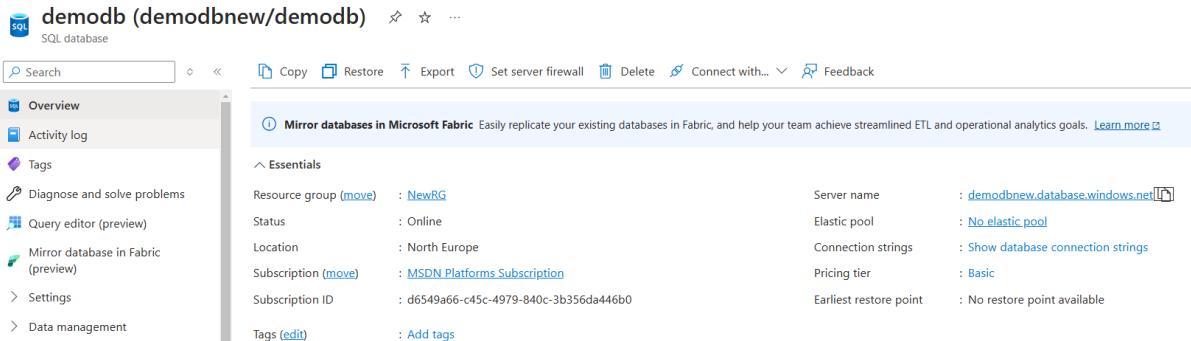


😊 Azure SQL – Using an Application to Generate Data

In this lab, you use an application to send data to an Azure SQL Database. First, download the necessary files, extract them, and open the SQL app in Visual Studio. Create an empty log data table in the database. Add your CSV file path on line 5 and the Azure SQL connection string (with your password) on line 9 of the app. After saving changes, run the app to load data into the table. Verify with a SELECT query. Finally, upgrade the database tier from Basic to Standard in the Azure Portal. The goal is to populate and scale your database efficiently!

1. In this lab we will use an application through which we will send our data to the Azure SQL Database.
2. Now you have to download 3 supporting files for this lab from GitHub. Then extract them all to access the data. After that, you have to open the SQL app in the Visual Studio.
3. Also, there are some prerequisites for this lab, you should have an Azure SQL Database running and it should be connected with the SQL Server Management Studio.



The screenshot shows the Azure portal interface for a database named 'demodb'. The left sidebar has options like Overview, Activity log, Tags, Diagnose and solve problems, Query editor (preview), Mirror database in Fabric (preview), Settings, and Data management. The main content area is titled 'Overview' and shows the following details:

Essentials	
Resource group (move)	: NewRG
Status	: Online
Location	: North Europe
Subscription (move)	: MSDN Platforms Subscription
Subscription ID	: d6549a66-c45c-4979-840c-3b356da446b0
Tags (edit)	: Add tags
Server name	: demodbnew.database.windows.net
Elastic pool	: No elastic pool
Connection strings	: Show database connection strings
Pricing tier	: Basic
Earliest restore point	: No restore point available

4. Our Database is connected and it does not have any tables in place. So, we have created a new table using the query as you can see below. Also, this is just an empty table.

```
CREATE TABLE [logdata]
(
    [Id] int,
    [Operationname] [varchar](200) NULL,
    [Status] [varchar](100) NULL,
    [Eventcategory] [varchar](100) NULL,
    [Resourcetype] [varchar](1000) NULL,
    [Resource] [varchar](2000) NULL
)
```

The screenshot shows two windows of Microsoft SQL Server Management Studio.

Object Explorer: Shows the database structure for 'demodb' under 'demodbnew.database.windows.net'. It includes 'Tables' (logdata), 'Views', 'External Resources', 'Synonyms', 'Programmability', 'Query Store', 'Extended Events', 'Storage', and 'Security'.

SQL Query Window: Contains the following SQL code:

```

CREATE TABLE [logdata]
(
    [Id] int,
    [Operationname] [varchar](200) NULL,
    [Status] [varchar](100) NULL,
    [Eventcategory] [varchar](100) NULL,
    [Resourcetype] [varchar](1000) NULL,
    [Resource] [varchar](2000) NULL
)

```

Messages pane: Commands completed successfully.
Completion time: 2025-03-03T10:46:44.3180008+05:30

Task List: Query executed successfully.

Results Window: Shows the following T-SQL command:

```
Select * From [logdata]
```

Results pane: A table structure with columns: Id, Operationname, Status, Eventcategory, Resourcetype, Resource.

- Now we will use the SQL app to load the data into our table present in the database using the Log.csv file that we have on our laptop we have to mention the location of the file in line 5 of the application as highlighted in the snapshot. If you see in the **application on line number 9** you have to enter the SQL Connection String.

```

1     using System.Data;
2     using System.Data.SqlClient;
3     using System.Reflection.Metadata;
4
5     string strFilePath = "C:\\Temp\\Log.csv";
6     StreamReader? logReader = null;
7     int Id = 1;
8     SqlConnection appdbConnection =
9     new SqlConnection("");
10
11    SqlParameter paramId = new SqlParameter();
12    paramId.ParameterName = "@Id";
13
14    SqlParameter paramOperationname = new SqlParameter();
15    paramOperationname.ParameterName = "@Operationname";
16

```

- To obtain this you have to go to your database in Azure Portal and click on show database connection strings. Then you have to copy the ADO.NET connection string and paste it in the application.

