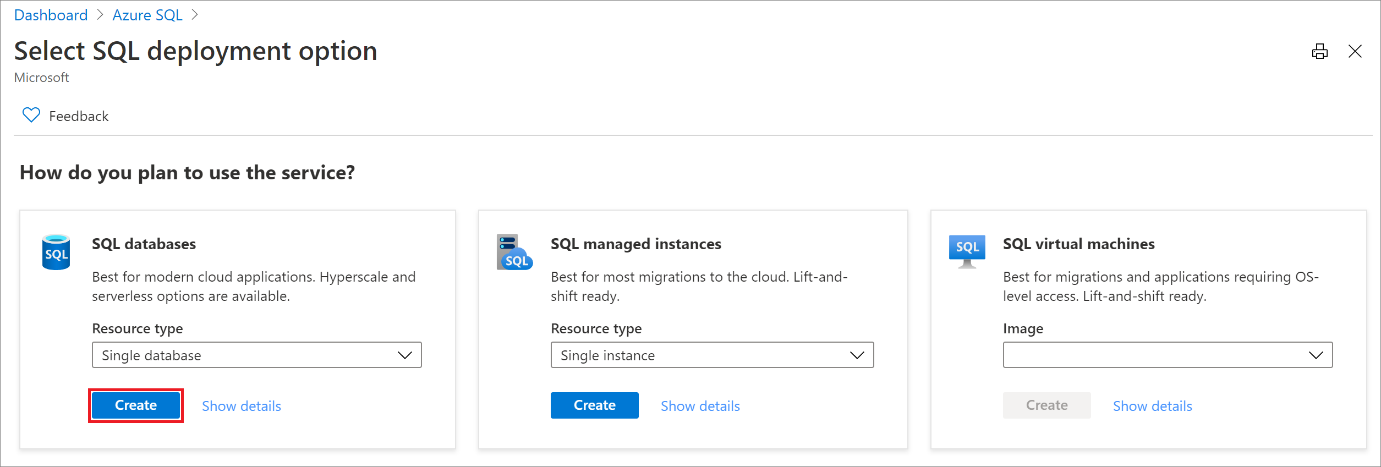
**Create an Azure SQL Database single database**

<https://docs.microsoft.com/en-us/azure/azure-sql/database/single-database-create-quickstart?tabs=azure-portal>

To create a single database in the Azure portal

Browse to the [Select SQL Deployment option](https://portal.azure.com/#create/Microsoft.AzureSQL) page.

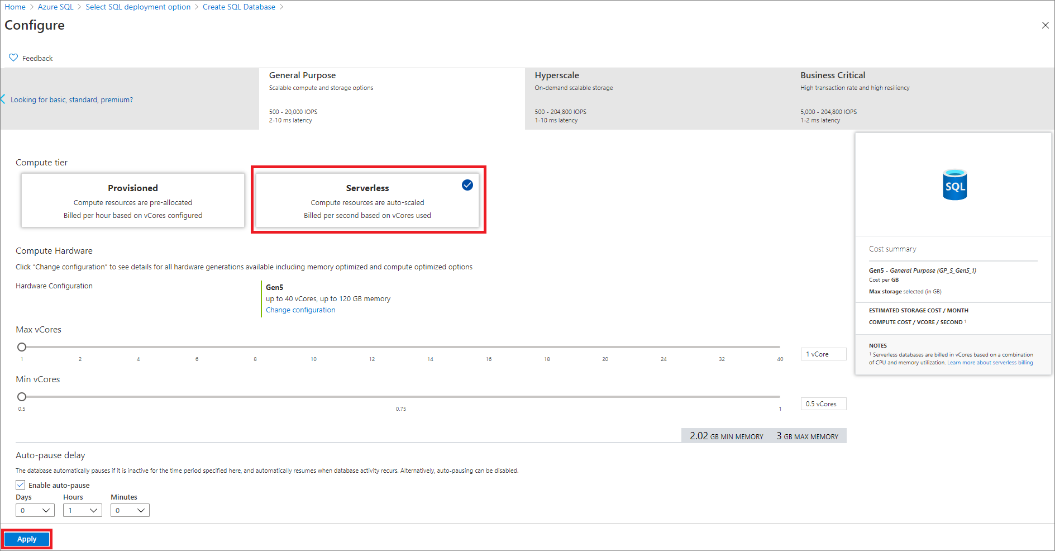
1. Under **SQL databases**, leave **Resource type** set to **Single database**, and select **Create**.



