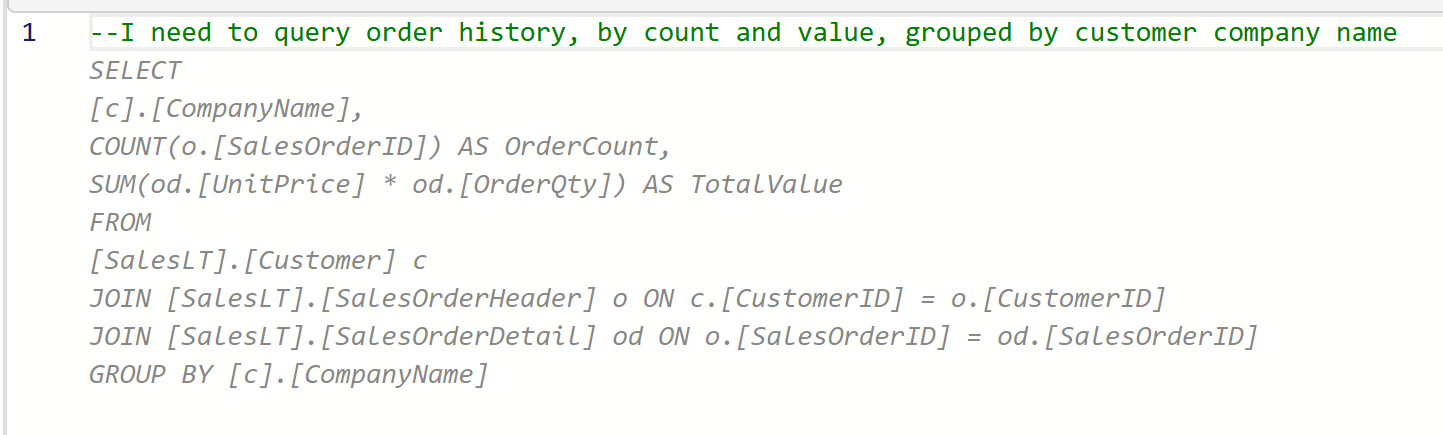
### **Task-1: Ask Copilot to write a query**

Copilot can also be used to write a query, and will translate [natural language prompts to T-SQL](https://learn.microsoft.com/en-us/fabric/database/sql/copilot-chat-pane).

1. Start with a database with some data. For example, try the AdventureWorks sample data.
2. Select **New Query** in the ribbon.
3. Type the following text as a T-SQL comment, then press the tab key:

**SQL**

--I need to query order history, by count and value, grouped by customer company name

1. [](https://learn.microsoft.com/en-us/fabric/database/sql/media/query/t-sql-query-with-copilot.png#lightbox)You should see a suggestion like this:
2. Press the tab key to accept the suggested T-SQL code.
3. Review the query.
4. Select **Run** from the toolbar of the query window and review the results.

## Exercise 7 – Query the SQL Analytics Endpoint of your SQL Database in Fabric

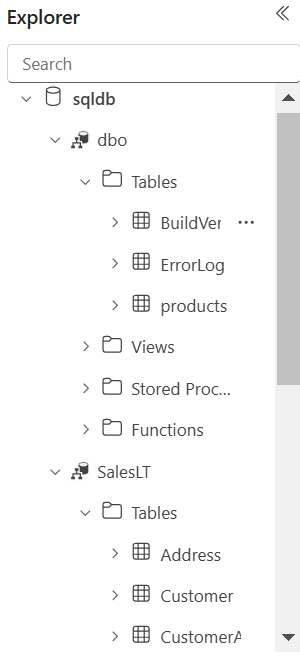
### **Task-1: Access the SQL Analytics Endpoint**