Server name	: demodbnew.database.windows.net
Elastic pool	: No elastic pool
Connection strings	: Show database connection strings
Pricing tier	: Basic
Earliest restore point	: No restore point available

ADO.NET JDBC ODBC PHP Go

ADO.NET (Microsoft Entra passwordless authentication)

Microsoft.Data.SqlClient Quickstart Entity Framework Core Quickstart

```
Server=tcp:demodbnew.database.windows.net,1433;Initial Catalog=demodb;Encrypt=True;TrustServerCertificate=False;Connection Timeout=30;Authentication="Active Directory Default";
```

ADO.NET (SQL authentication)

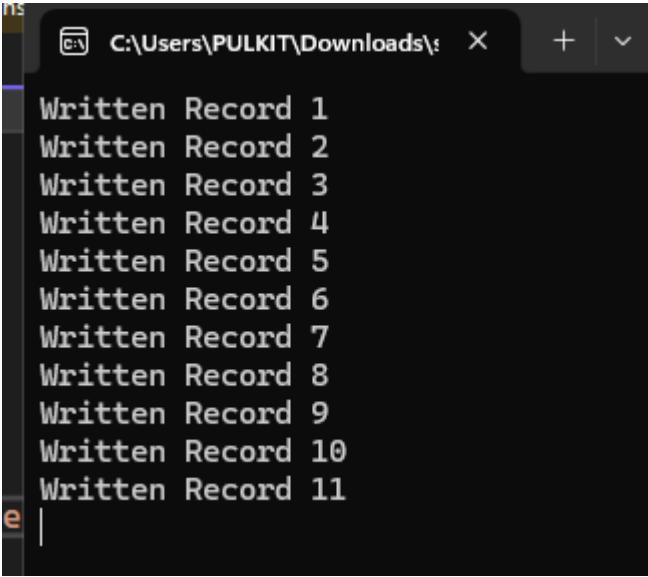
```
Server=tcp:demodbnew.database.windows.net,1433;Initial Catalog=demodb;Persist Security Info=False;User ID=sqladmin;Password=[your_password];MultipleActiveResultSets=False;Encrypt=True;TrustServerCertificate=False;Connection Timeout=30;
```

Copy Copied

- Also, in the connection string if you look carefully you will see that you have to enter your password. Please do that and save the changes.

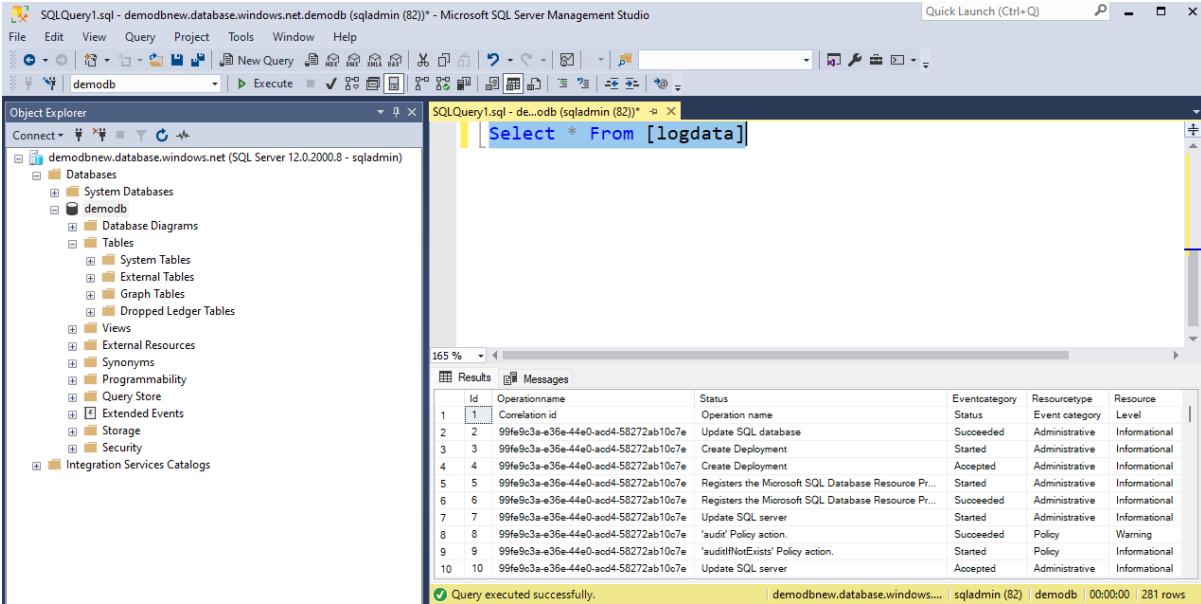
```
ID=sqladmin;Password={your_password};MultipleActiveResultSets=False;
```

- After all the steps are completed you have to run the SQL application and you will see that the records are being written onto the table.



```
Written Record 1
Written Record 2
Written Record 3
Written Record 4
Written Record 5
Written Record 6
Written Record 7
Written Record 8
Written Record 9
Written Record 10
Written Record 11
```

- Now you can see that if we run the Select query, we can see the data in place.



