

# Lab 3 Transform Data with SQL

## BUAN 6390.001 – Analytics Practicum

Name: Swastik Bhatnagar  
NetID: sxb220210

Connecting to PowerShell:

labclient.labondemand.com/LabClient/c8f4256a-b0be-4456-a52e-d98e787c346d

Home - Microsoft Azure

Search resources, services, and docs (5+)

Copilot

User1-48405095@LODS...  
LODS-PROD-MCA-BUONPROD

Azure services

Create a resource

Quickstart Center

Azure AI services

Kubernetes services

Virtual machines

App Services

Storage accounts

SQL databases

Azure Cosmos DB

More services

Resources

Recent Favorite

Name Type Last Viewed

No resources have been viewed recently

View all resources

Navigate

Switch to Bash Restart Manage files New session Editor Web preview Settings Help

Registering resource providers...  
Microsoft.Synapse : Registered  
Microsoft.Sql : Registered  
Microsoft.Storage : Registered  
Microsoft.Compute : Registered  
Your randomly-generated suffix for Azure resources is b7r9cud  
Finding an available region. This may take several minutes...  
Trying westus2  
Using westus2  
Creating dp203-b7r9cud resource group in westus2 ...  
Creating synapseb7r9cud Synapse Analytics workspace in dp203-b7r9cud resource group...  
(This may take some time!)

Transform Data with SQL  
1 Hr 54 Min Remaining

Instructions Resources Help

Note: If you have previously created a cloud shell that uses a Bash environment, use the drop-down menu at the top left of the cloud shell pane to change it to **PowerShell**.

3. Note that you can resize the cloud shell by dragging the separator bar at the top of the pane, or by using the  $\text{--}$ ,  $\text{+}$ , and  $\text{X}$  icons at the top right of the pane to minimize, maximize, and close the pane. For more information about using the Azure Cloud Shell, see the [Azure Cloud Shell documentation](#).

4. In the PowerShell pane, enter the following commands to clone this repo:

```
rm -r dp-203 -f
git clone https://github.com/Microso
```

5. After the repo has been cloned, enter the following commands to change to the folder for this exercise and run the **setup.ps1** script it contains:

```
cd dp-203/Allfiles/Labs/03
./setup.ps1
```

6. If prompted, choose which subscription you want to use (this will only happen if you have access to multiple Azure subscriptions).

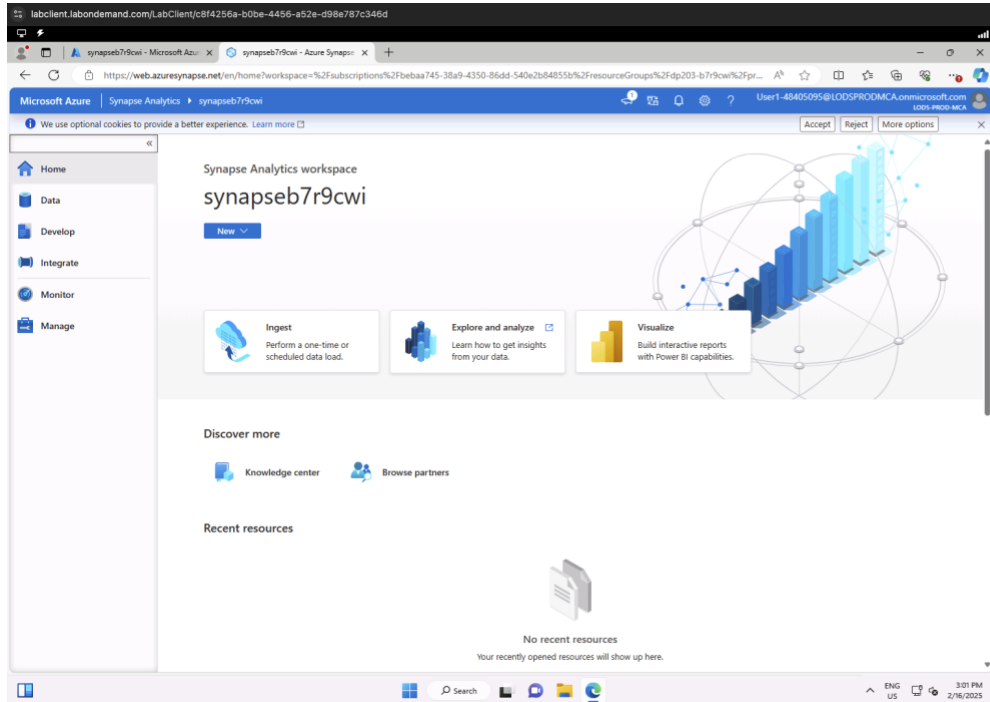
7. When prompted, enter a suitable password to be set for your Azure Synapse SQL pool.

