

Exercise - Create an Azure SQL Database

 coursera.org/learn/microsoft-azure-cloud-services/supplement/6Q5eN/exercise-create-an-azure-sql-database

Tailwind Traders has chosen Azure SQL Database for part of its migration. You've been tasked with creating the database.

In this exercise, you'll create a SQL database in Azure and then query the data in that database.

Task 1: Create the database

In this task, you create a SQL database based on the *AdventureWorksLT* sample database.

Task 1: Create the database

In this task, you create a SQL database based on the *AdventureWorksLT* sample database.

1. Sign in to the Azure portal.
2. Select **Create a resource > Databases > SQL database**.
3. On the **Basics** tab, fill in the following information:

Setting	Value
Subscription	Choose Concierge Subscription
Resource Group	Choose [sandbox resource group name]
Database name	db1

4. For the **Server**, select **Create new**. i. Enter the following information (replace **nnnn** in the name of the server with letters and digits, such that the name is globally unique)

Setting	Value
Server Name	sqlservernnn (must be unique)
Server admin login	sqluser

Setting	Value
Password	Pa\$\$w0rd1234
Location	(US) East US

The screenshot shows the 'Create SQL Database' wizard in the Azure portal. A 'New server' dialog box is open, allowing configuration of a new logical server. The settings in the dialog are: Server name: sqlserver4321, Server admin login: sqluser, Password: (masked), Confirm password: (masked), and Location: (US) East US. The background shows the 'Create SQL Database' page with the 'Database details' section partially visible, showing 'Database name' as db1 and 'Server' as (new) sqlserver4321 (East US).

ii. Select OK when you have finished.

5. Select the **Next: Networking** > at the bottom, and configure the following settings (leave others with their defaults):

Setting	Value
Server Name	sqlservernnn (must be unique)
Server admin login	sqluser
Password	Pa\$\$w0rd1234
Location	(US) East US

6. Select the **Next: Networking** > at the bottom, and configure the following settings (leave others with their defaults):

Setting	Value
Connectivity Method	Public Endpoint (Default)

The screenshot shows the 'Create SQL Database' page in the Azure portal. The breadcrumb navigation at the top reads 'Home > New > Create SQL Database'. The page title is 'Create SQL Database' with the Microsoft logo below it. There are four tabs: 'Basics', 'Networking', 'Additional settings' (which is selected and underlined), 'Tags', and 'Review + create'. Below the tabs, a message says 'Customize additional configuration parameters including collation & sample data.' The 'Data source' section has the text 'Start with a blank database, restore from a backup or select sample data to populate your new database.' Below this, 'Use existing data *' is followed by three buttons: 'None', 'Backup', and 'Sample' (which is selected). A note states 'AdventureWorksLT will be created as the sample database.' The 'Database collation' section explains that collation defines sorting and comparison rules and cannot be changed after creation. The default is 'SQL_Latin1_General_CP1_CI_AS', which is also shown in a dropdown menu. The 'Advanced data security' section describes a unified security package and offers a '30 day free trial period'. At the bottom, there are three buttons: 'Review + create' (in blue), '< Previous', and 'Next : Tags >'.

7. Select **Review + create** > **Create** to deploy the server and database. i. It can take approximately 2 to 5 minutes to create the server and deploy the sample database.

8. Select **Go to resource**.

9. Select **Set server firewall** and Allow Azure services and resources to access this server = **Yes**.

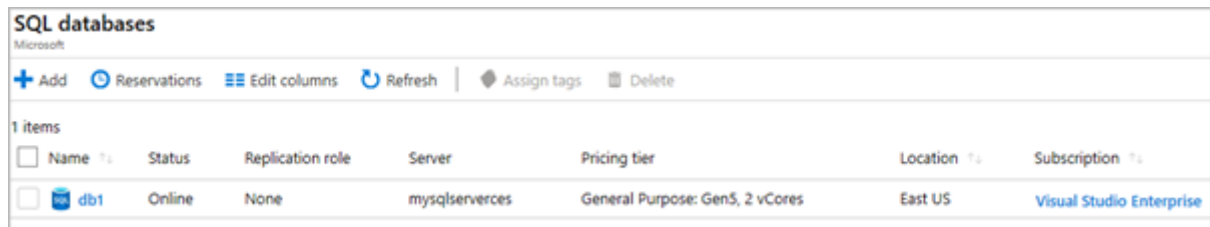
10. Select **Save**.