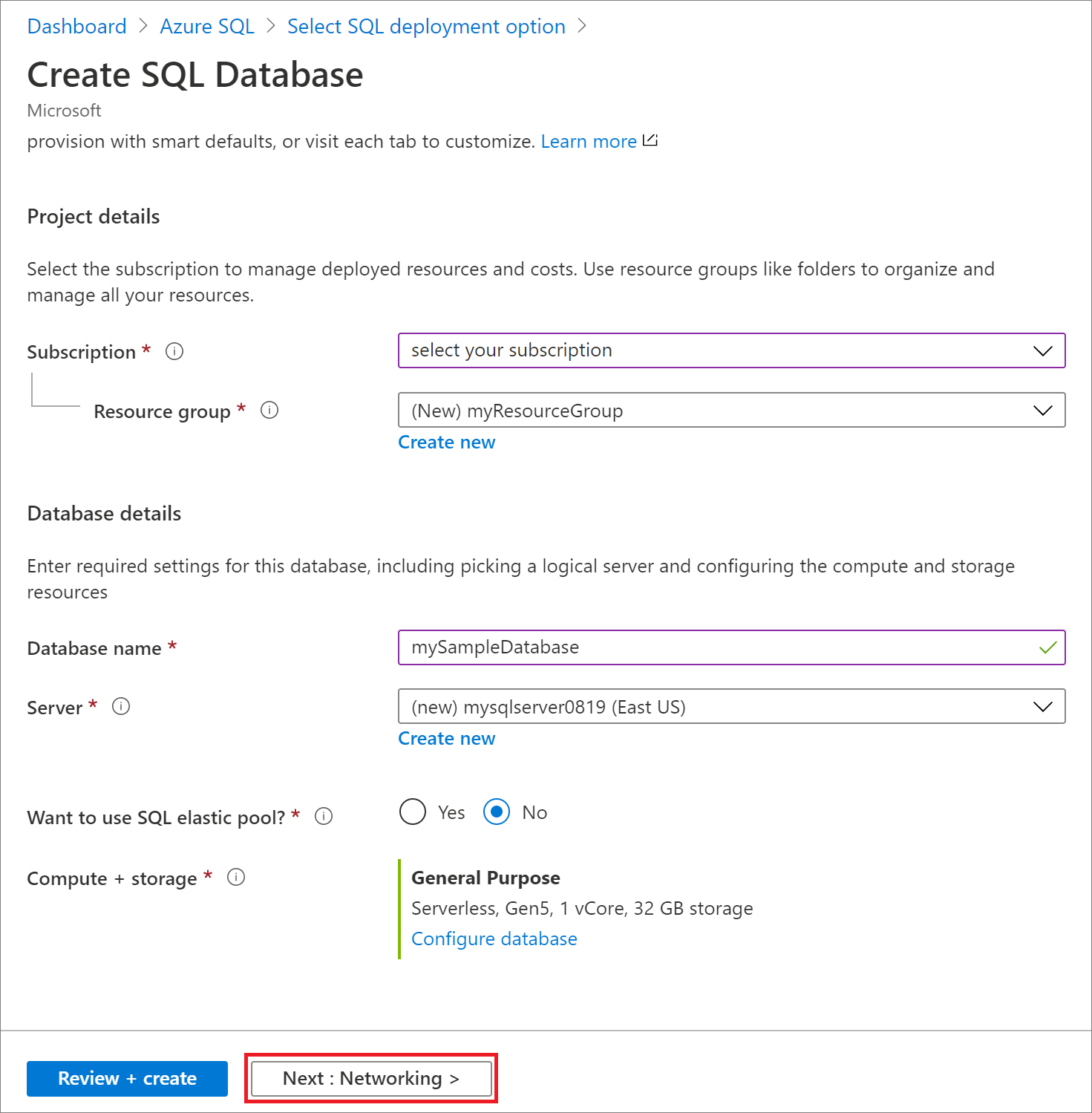
1. On the **Basics** tab of the **Create SQL Database** form, under **Project details**, select the desired Azure **Subscription**.
2. For **Resource group**, select **Create new**, enter myResourceGroup, and select **OK**.
3. For **Database name** enter mySampleDatabase.
4. For **Server**, select **Create new**, and fill out the **New server** form with the following values:
   * **Server name**: Enter mysqlserver, and add some characters for uniqueness. We can't provide an exact server name to use because server names must be globally unique for all servers in Azure, not just unique within a subscription. So enter something like mysqlserver12345, and the portal lets you know if it is available or not.
   * **Server admin login**: Enter azureuser.
   * **Password**: Enter a password that meets requirements, and enter it again in the **Confirm password** field.
   * **Location**: Select a location from the dropdown list.

