**Synapse Serverless SQL pool.**

Azure **Synapse Serverless SQL Pool** is a **pay-per-query** solution that enables **on-demand querying of data stored in Azure Data Lake (ADLS)** without the need for preloaded data.

**Key Features of Serverless SQL**

**No Data Storage** – Queries directly from **ADLS without needing a database**.  
**Schema-on-Read** – Data structure is **determined at query time**, allowing flexible queries.  
**Pay per Query** – **Only pay for the data scanned** instead of paying for infrastructure.  
**Built-in Security** – Uses **Managed Identity, SAS tokens, or Credentials** for access control.

**Connect to serverless sql**

Go to Azure portal and in synapse workspace 🡪 setting 🡪 properties 🡪 Serverless SQL endpoint and copy it. A screenshot of a computer

Description automatically generated and use this server name and admin-password same as normal sqldb

**Create External tables and views.**

We need to **securely access ADLS data** using **SAS tokens**.

A screenshot of a computer

Description automatically generated **Master Key:** Encrypts stored credentials securely.

Uses a **SAS token** for secure access to **ADLS Gen2**.  
  
To get the SAS Token go to ADLS gen2 in Azure portal, select shared access signature A screenshot of a computer screen

Description automatically generatedand Enable services, container, Objects and select generate SAS and connecting string A screenshot of a computer

Description automatically generatedand copy SAS token

### **Create External Data Source**

Defines **where the external data is stored**.

A close-up of a computer screen

Description automatically generated **Purpose:**

* Allows **Synapse SQL** to access **ADLS Gen2**.
* Uses the **SAS credential** (democredential).

To get the location , go to ADLS gen2 -🡪 Setting 🡪 Endpoint 🡪 from Datalake Storage copy the locationA screenshot of a computer

Description automatically generated **Define File Format**

**CSV**

Specifies how **CSV files** should be parsed.

**A computer screen shot of a computer program

Description automatically generated**

**Purpose:**

* Defines **CSV parsing rules** (delimiter, header row, text format).

In the same way define a file format for parquet and delta formats

**Parquet**

A close-up of a file format

Description automatically generated

**Delta** **A screenshot of a computer

Description automatically generated**

### **Create an External Table**

**CSV**

Creates a **permanent external table** to query data from ADLS. A screenshot of a computer code

Description automatically generated

**Purpose:**

* Queries **CSV files stored in ADLS (scdtype1/ folder)**.
* **Uses storageaccount1** to connect to ADLS.
* **Uses file\_format\_csv1** for **parsing CSV files**.

In the same way create tables for parquet and delta files using External data source and file format to connect with ADLS

A computer screen shot of a code

Description automatically generated A computer screen shot of a computer code

Description automatically generatedAll the tables are created successfully in synapse serverless SQL pool.

### A screenshot of a computer Description automatically generated A screenshot of a computer Description automatically generated A screenshot of a computer Description automatically generated **Create a View Using OPENROWSET**

Allows **direct querying** without creating an external table.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated **A screenshot of a computer

Description automatically generated Purpose:**

* Reads **CSV files directly from ADLS** (without storing metadata).
* **Dynamic query** for **ad-hoc analysis**.

**PARQUET**

**A screenshot of a computer

Description automatically generated** **A screenshot of a computer

Description automatically generated**WILDCARD PATH A screenshot of a computer

Description automatically generated All the files in that path we can retreive using \*.parquetA screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generatedWe cannot do wild path for delta format because delta files are stores in .parquet format

### 1. **Performance:**

* **Partitioning:** Split data into smaller chunks for faster queries.
* **Indexes:** Create indexes on frequently queried columns to speed up access.
* **Caching:** Cache frequent query results to improve performance.

### 2. **Error Handling:**

* **Retry Logic:** Implement retries for transient failures.
* **Data Validation:** Check for missing or invalid data before loading.

### 3. **Data Cleaning:**

* **Remove Duplicates:** Use DISTINCT or ROW\_NUMBER() to clean data.
* **Handle Missing Data:** Replace or remove missing values.
* **Data Types:** Ensure data types are consistent (e.g., INT vs. VARCHAR).

### 4. **Best Practices:**

* **Compression:** Use compressed formats like Parquet or Delta to save storage and improve speed.
* **ETL Pipelines:** Use Azure Data Factory or Databricks to automate extraction, transformation, and loading (ETL).