11. Select **OK**.

Task 2: Test the database

In this task, you configure the server and run a SQL query.

1. From the All resources pan, search and select SQL databases and ensure that your new database was created. You might need to refresh the page.

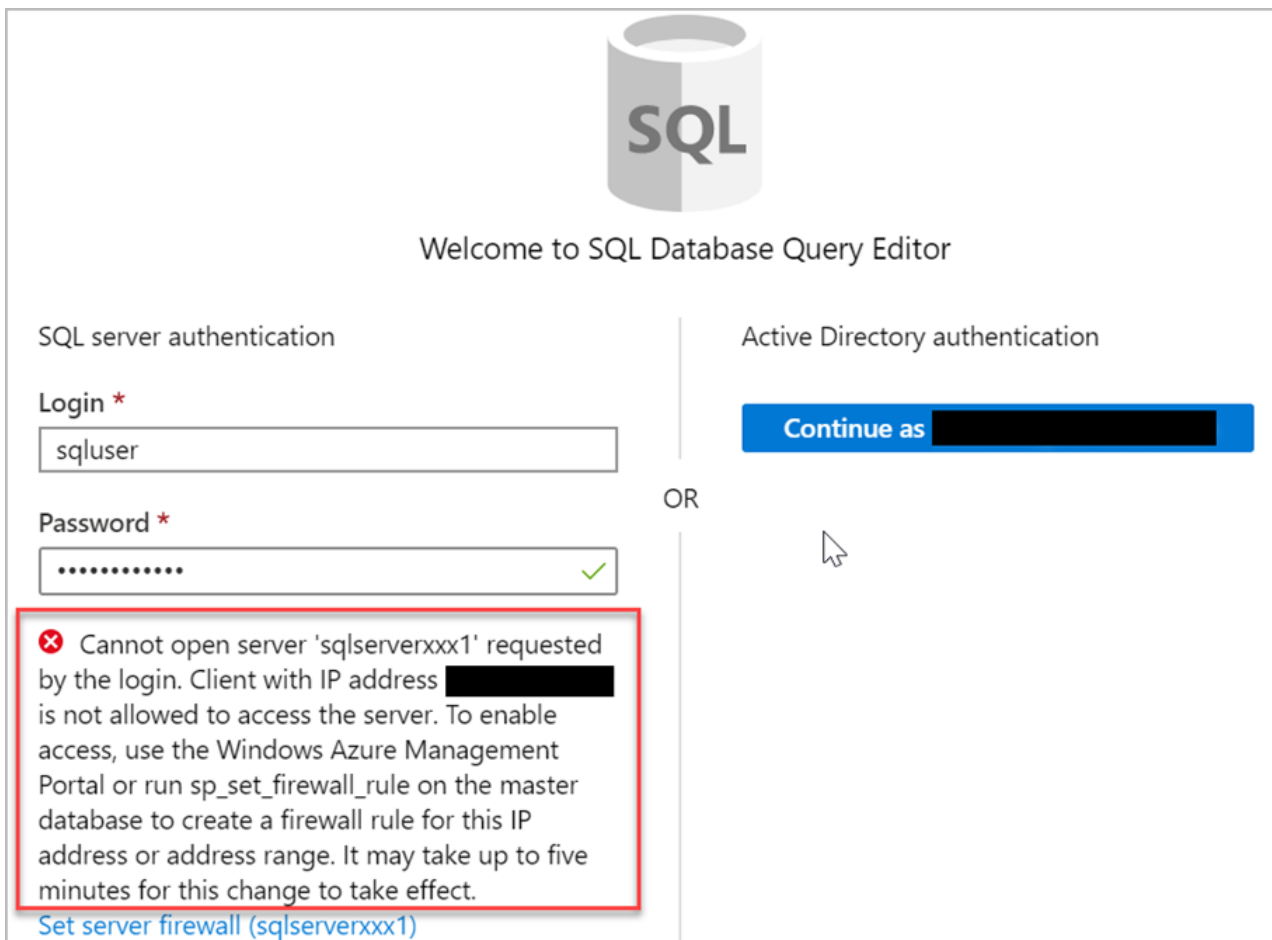


Name	Status	Replication role	Server	Pricing tier	Location	Subscription
db1	Online	None	mysqlserververcs	General Purpose: Gen5, 2 vCores	East US	Visual Studio Enterprise

2. Select the **db1** entry representing the SQL database you created, and then select **Query editor (preview)** on the left side

3. Sign in as **sqluser**, with the password **Pa\$\$word1234**.

4. You will not be able to sign in. Read the error closely and make note of the IP address that needs to be allowed through the firewall.



