

Portfolio Project:

Loading data into a warehouse using T-SQL

Project Overview

This project focused on loading data into a data warehouse using T-SQL within Microsoft Fabric. The objective was to create a workspace, establish a lakehouse for data storage, and utilize SQL queries to perform analytical operations on the data. This exercise provided hands-on experience with data ingestion, table creation, and running analytical queries in a data warehouse environment.

Objectives

- 1. Create a Workspace**
 - Set up a workspace in Microsoft Fabric with the necessary capacity enabled.
- 2. Create a Lakehouse and Upload Files**
 - Establish a data lakehouse and upload required data files for analysis.
- 3. Create a Table in the Lakehouse**
 - Load data from the uploaded CSV file into a new table within the lakehouse.
- 4. Create a Warehouse**
 - Set up a data warehouse to facilitate large-scale analytics.
- 5. Create Fact Table, Dimensions, and View**
 - Define the structure of the data warehouse by creating fact tables, dimension tables, and a view for simplified data access.
- 6. Load Data to the Warehouse**
 - Implement a stored procedure to load data from the lakehouse into the warehouse.
- 7. Run Analytical Queries**
 - Execute SQL queries to analyze the data and derive insights.

Experience

Create a Workspace

- Navigated to the Microsoft Fabric home page and signed in with credentials.
- Selected Workspaces from the menu bar and created a new workspace.

Create a Lakehouse and Upload Files

- Selected + New item and created a new Lakehouse.
- Downloaded the CSV file and uploaded it to the lakehouse.
- Verified that the CSV file has been successfully uploaded.

Upload files



Files/

☐ Overwrite if files already exist

Upload

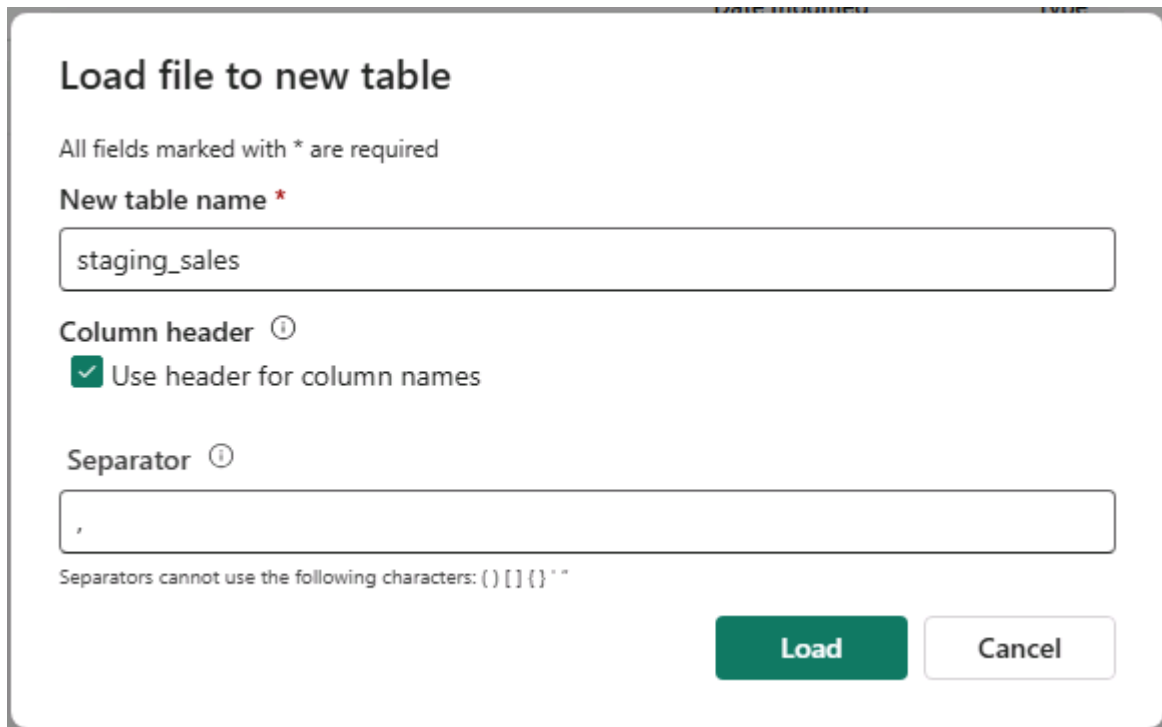
Current uploads

Dismiss: [Completed](#) [All](#)

File Name	Lakehouse Name		
sales.csv	loaddata_ls	3 MB / 3 MB	

Create a Table in the Lakehouse

- In the menu for the uploaded CSV file, selected Load to tables, then New table.
- Provided the new table name (e.g., staging_sales), selected the option to use headers for column names, and set the separator to a comma. Selected Load.