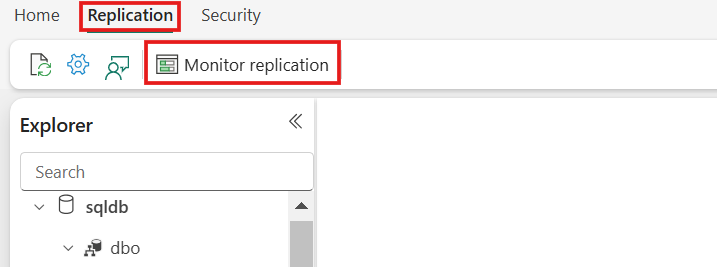
The SQL analytics endpoint can be queried with T-SQL multiple ways:

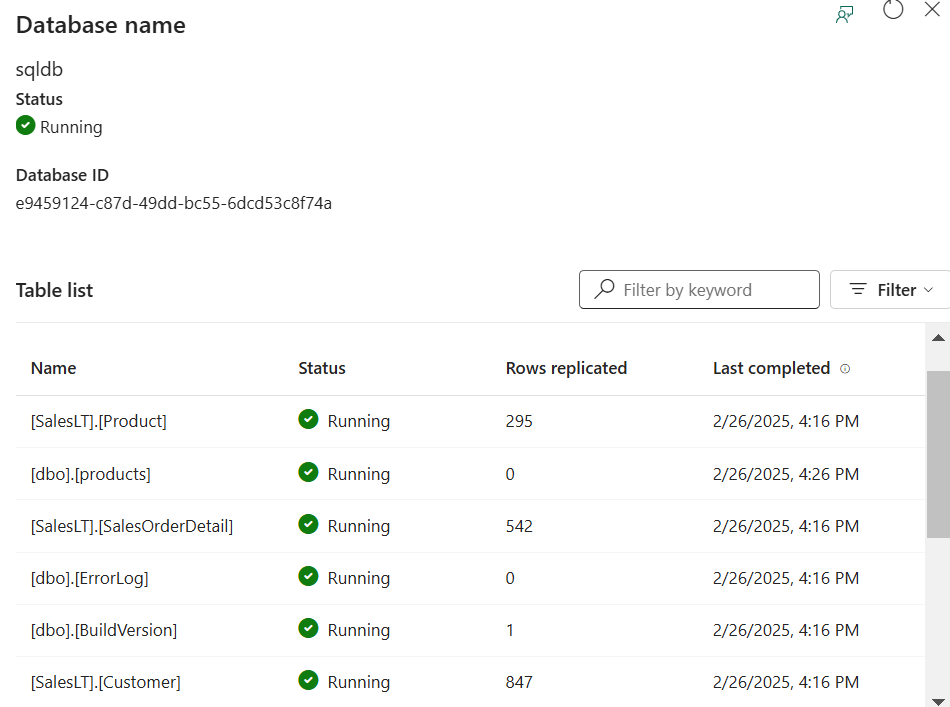
* The first is via the workspace. Every SQL database is paired with a default semantic model and a SQL analytics endpoint. The semantic model and the SQL analytics endpoint always show up together with the SQL database in item listing of the workspace. You can access any of them by selecting them by name from the list.
* The SQL analytics endpoint can also be accessed from within the SQL query editor. This can be especially useful when toggling between the database and the SQL analytics endpoint. Use the pulldown in the upper right corner to change from the editor to the analytics endpoint.
* The SQL analytics endpoint also has its own SQL connection string if you want to query it directly from tools like [SQL Server Management Studio](https://learn.microsoft.com/en-us/fabric/database/sql/connect#connect-with-sql-server-management-studio-manually) or [the mssql extension with Visual Studio Code](https://learn.microsoft.com/en-us/sql/tools/visual-studio-code/mssql-extensions?view=fabric&preserve-view=true). To get the connection strings, see [Find SQL connection strings](https://learn.microsoft.com/en-us/fabric/database/sql/connect#find-sql-connection-string).