SQL Database Query Editor

SQL server authentication

Login *

sqluser

Password *

.....

Active Directory authentication

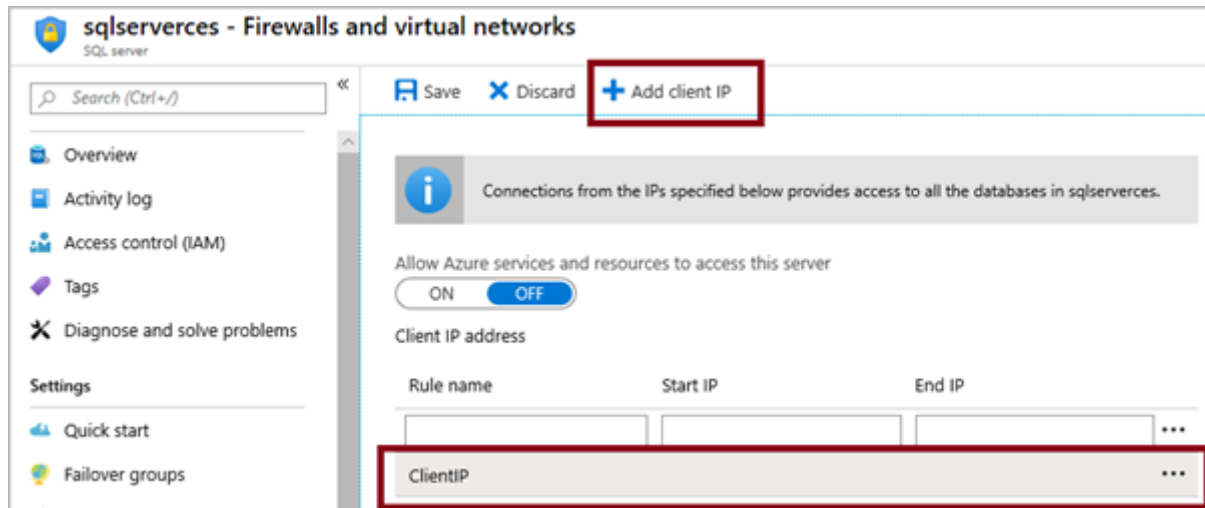
Continue as [redacted]

OR

✖ Cannot open server 'sqlserverxxx1' requested by the login. Client with IP address [redacted] is not allowed to access the server. To enable access, use the Windows Azure Management Portal or run sp_set_firewall_rule on the master database to create a firewall rule for this IP address or address range. It may take up to five minutes for this change to take effect. [Set server firewall \(sqlserverxxx1\)](#)

5. Select **Overview > Set server firewall**.

6. In **Client IP address** your IP will be shown, create a **Rule name** > Add your IP in both **Start IP** and **END IP** and then select **Save**.



7. Return to your SQL database and the Query Editor sign-in page. Try to sign in again as **sqluser**, with the password **Pa\$\$word1234**. This time you should succeed. It might take a couple of minutes for the new firewall rule to be deployed. If you wait and still get an error, try selecting **Firewall settings** > again.

8. After you sign in successfully, the query pane appears. Enter the following query into the editor pane:

```
1  
2  
3  
4  
  
SELECT TOP 20 pc.Name as CategoryName, p.name as ProductName  
FROM SalesLT.ProductCategory pc  
JOIN SalesLT.Product p  
ON pc.productcategoryid = p.productcategoryid;
```



9. Select **Run**, and then review the query results in the **Results** pane. The query should run successfully.

Query 1 X

▶ Run

■ Cancel query

```
1 SELECT TOP 20 pc.Name as CategoryName, p.name as ProductName
2 FROM SalesLT.ProductCategory pc
3 JOIN SalesLT.Product p
4 ON pc.productcategoryid = p.productcategoryid;
```

Results

Messages

🔍 Search to filter items...

CATEGORYNAME	PRODUCTNAME
Road Frames	HL Road Frame - Black, 58
Road Frames	HL Road Frame - Red, 58
Helmets	Sport-100 Helmet, Red
Helmets	Sport-100 Helmet, Black
Socks	Mountain Bike Socks, M

✔ Query succeeded | 1s

Congratulations! You've created a SQL database in Azure and successfully queried the data in that database.