

Explore Azure SQL Database

Before you start

Provision an
Azure SQL
Database
resource

In this exercise you'll provision an Azure SQL Database resource in your Azure subscription, and then use SQL to query the tables in a relational database.

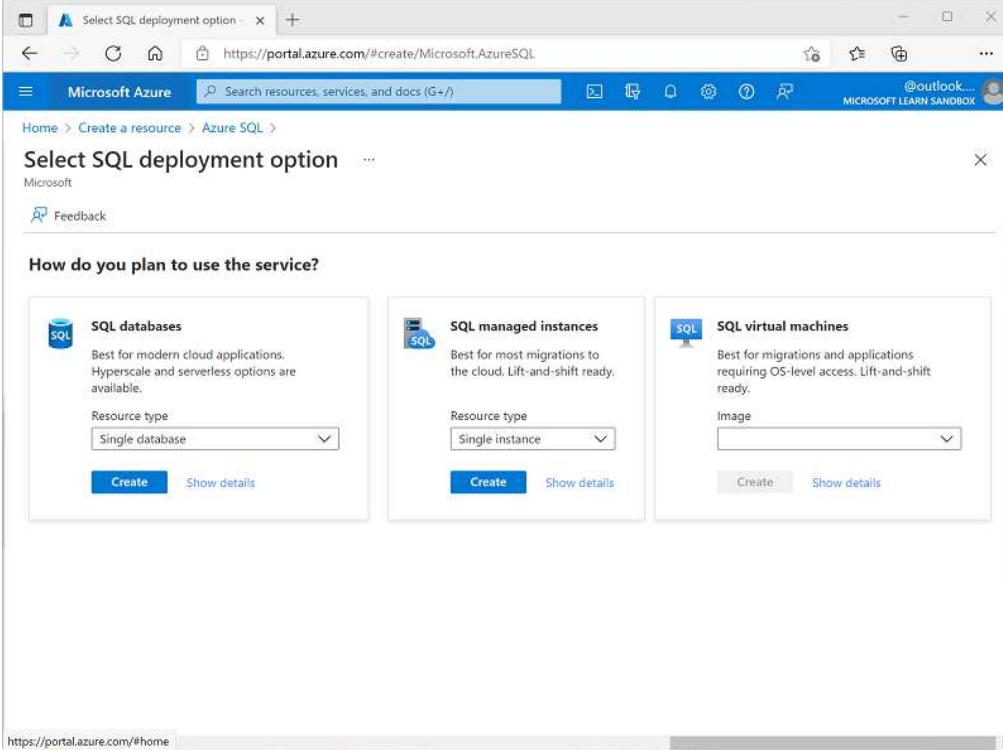
This lab will take approximately **15** minutes to complete.

Before you start

You'll need an [Azure subscription](#) in which you have administrative-level access.

Provision an Azure SQL Database resource

1. In the [Azure portal](#), select **+ Create a resource** from the upper left-hand corner and search for **Azure SQL**. Then in the resulting **Azure SQL** page, select **Create**.
2. Review the Azure SQL options that are available, and then in the **SQL databases** tile, ensure **Single database** is selected and select **Create**.



3. Enter the following values on the **Create SQL Database** page, and leave all other properties with their default setting:

- **Subscription:** Select your Azure subscription.
- **Resource group:** Create a new resource group with a name of your choice.
- **Database name:** *AdventureWorks*
- **Server:** Select **Create new** and create a new server with a unique name in any available location. Use **SQL authentication** and specify your name as the server admin login and a suitably complex password (remember the password - you'll need it later!)
- **Want to use SQL elastic pool?:** No
- **Workload environment:** Development
- **Compute + storage:** Leave unchanged
- **Backup storage redundancy:** *Locally-redundant backup storage*

4. On the **Create SQL Database** page, select **Next :Networking >**, and on the **Networking** page, in the **Network connectivity** section, select **Public endpoint**. Then select **Yes** for both options in the **Firewall rules** section to allow access to your database server from Azure services and your current client IP address.

5. Select **Next: Security >** and set the **Enable Microsoft Defender for SQL** option to **Not now**.

6. Select **Next: Additional Settings >** and on the **Additional settings** tab, set the **Use existing data** option to **Sample** (this will create a sample database that you can explore later).