Note: Be sure to remember this password!

8. Wait for the script to complete - this typically takes around 10 minutes, but in some cases may take longer. While you are waiting, review the [CETAS](#) with

End >

## Opening Synapse Studio:



The screenshot shows the Microsoft Azure Synapse Analytics workspace 'synapseb7r9cwi'. The left sidebar contains navigation links: Home, Data, Develop, Integrate, Monitor, and Manage. The main content area is titled 'Synapse Analytics workspace synapseb7r9cwi' and features a 'New' button. Below this, there are three cards: 'Ingest' (Perform a one-time or scheduled data load), 'Explore and analyze' (Learn how to get insights from your data), and 'Visualize' (Build interactive reports with Power BI capabilities). The 'Discover more' section includes 'Knowledge center' and 'Browse partners'. The 'Recent resources' section is empty, with a message: 'No recent resources. Your recently opened resources will show up here.'

Transform Data with SQL  
1 Hr 48 Min Remaining

Instructions Resources Help

Synapse SQL article in the Azure Synapse Analytics documentation.

### Query data in files

The script provisions an Azure Synapse Analytics workspace and an Azure Storage account to host the data lake, then uploads some data files to the data lake.

### View files in the data lake

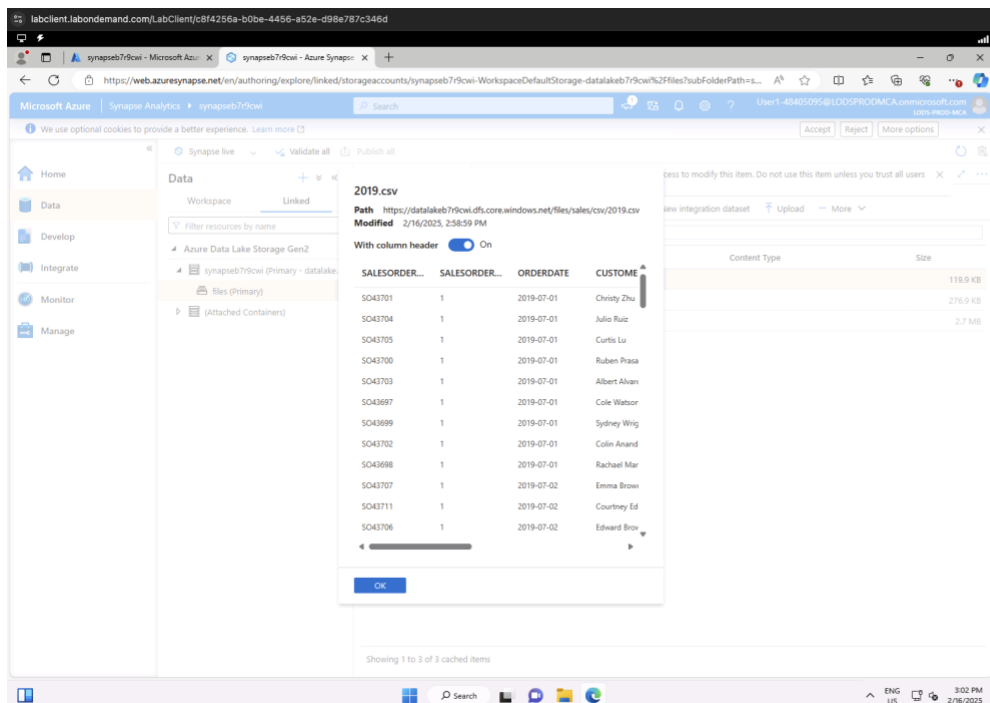
1. After the script has completed, in the Azure portal, go to the **dp203-xxxxxxx** resource group that it created, and select your Synapse workspace.
2. In the **Overview** page for your Synapse workspace, in the **Open Synapse Studio** card, select **Open** to open Synapse Studio in a new browser tab, signing in if prompted.
3. On the left side of Synapse Studio, use the **>>** icon to expand the menu - this reveals the different pages within Synapse Studio that you'll use to manage resources and perform data analytics tasks.
4. On the **Data** page, view the **Linked** tab and verify that your workspace includes a link to your Azure Data Lake Storage Gen2 storage account, which should have a name similar to **synapse\*xxxxxxx\* (Primary - datalake\*xxxxxxx\*)**.
5. Expand your storage account and verify that it contains a file system container named **files**.
6. Select the **files** container, and note that it contains a folder named **sales**. This folder contains the data files you are going to query.
7. Open the **sales** folder and the **csv** folder it contains, and observe that this folder contains **.csv** files for three years of sales data.
8. Right-click any of the files and select **Preview** to see the data it contains. Note that the files contain a header row.
9. Close the preview, and then use the **↑** button to navigate back to the **sales** folder.

