

06 - Create a SQL database

In this walkthrough, we will create a SQL database in Azure and then query the data in that database.

Task 1: Create the database (5 min)

In this task, we will create a SQL database based on the AdventureWorksLT sample database.

1. Sign in to the Azure portal at <https://portal.azure.com>.
2. From the **All services** blade, search for and select **SQL databases**, and then click **+ Add**.
3. On the **Basics** tab, fill in this information.

Setting	Value
Subscription	Choose your subscription
Resource group	myRGDb (create new)
Database name	db1

4. Next to the **Server** drop down list, click **Create new** and enter this information (replace **xxxxx** in the name of the server with letters and digits such that the name is globally unique). Click **OK** when finished.

Setting	Value
Server name	sqlserverxxxxx (must be unique)
Server admin login	sqluser
Password	Pa\$\$w0rd1234
Location	(US) East US
Allow Azure services to access server	Select the checkbox

Home > SQL databases > Create SQL Database

Create SQL Database

Microsoft

[Basics](#)
[Networking](#)
[Additional settings](#)
[Tags](#)
[Review + create](#)

Create a SQL database with your preferred configurations. Complete the Basics tab then go to Review + Create to provision with smart defaults, or visit each tab to customize. [Learn more](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

Azure Pass - Sponsorship

Resource group * ⓘ

(New) myRGDb

[Create new](#)

Database details

Enter required settings for this database, including picking a logical server and configuring the compute and storage resources

Database name *

db1

Server ⓘ

(new) sqlserver4321 (East US)

[Create new](#)

Want to use SQL elastic pool? * ⓘ

☐ Yes ☒ No

Compute + storage * ⓘ

General Purpose

Gen5, 2 vCores, 32 GB storage

[Configure database](#)

Review + create

Next : Networking >

New server

Microsoft

Server name *

sqlserver4321

.database.windows.net

Server admin login *

sqluser

Password *

.....

Confirm password *

.....

Location *

(US) East US

☒ Allow Azure services to access server ⓘ

OK

5. Move to the **Networking** tab and configure the following settings (leave others with their defaults)

Setting	Value
Connectivity method	Public endpoint
Allow Azure services and resources to access this server	Yes
Add current client IP address	No

Create SQL Database

Microsoft

[Basics](#) [Networking](#) [Additional settings](#) [Tags](#) [Review + create](#)

Configure network access and connectivity for your server. The configuration selected below will apply to the selected server 'sqlserver4321' and all databases it manages. [Learn more](#)

Network connectivity

Choose an option for configuring connectivity to your server via public endpoint or private endpoint. Choosing no access creates with defaults and you can configure connection method after server creation. [Learn more](#)

Connectivity method * ⓘ

☐ No access

☒ Public endpoint

☐ Private endpoint

Firewall rules

Setting 'Allow Azure services and resources to access this server' to Yes allows communications from all resources inside the Azure boundary, that may or may not be part of your subscription. [Learn more](#)

Setting 'Add current client IP address' to Yes will add an entry for your client IP address to the server firewall.

Allow Azure services and resources to access this server *

Add current client IP address *

[Review + create](#)

< Previous

Next : Additional settings >

6. Move to the **Additional settings** tab. We will be using the AdventureWorksLT sample database.

Setting	Value
Use existing data	Sample
Collation	<i>use default</i>
Enable advanced data security	Not now

Home > New > Create SQL Database

Create SQL Database

Microsoft

Basics Networking **Additional settings** Tags Review + create

Customize additional configuration parameters including collation & sample data.

Data source

Start with a blank database, restore from a backup or select sample data to populate your new database.

Use existing data *

None Backup **Sample**

AdventureWorksLT will be created as the sample database.

Database collation

Database collation defines the rules that sort and compare data, and cannot be changed after database creation. The default database collation is SQL_Latin1_General_CP1_CI_AS. [Learn more](#)

Collation ⓘ SQL_Latin1_General_CP1_CI_AS

Advanced data security

Protect your data using advanced data security, a unified security package including data classification, vulnerability assessment and advanced threat protection for your server. [Learn more](#)

Get started with a 30 day free trial period, and then 15 USD/server/month.

Enable advanced data security * ⓘ Start free trial **Not now**

Review + create < Previous Next : Tags >

- Click **Review + create** and then click **Create** to deploy and provision the resource group, server, and database. It can take approx. 2 to 5 minutes to deploy.
- Go to the resource tab to locate the SQL database you created. You may need to refresh.

Task 2: Test the database.

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

- From the **All services** blade, search and select **SQL databases** and ensure your new database was created. You may need to **Refresh** the page.

SQL databases

Microsoft

+ Add ⌚ Reservations ≡ Edit columns ↺ Refresh | ⚙ Assign tags 🗑 Delete

1 items

<input type="checkbox"/>	Name ⓘ	Status	Replication role	Server	Pricing tier	Location ⓘ	Subscription ⓘ
<input type="checkbox"/>	db1	Online	None	mysqlserverces	General Purpose: Gen5, 2 vCores	East US	Visual Studio Enterprise

- Click the **db1** entry representing the SQL database you created, and then click **Query editor (preview)**.
- Login as **sqluser** with the password **Pa\$\$w0rd1234**.
- You will not be able to login. Read the error closely and make note of the IP address that needs to be allowed through the firewall.



Welcome to SQL Database Query Editor

SQL server authentication

Login *

sqluser

Password *

..... ✓

✖ 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 \(sqlserverxxx1\)](#)

Active Directory authentication

Continue as [redacted]

OR




5. From the **db1** blade, click **Overview**.




6. From the SQL server **Overview** blade, click **Set server firewall**.

7. Click **Add client IP** (top menu bar) to add the IP address referenced in the error. Be sure to **Save** your changes.

Firewall settings
sqlserver4321 (SQL server)


 Save  Discard  Add client IP

Deny public network access ⓘ Yes **No**

 Setting to **Yes** allows connections via approved private endpoint only and disables any existing firewall rules. [Learn more.](#)

Connection Policy ⓘ **Default** Proxy Redirect

Allow Azure services and resources to access this server **Yes** No


 Connections from the IPs specified below provides access to all the databases in sqlserver43210.

Client IP address

Rule name	Start IP	End IP
<input type="text"/>	<input type="text"/>	<input type="text"/> ...
ClientIPAddress_2020-5-...	your IP address	your IP address ...

8. Return to your SQL database and the **Query Editor (Preview)** login page. Try to login again as **sqluser** with the password **Pa\$\$w0rd1234**. This time you should succeed. Note that it may take a couple of minutes for the new firewall rule to be deployed.

9. Once you log in successfully the query pane appears, enter the following query into the editor pane.

Code  Copy

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

db1 - Query editor (preview)

SQL database

Search (Ctrl+F)

Login Edit Data (Preview) + New Query Open query Save query Feedback

db1 (azureuser)

Showing limited object explorer here. For full capability please open SSDT.

Tables Views Stored Procedures

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...

10. Click **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 have created a SQL database in Azure and successfully queried the data in that database.

Note: To avoid additional costs, you can remove this resource group. Search for resource groups, click your resource group, and then click **Delete resource group**. Verify the name of the resource group and then click **Delete**. Monitor the **Notifications** to see how the delete is proceeding.