Select **OK**.

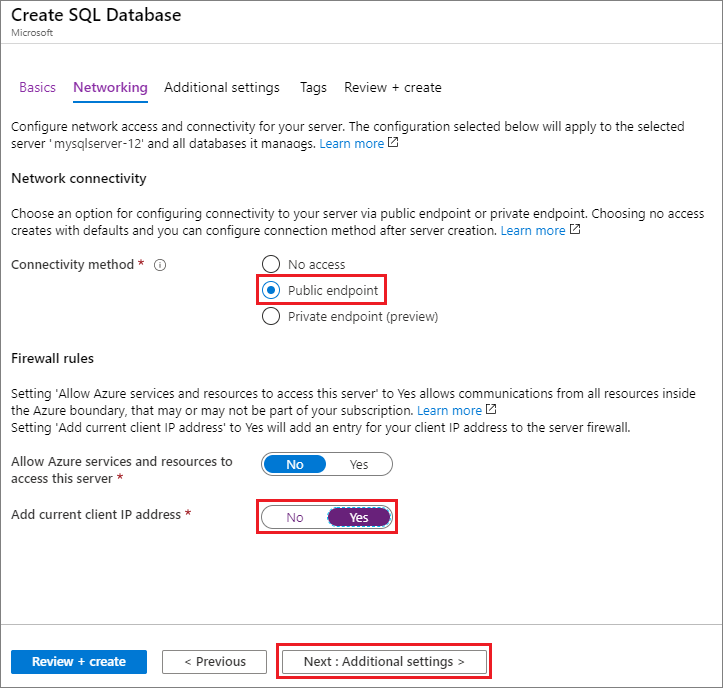
1. Leave **Want to use SQL elastic pool** set to **No**.
2. Under **Compute + storage**, select **Configure database**.
3. This quickstart uses a serverless database, so select **Serverless**, and then select **Apply**.