### Use SQL to query CSV files

1. Select the **csv** folder, and then in the **New SQL script** list on the toolbar, select **Select TOP 100 rows**.

ENG US 3:01 PM 2/16/2023

## Viewing the Data files:



The screenshot shows the Microsoft Azure Synapse Analytics workspace 'synapseb7r9cwi' with the 'Data' page selected. The left sidebar contains navigation links: Home, Data, Develop, Integrate, Monitor, and Manage. The main content area is titled 'Data' and shows a list of files under the 'Linked' tab. The file '2019.csv' is selected, and its details are shown: Path: https://datalakeb7r9cwi.dfs.core.windows.net/files/sales/csv/2019.csv, Modified: 2/16/2023, 2:58:59 PM. The file is previewed in a table view with columns: SALESORDER..., SALESORDER..., ORDERDATE, and CUSTOMER. The table contains 10 rows of data.

Transform Data with SQL  
1 Hr 47 Min Remaining

Instructions Resources Help

Synapse SQL article in the Azure Synapse Analytics documentation.

### Query data in files

The script provisions an Azure Synapse Analytics workspace and an Azure Storage account to host the data lake, then uploads some data files to the data lake.

### View files in the data lake

1. After the script has completed, in the Azure portal, go to the **dp203-xxxxxxx** resource group that it created, and select your Synapse workspace.
2. In the **Overview** page for your Synapse workspace, in the **Open Synapse Studio** card, select **Open** to open Synapse Studio in a new browser tab, signing in if prompted.
3. On the left side of Synapse Studio, use the **>>** icon to expand the menu - this reveals the different pages within Synapse Studio that you'll use to manage resources and perform data analytics tasks.
4. On the **Data** page, view the **Linked** tab and verify that your workspace includes a link to your Azure Data Lake Storage Gen2 storage account, which should have a name similar to **synapse\*xxxxxxx\* (Primary - datalake\*xxxxxxx\*)**.
5. Expand your storage account and verify that it contains a file system container named **files**.
6. Select the **files** container, and note that it contains a folder named **sales**. This folder contains the data files you are going to query.
7. Open the **sales** folder and the **csv** folder it contains, and observe that this folder contains **.csv** files for three years of sales data.
8. Right-click any of the files and select **Preview** to see the data it contains. Note that the files contain a header row.
9. Close the preview, and then use the **↑** button to navigate back to the **sales** folder.

### Use SQL to query CSV files

1. Select the **csv** folder, and then in the **New SQL script** list on the toolbar, select **Select TOP 100 rows**.

ENG US 3:02 PM 2/16/2023

## Using SQL to query files:

labclient.labondemand.com/LabClient/c8f4256a-b0be-4456-a52e-d98e787c346d

https://web.azure.synapse.net/en/authoring/explore/linked/sqlscripts/SQL%20script%2017workspace=%2Fsubscriptions%2Fbebaa745-38a9-4350-86dd-540e2b648... User1-48405095@LOOSPRODCA.onmicrosoft.com

Microsoft Azure | Synapse Analytics | synapseb79cwi

Home Data Develop Integrate Monitor Manage

Workspace Linked

Filter resources by name

Azure Data Lake Storage Gen2 2

synapseb79cwi (Primary - data lake...)