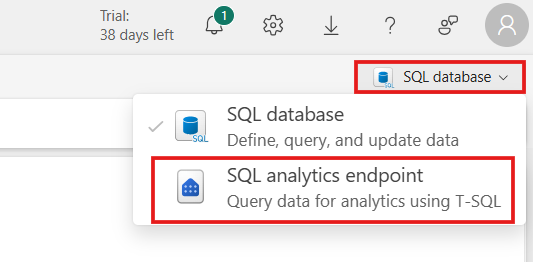
### **Task-2: Query the SQL Analytics Endpoint**

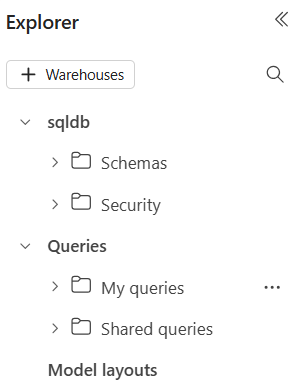
1. Open an existing database that was loaded with the sample data.
2. Expand the **Object Explorer** and make note of the tables in the database.

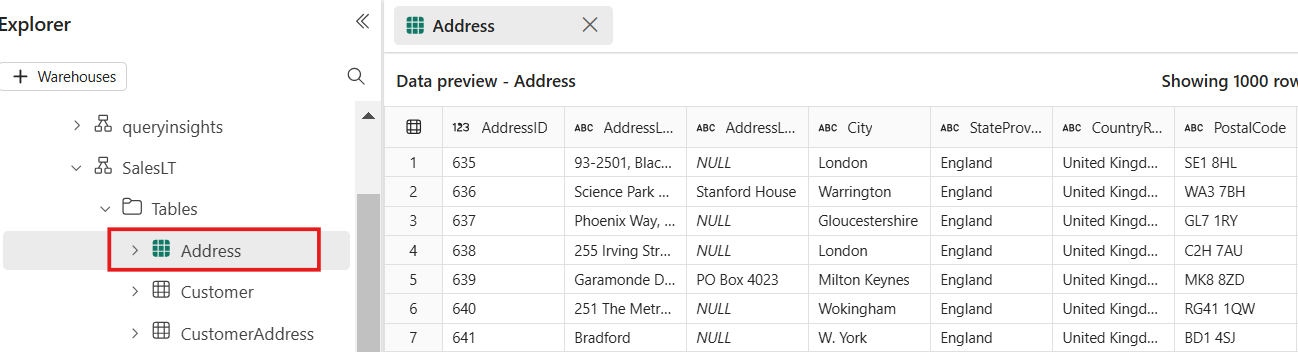


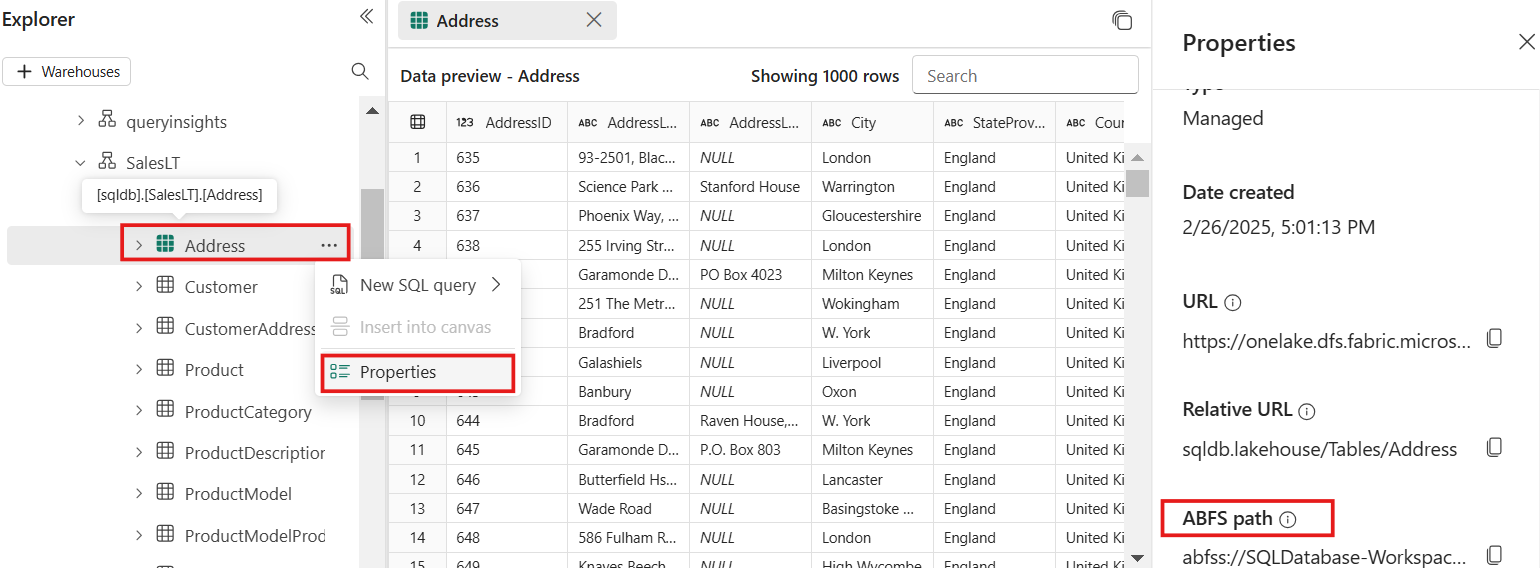
1. Select the replication menu at the top of the editor, select **Monitor Replication**.
2. A list containing the tables in the database will appear. If this is a new database, you'll want to wait until all of the tables have been replicated. There is a refresh button in the toolbar. If there are any problems replicating your data, it is displayed on this page.