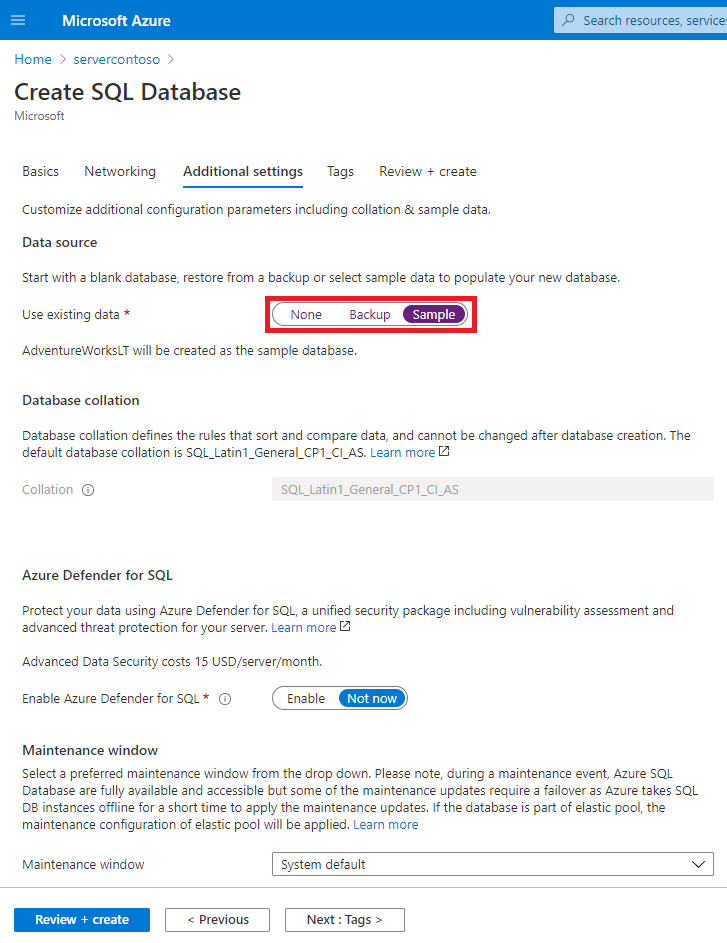
1. Select **Next: Networking** at the bottom of the page.



1. On the **Networking** tab, for **Connectivity method**, select **Public endpoint**.
2. For **Firewall rules**, set **Add current client IP address** to **Yes**. Leave **Allow Azure services and resources to access this server** set to **No**.
3. Select **Next: Additional settings** at the bottom of the page.



1. On the **Additional settings** tab, in the **Data source** section, for **Use existing data**, select **Sample**. This creates an AdventureWorksLT sample database so there's some tables and data to query and experiment with, as opposed to an empty blank database.
2. Optionally, enable [Azure Defender for SQL](https://docs.microsoft.com/en-us/azure/azure-sql/database/azure-defender-for-sql).
3. Optionally, set the [maintenance window](https://docs.microsoft.com/en-us/azure/azure-sql/database/maintenance-window) so planned maintenance is performed at the best time for your database.
4. Select **Review + create** at the bottom of the page:

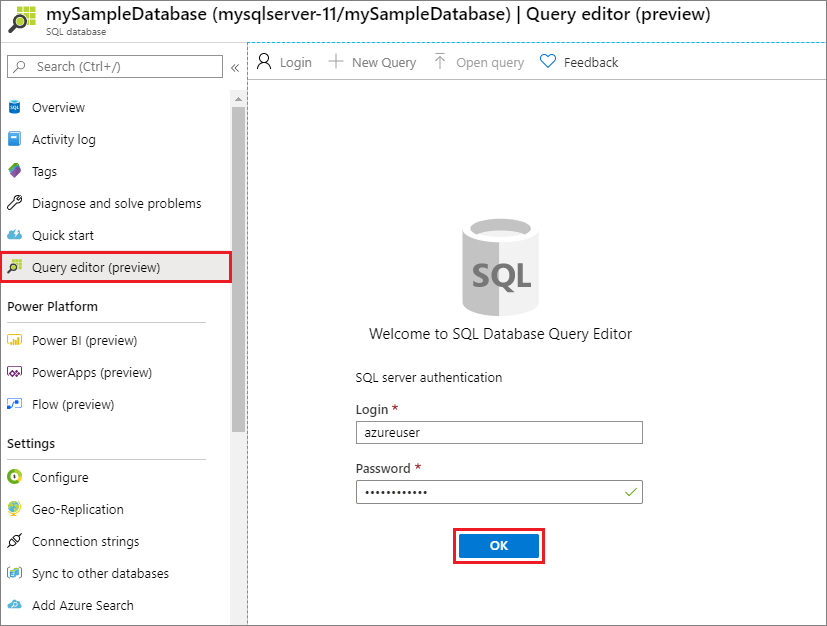


1. On the **Review + create** page, after reviewing, select **Create**.

## Query the database

Once your database is created, you can use the **Query editor (preview)** in the Azure portal to connect to the database and query data.