Load file to new table

All fields marked with * are required

New table name *

staging_sales

Column header ⓘ

☒ Use header for column names

Separator ⓘ

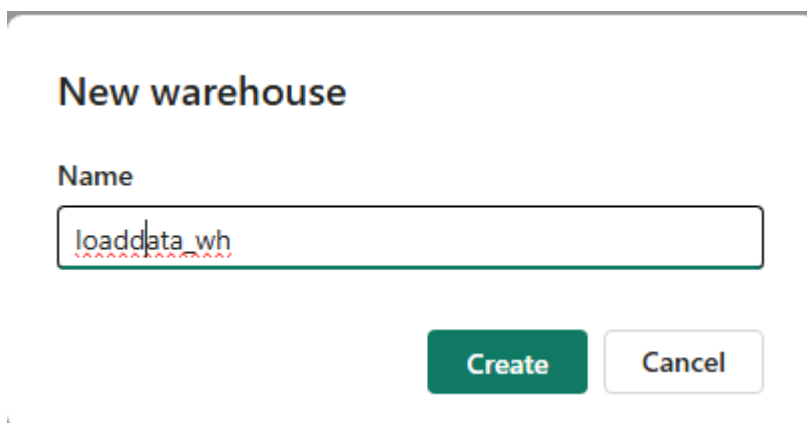
,

Separators cannot use the following characters: () [] {} ' ' "

Load **Cancel**

Create a Warehouse

- On the menu bar, selected Create, then Warehouse under the Data Warehouse section. Gave it a unique name.



New warehouse

Name

loaddata_wh

Create **Cancel**

Create Fact Table, Dimensions, and View

- Selected the warehouse and opened a new SQL query.
- Executed SQL commands to create the necessary schemas, fact tables, and dimension tables for the Sales data.
- Created a view that points to the lakehouse staging table.

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'Sales' database is expanded, showing its structure: Tables (Dim_Customer, Dim_Item, Fact_Sales), Views, Functions, and Stored Procedures. In the center, a SQL script is displayed in a query window, showing the creation of a view named 'Sales.Staging_Sales' which selects all data from the 'staging_sales' table in the 'loaddata_ls' database. On the right, the 'Sales' database is again expanded, showing the 'Staging_Sales' view under the 'Views' folder.

```
1 CREATE VIEW Sales.Staging_Sales
2 AS
3 SELECT * FROM [loaddata_ls].[dbo].[staging_sales];
```

Load Data to the Warehouse

- Created a stored procedure to load data from the lakehouse into the warehouse.
- Executed the stored procedure to load data for a specific year (e.g., 2021).

```
1 EXEC Sales.LoadDataFromStaging 2021
```

Run Analytical Queries

- Executed various SQL queries to analyze the data in the warehouse, such as total sales by customer, top-selling items, and categorized sales.

```

SELECT c.CustomerName, SUM(s.UnitPrice * s.Quantity) AS TotalSales
FROM Sales.Fact_Sales s
JOIN Sales.Dim_Customer c
ON s.CustomerID = c.CustomerID
WHERE YEAR(s.OrderDate) = 2021
GROUP BY c.CustomerName
ORDER BY TotalSales DESC;

```

	ABC CustomerName	12F TotalSales
1	Jordan Turner	14686.6992
2	Nicole Blue	11494.935
3	Maurice Shan	10525.5982
4	Janet Munoz	10070.1096
5	Alexandra Hall	9710.76
6	Jordan King	9273.92
7	Jaclyn Xu	9197.025
8	Jordan Roberts	8388.62
9	Larry Vazquez	7321.35
10	Kaitlyn Henderson	7317.95

```

1  SELECT i.ItemName, SUM(s.UnitPrice * s.Quantity) AS TotalSales
2  FROM Sales.Fact_Sales s
3  JOIN Sales.Dim_Item i
4  ON s.ItemID = i.ItemID
5  WHERE YEAR(s.OrderDate) = 2021
6  GROUP BY i.ItemName
7  ORDER BY TotalSales DESC;

```

	ABC ItemName	12F TotalSales
1	Mountain-200 Black, 46	718987.579999998
2	Mountain-200 Silver, 46	687794.173199998
3	Mountain-200 Black, 38	668006.016399998
4	Mountain-200 Black, 42	647760.926199998
5	Mountain-200 Silver, 38	641145.803999998
6	Mountain-200 Silver, 42	631617.273599998
7	Road-250 Black, 52	376275.899999999
8	Road-250 Black, 48	340061.9625
9	Road-250 Black, 44	338753.025
10	Road-250 Black, 58	333430.0125
11	Road-250 Black, 50	316501.0075

Results

- ✓ A workspace and lakehouse were successfully created, allowing for data ingestion.
- ✓ The sales data was uploaded and processed into a new table within the lakehouse.
- ✓ A data warehouse was established, including fact and dimension tables, along with a view for data access.
- ✓ Data was successfully loaded into the warehouse using a stored procedure.
- ✓ Various analytical queries were executed, providing insights into customer sales and item performance.

Conclusion

This project provided a comprehensive introduction to loading data into a data warehouse using T-SQL in Microsoft Fabric. Participants gained valuable experience in workspace setup, data ingestion, and executing analytical queries. The exercise highlighted the capabilities of Microsoft Fabric for managing and analyzing large datasets, enabling effective data manipulation and reporting.

Resources

Source file:

<https://raw.githubusercontent.com/MicrosoftLearning/dp-data/main/orders.csv>

GitHub profile: <https://github.com/ThatoMTNG/Microsoft-Fabric-Analytics-Engineer-DP-600->

Mentions

Project Author: Thato Metsing (<https://www.linkedin.com/in/thatometsing/>)

Project Mentor: Maureen Direro (<https://www.linkedin.com/in/maureen-direro-46a6b220/>)