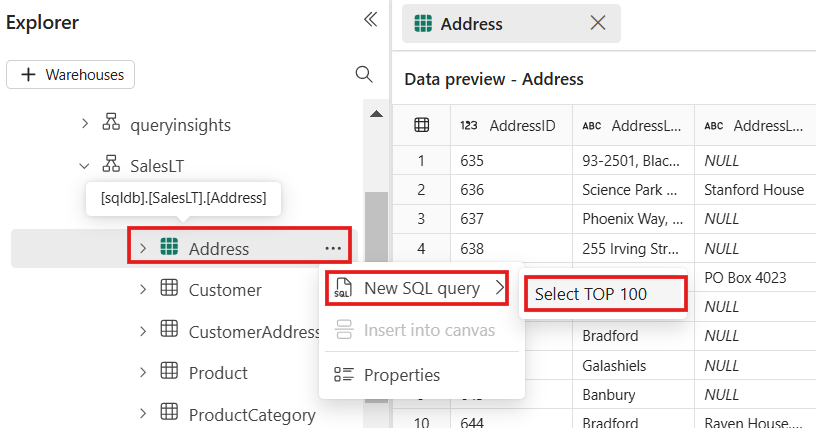


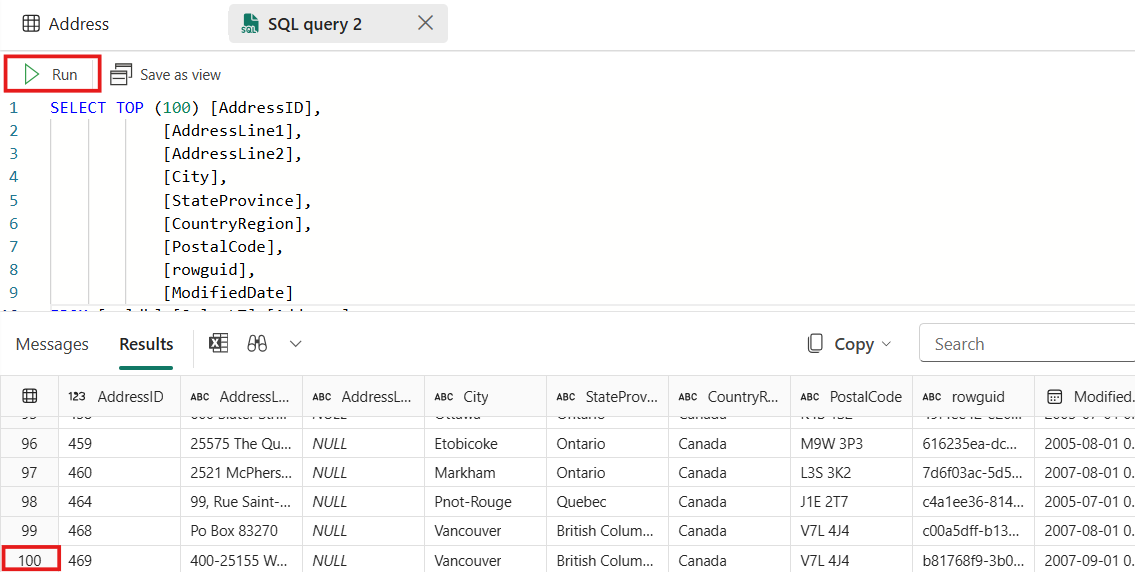
1. Once your tables are replicated, close the **Monitor Replication** page.
2. Select the SQL analytics endpoint from the dropdown in the SQL query editor.
3. You now see that the **Object Explorer** changed over to the warehouse experience.