1. In the portal, search for and select **SQL databases**, and then select your database from the list.
2. On the page for your database, select **Query editor (preview)** in the left menu.
3. Enter your server admin login information, and select **OK**.



1. Enter the following query in the **Query editor** pane.

SQLCopy

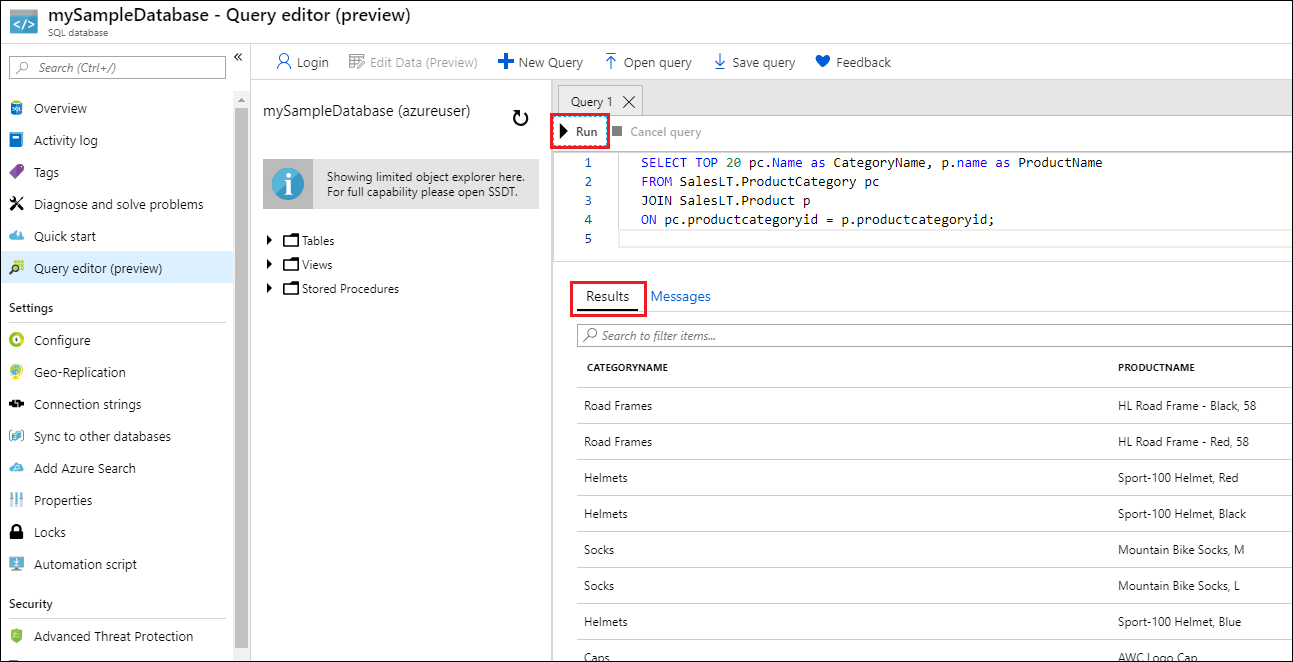
SELECT TOP 20 pc.Name as CategoryName, p.name as ProductName

FROM SalesLT.ProductCategory pc

JOIN SalesLT.Product p

ON pc.productcategoryid = p.productcategoryid;

1. Select **Run**, and then review the query results in the **Results** pane.



1. Close the **Query editor** page and select **OK** when prompted to discard your unsaved edits.

## Connect to your database using SSMS *(SQL Server Management Studio)*

1. Open SSMS.
2. The **Connect to Server** dialog box appears. Enter the following information:

| **Setting** | **Suggested value** | **Description** |
| --- | --- | --- |
| **Server type** | Database engine | Required value. |
| **Server name** | The fully qualified server name | Something like: **servername.database.windows.net**. |
| **Authentication** | SQL Server Authentication | This tutorial uses SQL Authentication. |
| **Login** | Server admin account user ID | The user ID from the server admin account used to create the server. |
| **Password** | Server admin account password | The password from the server admin account used to create the server. |
|  |  |  |