7. Select **Review + Create**, and then select **Create** to create your Azure SQL database.

8. Wait for deployment to complete. Then go to the resource that was deployed, which should look like this:

The screenshot shows the Microsoft Azure portal interface for the 'AdventureWorks' database. The left sidebar lists navigation options: Overview, Activity log, Tags, Diagnose and solve problems, Quick start, Query editor (preview), Power Platform (Power BI, Power Apps, Power Automate), Settings (Compute + storage, Connection strings, Properties, Locks), and Data management. The main content area is titled 'AdventureWorks (gmalc-sql/AdventureWorks)'. It displays the following details:

Essentials	
Resource group (move) learn-ef50097a-8759-4218-b365-7a4e7018fe45	Server name gmalc-sql.database.windows.net
Status Online	Elastic pool No elastic pool
Location East US	Connection strings Show database connection strings
Subscription (move) Concierge Subscription	Pricing tier General Purpose Gen5, 2 vCores
Subscription ID 00eb43f9-857f-4e17-8bf5-2a8f98e020b1	Earliest restore point No restore point available
Tags (edit) Click here to add tags	Show data for last: 1 hour 24 hours 7 days
Compute utilization	
	

9. In the pane on the left side of the page, select **Query editor (preview)**, and then sign in using the administrator login and password you specified for your server.

*If an error message stating that the client IP address isn't allowed is displayed, select the **Allowlist IP ...** link at the end of the message to allow access and try to sign in again (you previously added your own computer's client IP address to the firewall rules, but the query editor may connect from a different address depending on your network configuration.)*

The query editor looks like this:

The screenshot shows the Microsoft Azure Query editor (preview) interface. On the left, there's a sidebar with various navigation links such as Overview, Activity log, Tags, Diagnose and solve problems, Quick start, and Query editor (preview). Below these are sections for Power Platform (Power BI, Power Apps, Power Automate), Settings (Compute + storage, Connection strings, Properties, Locks), and Data management. The main area is titled 'AdventureWorks (Graeme)' and contains a 'Query 1' pane with the following content:

```
1 SELECT * FROM SalesLT.Product;
```

Below the query pane, there are 'Results' and 'Messages' tabs, and a search bar. A message box in the center states: "Showing limited object explorer here. For full capability please open SSDT."

10. Expand the **Tables** folder to see the tables in the database.

11. In the **Query 1** pane, enter the following SQL code:

The screenshot shows the Microsoft Azure Query editor (preview) interface. The 'Query 1' pane contains the following SQL code:

```
Sql
SELECT * FROM SalesLT.Product;
```

There is a 'Copy' button next to the code input field.

12. Select **Run** above the query to run it and view the results, which should include all columns for all rows in the **SalesLT.Product** table as shown here:

The screenshot shows the Microsoft Azure Query editor (preview) interface. The 'Tables' folder in the sidebar is expanded, showing a list of tables including 'dbo.BuildVersion', 'dbo.ErrorLog', 'SalesLT.Address', 'SalesLT.Customer', 'SalesLT.CustomerAddress', 'SalesLT.Product', 'SalesLT.ProductCategory', 'SalesLT.ProductDescription', 'SalesLT.ProductModel', 'SalesLT.ProductModelDetail', 'SalesLT.SalesOrderDetail', and 'SalesLT.SalesOrderHeader'. The 'Query 1' pane shows the results of the query 'SELECT * FROM SalesLT.Product;':

ProductID	Name	ProductNumber
680	HL Road Frame - Black, 58	FR-R92B-58
706	HL Road Frame - Red, 58	FR-R92R-58
707	Sport-100 Helmet, Red	HL-US09-R

A message at the bottom of the results pane says 'Query succeeded | 1s'.

13. Replace the SELECT statement with the following code, and then select **Run** to run the new query and review the results (which includes only the **ProductID**, **Name**, **ListPrice**, **ProductCategoryID** columns):

Sql

Copy

```
SELECT ProductID, Name, ListPrice, ProductCategoryID
FROM SalesLT.Product;
```

14. Now try the following query, which uses a JOIN to get the category name from the **SalesLT.ProductCategory** table:

Sql

Copy

```
SELECT p.ProductID, p.Name AS ProductName,
       c.Name AS Category, p.ListPrice
  FROM SalesLT.Product AS p
 JOIN [SalesLT].[ProductCategory] AS c
    ON p.ProductCategoryID = c.ProductCategoryID;
```

15. Close the query editor pane, discarding your edits.

Tip: If you've finished exploring Azure SQL Database, you can delete the resource group that you created in this exercise.