1. nSelect some of your **tables** to see the data appear, reading directly from OneLake.
2. Select the context menu (...) for any table, and select **Properties** from the menu. Here you can see the OneLake information and ABFS file path.



1. Close the **Properties** page and select the context menu (...) for one the tables again. Select **New Query** and **SELECT TOP 100**.
2. Run the query to see the top 100 rows of data, queried from the SQL analytics endpoint, a copy of the database in OneLake.



1. If you have other databases in your workspace, you can also run queries with cross-database joins. Select the **+ Warehouse** button in the **Object Explorer** to add the SQL analytics endpoint for another database. You can write T-SQL queries similar to the following that join different Fabric data stores together:

**SQL**

SELECT TOP (100) [a.AccountID],

[a.Account\_Name],

[o.Order\_Date],

[o.Order\_Amount]

FROM [Contoso Sales Database].[dbo].[dbo\_Accounts] a

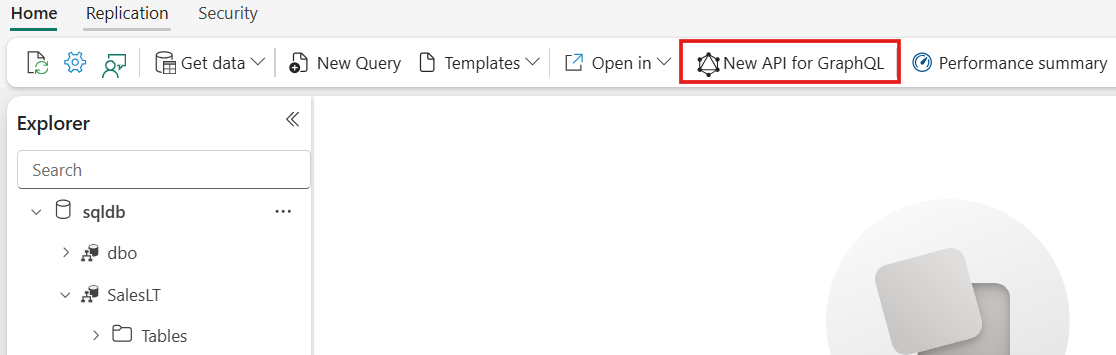
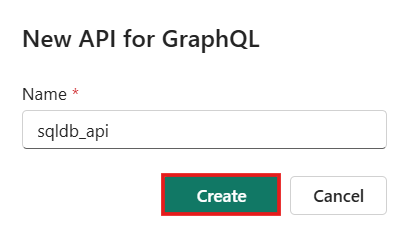
INNER JOIN [Contoso Order History Database].[dbo].[dbo\_Orders] o

ON a.AccountID = o.AccountID;