SQLQuery1.sql - demodbnew.database.windows.net.demodb (sqladmin (82)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

New Query | Execute | Results | Messages

demodb | demodb | SQLQuery1.sql - de...odb (sqladmin (82)) | Select * From [logdata]

Object Explorer

Connect | demodbnew.database.windows.net (SQL Server 12.0.2000.8 - sqladmin)

Databases | System Databases | demodb | Tables | Views | External Resources | Synonyms | Programmability | Query Store | Extended Events | Storage | Security | Integration Services Catalogs

Results | Messages

	Id	Operationname	Status	Eventcategory	ResourceType	Resource Level
1	1	Correlation id	Operation name	Status	Event category	Informational
2	2	99fe9c3a-e36e-44e0-acd4-58272ab10c7e	Update SQL database	Succeeded	Administrative	Informational
3	3	99fe9c3a-e36e-44e0-acd4-58272ab10c7e	Create Deployment	Started	Administrative	Informational
4	4	99fe9c3a-e36e-44e0-acd4-58272ab10c7e	Create Deployment	Accepted	Administrative	Informational
5	5	99fe9c3a-e36e-44e0-acd4-58272ab10c7e	Registers the Microsoft SQL Database Resource Pr...	Started	Administrative	Informational
6	6	99fe9c3a-e36e-44e0-acd4-58272ab10c7e	Registers the Microsoft SQL Database Resource Pr...	Succeeded	Administrative	Informational
7	7	99fe9c3a-e36e-44e0-acd4-58272ab10c7e	Update SQL server	Started	Administrative	Informational
8	8	99fe9c3a-e36e-44e0-acd4-58272ab10c7e	'audit' Policy action.	Succeeded	Policy	Warning
9	9	99fe9c3a-e36e-44e0-acd4-58272ab10c7e	'auditIfNotExists' Policy action.	Started	Policy	Informational
10	10	99fe9c3a-e36e-44e0-acd4-58272ab10c7e	Update SQL server	Accepted	Administrative	Informational

Query executed successfully.

- Now to move forward with the next labs we have to change the pricing for our database. For that navigate to your database in Portal. Currently, you see we have the basic pricing tier, and if we click on it.

Copy Restore Export Set server firewall Delete Connect with... Feedback

① Mirror databases in Microsoft Fabric Easily replicate your existing databases in Fabric, and help your team achieve streamlined ETL and operational analytics goals. [Learn more ↗](#)

^ Essentials

Resource group (move)	:	NewRG	Server name	:	demodbnew.database.windows.net
Status	:	Online	Elastic pool	:	No elastic pool
Location	:	North Europe	Connection strings	:	Show database connection strings
Subscription (move)	:	MSDN Platforms Subscription	Pricing tier	:	Basic
Subscription ID	:	d6549a66-c45c-4979-840c-3b356da446b0	Earliest restore point	:	2025-03-03 05:26 UTC

11. We have the option to change the service tier from Basic to Standard. Then click on Apply.

Service and compute tier

Select from the available tiers based on the needs of your workload. The vCore model provides a wide range of configuration controls and offers Hyperscale and Serverless to automatically scale your database based on your workload needs. Alternately, the DTU model provides set price/performance packages to choose from for easy configuration. [Learn more ↗](#)

SQL Database Hyperscale: Low price, high scalability, and best feature set. [Learn more ↗](#)

Service tier Standard (Budget friendly) ▾

Compare service tiers ↗

DTUs [Compare DTU options ↗](#)

10

Data max size (GB) 2

Backup storage redundancy ①

Locally-redundant backup storage

Zone-redundant backup storage

Geo-redundant backup storage

Geo-Zone-redundant backup storage [Preview]

12. Now you can see that the pricing tier has been changes to Standard.

Copy Restore Export Set server firewall Delete Connect with... Feedback

① Mirror databases in Microsoft Fabric Easily replicate your existing databases in Fabric, and help your team achieve streamlined ETL and operational analytics goals. [Learn more ↗](#)

Scaling operation completed

^ Essentials

Resource group (move)	:	NewRG	Server name	:	demodbnew.database.windows.net
Status	:	Online	Elastic pool	:	No elastic pool
Location	:	North Europe	Connection strings	:	Show database connection strings
Subscription (move)	:	MSDN Platforms Subscription	Pricing tier	:	Standard S0: 10 DTUs
Subscription ID	:	d6549a66-c45c-4979-840c-3b356da446b0	Earliest restore point	:	2025-03-03 05:26 UTC

13. While changing your pricing tier there comes a time when your database stops receiving data but when the tier changes then your database starts up again.