Files (Primary)

(Attached Containers)

Files

Query Sales CSV

Other users in your workspace may have access to modify this item. Do not use this item unless you trust all users who may have access to the workspace.

Connect to Built-in Use database master

Run Undo Publish Query plan

1 -- This is auto-generated code

2 SELECT

3 TOP 100 \*

4 FROM

5 OPENROWSET(

6 BULK 'https://datalakeb79cwi.dfs.core.windows.net/files/sales/csv/\*',

7 FORMAT = 'CSV',

8 PARSER\_VERSION = '2.0',

9 HEADER\_ROW = TRUE

10 ) AS [result]

11

Results Messages

View Table Chart Export results

Search


SalesOrderNo...	SalesOrderLine...	OrderDate	CustomerName	EmailAddress	Item	Quantity	UnitPrice
SO49171	1	2021-01-01	Mariah Foster	mariah21@adv...	Road-250 Black...	1	2181.5625
SO49172	1	2021-01-01	Brian Howard	brian23@advent...	Road-250 Red...	1	2443.35
SO49173	1	2021-01-01	Linda Alvarez	linda19@advent...	Mountain-200 ...	1	2071.4196
SO49174	1	2021-01-01	Gina Hernandez	gina4@advent...	Mountain-200 ...	1	2071.4196

00:00:01 Query executed successfully.

Transform Data with SQL

1 Hr 42 Min Remaining

Instructions Resources Help

files, and change the result settings to show **All rows**. Then in the toolbar, select **Publish** to save the script and use the **Properties** button (which looks similar to ) on the right end of the toolbar to hide the **Properties** pane.

4. Review the SQL code that has been generated, which should be similar to this:

```
SQL
-- This is auto-generated code
SELECT
TOP 100 *
FROM
OPENROWSET(
BULK 'https://datalakexxxxx...
FORMAT = 'CSV',
PARSER_VERSION='2.0'
) AS [result]
```

This code uses the OPENROWSET to read data from the CSV files in the sales folder and retrieves the first 100 rows of data.

5. In this case, the data files include the column names in the first row, so modify the query to add a **HEADER\_ROW = TRUE** parameter to the **OPENROWSET** clause, as shown here (don't forget to add a comma after the previous parameter):

```
SQL
SELECT
TOP 100 *
FROM
OPENROWSET(
BULK 'https://datalakexxxxx...
FORMAT = 'CSV',
PARSER_VERSION='2.0',
HEADER_ROW = TRUE
) AS [result]
```

6. In the **Connect to** list, ensure **Built-in** is selected - this represents the built-in SQL Pool that has been created with your workspace. Then on the toolbar, use the **Run** button.

End >

## Creating external data source and file format:

labclient.labondemand.com/LabClient/c8f4256a-b0be-4456-a52e-d98e787c346d

https://web.azure.synapse.net/en/authoring/explore/workspace/workspace=%2Fsubscriptions%2Fbebaa745-38a9-4350-86dd-540e2b648559/%2Fres... User1-48405095@LOOSPRODCA.onmicrosoft.com

Microsoft Azure | Synapse Analytics | synapseb79cwi

Home Data Develop Integrate Monitor Manage

Workspace Linked

Filter resources by name

SQL database 1

Sales (SQL)

External tables

External resources

External data sources

sales\_data

External file formats

Views

Schemas

Security

Select an item

Use the resource explorer to select or create a new item

Transform Data with SQL

1 Hr 32 Min Remaining

Instructions Resources Help

2. In the new script pane, add the following code (replacing **datalakexxxxx** with the name of your data lake storage account) to create a new database and add an external data source to it.

```
sql
-- Database for sales data
CREATE DATABASE Sales
COLLATE Latin1_General_100_BIN2_UTF8;
GO;

Use Sales;
GO;

-- External data is in the Files container in the workspace
CREATE EXTERNAL DATA SOURCE sales_data WITH (
LOCATION = 'https://datalakexxxxx.dfs.core.windows.net/files/sales'
);
GO;

-- Format for table files
CREATE EXTERNAL FILE FORMAT ParquetFormat
WITH (
FORMAT_TYPE = PARQUET,
DATA_COMPRESSION = 'org.apache.hadoop.io.compress.DefaultCodec'
);
GO;
```

3. Modify the script properties to change its name to **Create Sales DB**, and publish it.

4. Ensure that the script is connected to the **Built-in** SQL pool and the **master** database, and then run it.

5. Switch back to the **Data** page and use the **Refresh** button at the top right of Synapse Studio to refresh the page. Then view the **Workspace** tab in the **Data** pane, where a **SQL database** list is now displayed. Expand this list to verify that the **Sales** database has been created.