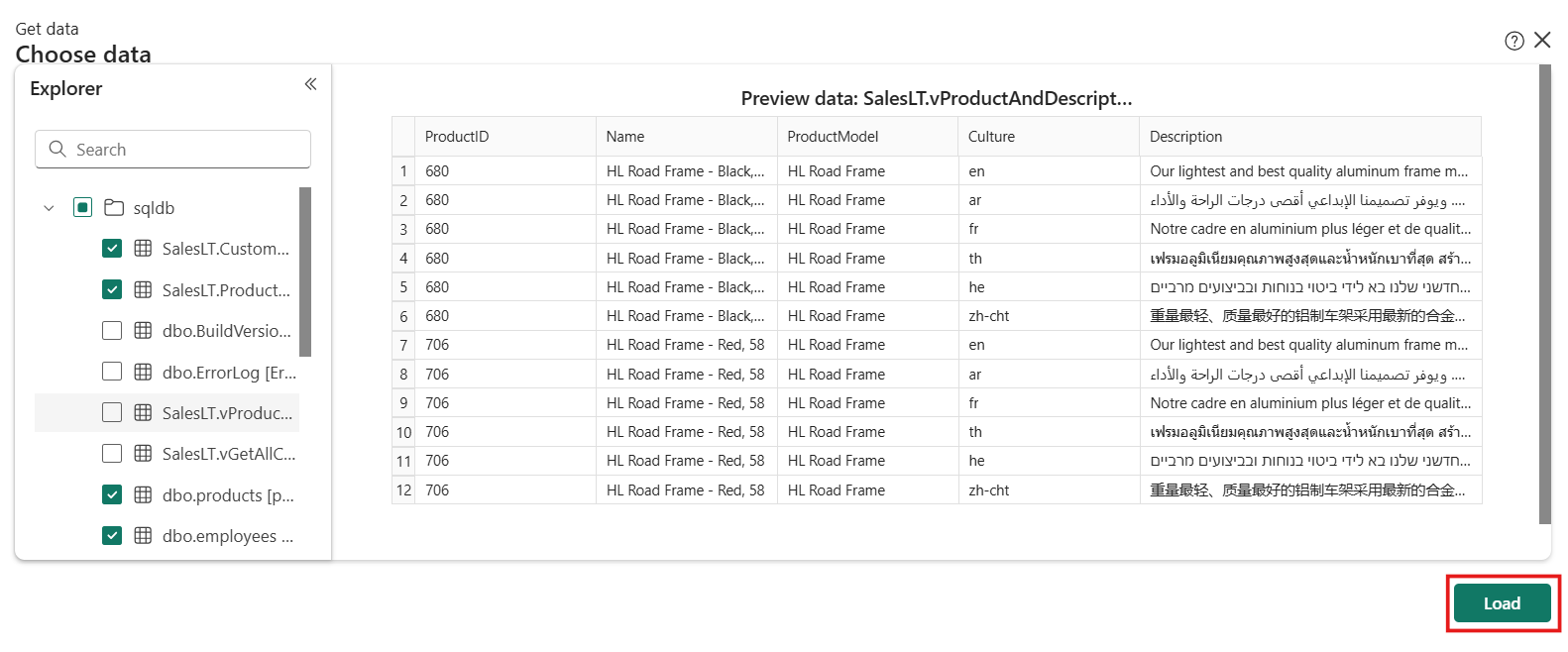
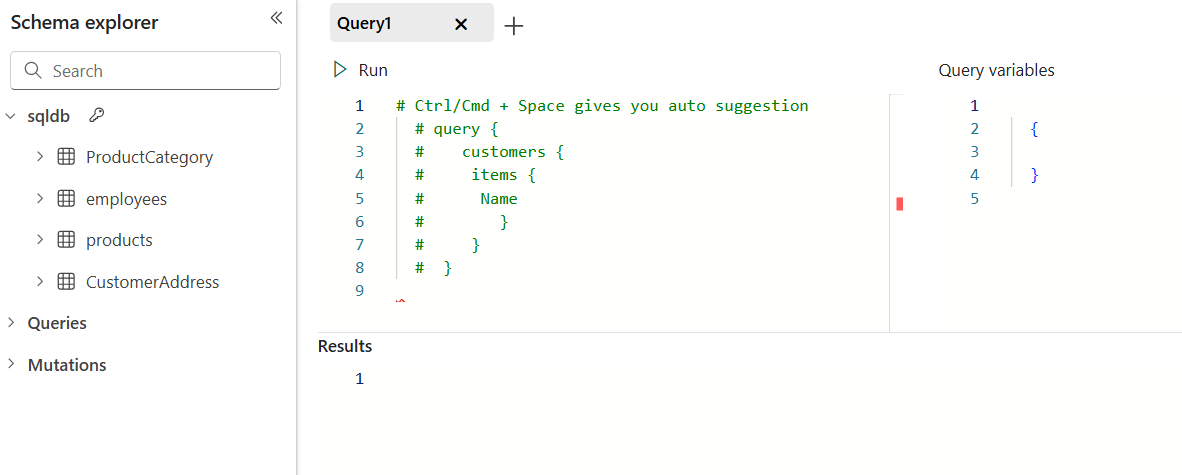
1. Next, select the **New Query** dropdown from the toolbar, and choose **New SQL query in notebook**
2. Once in the notebook experience, select context menu (...) next to a table, then select **SELECT TOP 100**.
3. To run the T-SQL query, select the play button next to the query cell in the notebook.

## Exercise 8 – Create GraphQL API from your SQL Database

To create an API for GraphQL:

1. Open the database where you want to create a GraphQL API.
2. Select **New** **API for GraphQL** from the toolbar.
3. Enter a **Name** for your item and select **Create**.

At this point, the API is ready but it's not exposing any data. APIs for GraphQL are defined in a schema organized in terms of types and fields, in a strongly typed system. Fabric automatically generates the necessary GraphQL schema based on the data you choose to expose to GraphQL clients.

1. The **Choose data** screen allows you to search and choose the objects you want exposed in your GraphQL schema. Select the checkboxes next to the individual tables or stored procedures you want to expose in the API. To select all the objects in a folder, select the checkbox with the data source name at the top.
2. Select **Load** to start the GraphQL schema generation process.
3. The schema is generated, and you can start prototyping GraphQL queries (read, list) or mutations (create, update, delete) to interact with your data. The following image shows the **Schema explorer** with an API call template.

Your API for GraphQL is now ready to accept connections and requests. You can use the API editor to test and prototype GraphQL queries and the Schema explorer to verify the data types and fields exposed in the API.

## Exercise 9 – Create simple reports on your SQL database in Power BI

1. Once the sample data is loaded, switch over to the SQL analytics endpoint via the dropdown menu in the Query editor.
2. Select the **Reporting** menu from the ribbon, then **New Report**.
3. Choose **Continue** to create a new report with all available data.
4. Once Power BI opens:
   * Create a new report by dragging data into the report.
   * There's also the option to use Copilot. That is what we use for this tutorial. Select the **Copilot** button in the toolbar.
5. Copilot presents some reporting suggestions. Select **Suggest content for a new report page**.
6. Review the selections by expanding their cards. Select **Create** on any and all suggestions that look interesting. A new tab is created at the bottom for each.
7. You now have a Power BI report of your SQL database data. Review the reports for accuracy and make any desired modifications.

## Exercise 10 – Share data and manage access to your SQL database in Microsoft Fabric

1. Open your workspace containing the database in the Fabric portal.
2. In the list of items, or in an open item, select the **Share** button. Screenshot of the Share button in the Fabric portal.
3. The **Grant people access** dialog opens. Enter the names of the people or groups that need access.

The dialog offers a few simple options to grant broad access to the SQL database for scenarios where a database has been created for a single user or purpose. We'll skip checking any of those boxes here.

1. Choose whether to notify recipients with an email and add a message.
2. Select **Grant**.
3. The users now have access to connect to the database but are unable to do anything yet. The users can be added to SQL roles by selecting **Manage SQL security** from the **Security** menu in the database editor.
4. Select the db\_datareader role and then **Manage Access**.
5. Add the users to the role and select **Save**.
6. Select the db\_datawriter role and then **Manage Access**.
7. Add the users to the role and select **Save**.

The users now have access to read and write every table within the database. They won't have rights on any other Fabric items in the workspace unless they have also been granted. Instead of the broad roles, consider that users could be granted rights on individual tables to follow the principle of least privilege.