6. Expand the **Sales** database, its **External resources** folder, and the **External data sources** folder under that to see the **sales\_data** external data source you created.

Create an External table

End >

## Creating an External Table:

The screenshot displays the Microsoft Azure Synapse Analytics web portal. The left sidebar shows navigation options: Home, Data, Develop, Integrate, Monitor, and Manage. The 'Develop' tab is active, showing a list of SQL scripts. A new script named 'Create ProductSales...' is being created. The SQL editor contains the following code:

```
1 CREATE EXTERNAL TABLE ProductSalesTotals
2 WITH (
3     LOCATION = 'sales/productsales/',
4     DATA_SOURCE = sales_data,
5     FILE_FORMAT = ParquetFormat
6 )
7 AS
8 SELECT Item AS Product,
9        SUM(Quantity) AS ItemsSold,
10       ROUND(SUM(UnitPrice) - SUM(TaxAmount), 2) AS NetRevenue
11 FROM
12 OPENROWSET(
13     BULK 'sales/csv/*.csv',
14     DATA_SOURCE = 'sales_data',
15     FORMAT = 'CSV',
16     PARSE_VERSION = '2.0',
17     HEADER_ROW = TRUE
18 ) AS orders
19 GROUP BY Item;
```

The 'Properties' pane on the right shows the table name 'Create ProductSalesTotals table' and the type 'sql script'. A message at the bottom states 'No results to show' and 'Your query yielded no displayable results'. On the right side of the image, a 'Transform Data with SQL' panel shows instructions and a sample table:

1. Run the script. The results should look similar to this:

Product	ItemsSold	NetRevenue
AWC Logo Cap	1063	8791.86
...	...	...

4. Modify the SQL code to save the results of query in an external table, like this:

```
sql
1 CREATE EXTERNAL TABLE ProductSalesTotals
2 WITH (
3     LOCATION = 'sales/productsales/',
4     DATA_SOURCE = sales_data,
5     FILE_FORMAT = ParquetFormat
6 )
7 AS
8 SELECT Item AS Product,
9        SUM(Quantity) AS ItemsSold,
10       ROUND(SUM(UnitPrice) - SUM(TaxAmount), 2)
11 FROM
12 OPENROWSET(
13     BULK 'sales/csv/*.csv',
14     DATA_SOURCE = 'sales_data',
15     FORMAT = 'CSV',
16     PARSE_VERSION = '2.0',
17     HEADER_ROW = TRUE
18 ) AS orders
19 GROUP BY Item;
```

5. Run the script. This time there's no output, but the code should have created an external table based on the results of the query.

6. Name the script **Create ProductSalesTotals table** and publish it.

7. On the data page in the Microsoft portal, view the contents of

## Encapsulating data transformation in a stored procedure:

The screenshot displays the Microsoft Azure Synapse Analytics web portal. The left sidebar shows navigation options: Home, Data, Develop, Integrate, Monitor, and Manage. The 'Develop' tab is active, showing a list of SQL scripts. A new script named 'SQL script 1' is being created. The SQL editor contains the following code:

```
6 -- drop existing table
7 IF EXISTS (
8     SELECT * FROM sys.external_tables
9     WHERE name = 'YearlySalesTotals'
10 )
11 DROP EXTERNAL TABLE YearlySalesTotals
12 -- create external table
13 CREATE EXTERNAL TABLE YearlySalesTotals
14 WITH (
15     LOCATION = 'sales/yearlysales/',
16     DATA_SOURCE = sales_data,
17     FILE_FORMAT = ParquetFormat
18 )
19 AS
20 SELECT YEAR(OrderDate) AS CalendarYear,
21        SUM(Quantity) AS ItemsSold,
22        ROUND(SUM(UnitPrice) - SUM(TaxAmount), 2) AS NetRevenue
23 FROM
24 OPENROWSET(
25     BULK 'sales/csv/*.csv',
26     DATA_SOURCE = 'sales_data',
27     FORMAT = 'CSV',
28     PARSE_VERSION = '2.0',
29     HEADER_ROW = TRUE
30 ) AS orders
31 GROUP BY YEAR(OrderDate)
32 END
33 EXEC sp_GetYearlySales;
```

The 'Properties' pane on the right shows the table name 'SQL script 1' and the type 'sql script'. A message at the bottom states 'No results to show' and 'Your query yielded no displayable results'. On the right side of the image, a 'Transform Data with SQL' panel shows instructions and a sample table:

3. Run the script to create the stored procedure.

4. Under the code you just ran, add the following code to call the stored procedure:

```
sql
1 EXEC sp_GetYearlySales;
```

5. Select only the **EXEC sp\_GetYearlySales;** statement you just added, and use the **Run** button to run it.

6. On the **Files** tab containing the file system for your data lake, view the contents of the **sales** folder (refreshing the view if necessary) and verify that a new **yearlysales** folder has been created.

7. In the **yearlysales** folder, observe that a parquet file containing the aggregated yearly sales data has been created.

8. Switch back to the SQL script and re-run the **EXEC sp\_GetYearlySales;** statement, and observe that an error occurs.

Even though the script drops the external table, the folder containing the data is not deleted. To re-run the stored procedure (for example, as part of a scheduled data transformation

## Deleting the Azure resources:

The screenshot displays the Microsoft Azure portal interface. The main content area shows the 'Overview' page for the resource group 'dp203-b7r9cwi'. A notification banner at the top indicates 'Deleting resource group dp203-b7r9cwi' with a 'Running' status. Below the notification, a table lists the resources within the group:

Name	Type	Location
datakeb7r9cwi	Storage account	West US 2
synapseb7r9cwi	Synapse workspace	West US 2

At the bottom of the portal, a terminal window shows the output of a PowerShell script, including properties like 'ContinuationToken', 'VersionId', 'IsLatestVersion', 'AccessTier', 'TagCount', 'Tags', 'ListBlobProperties', 'Context', and 'Name'.

On the right side, a 'Transform Data with SQL' sidebar is visible, containing instructions for deleting Azure resources. The instructions include:

- 7. In the **yearlysales** folder, observe that a parquet file containing the aggregated yearly sales data has been created.
- 8. Switch back to the SQL script and re-run the `EXEC sp_GetYearlySales;` statement, and observe that an error occurs.
- 9. Switch back to the **files** tab, and view the **sales** folder. Then select the **yearlysales** folder and delete it.
- 10. Switch back to the SQL script and re-run the `EXEC sp_GetYearlySales;` statement. This time, the operation succeeds and a new data file is generated.

Below the instructions, a section titled 'Delete Azure resources' provides further guidance:

If you've finished exploring Azure Synapse Analytics, you should delete the resources you've created to avoid unnecessary Azure costs.

1. Close the Synapse Studio browser tab and return to the Azure portal.
2. On the Azure portal, on the **Home** page, select **Resource groups**.
3. Select the **dp203-xxxxxxx** resource group for your Synapse Analytics workspace (not the managed resource group), and verify that it contains the Synapse workspace and storage account for your workspace.
4. At the top of the **Overview** page for your resource group, select **Delete resource group**.
5. Enter the **dp203-xxxxxxx** resource group name to confirm you want to delete it, and select **Delete**.

After a few minutes, your Azure Synapse workspace resource group and the managed workspace resource group associated with it will be deleted.

The sidebar concludes with the text: 'End the lab' and 'Please be sure to end the lab.'

Conclusion: In this lab, I learned how to transform data using SQL in Azure Synapse. I connected to PowerShell, explored data files, and used SQL to query them efficiently. I also created an external data source, defined a file format, and built an external table to structure the data. Additionally, I encapsulated transformations within a stored procedure for better reusability. Finally, I practiced good resource management by deleting Azure resources. This lab strengthened my understanding of data transformation in a cloud environment.