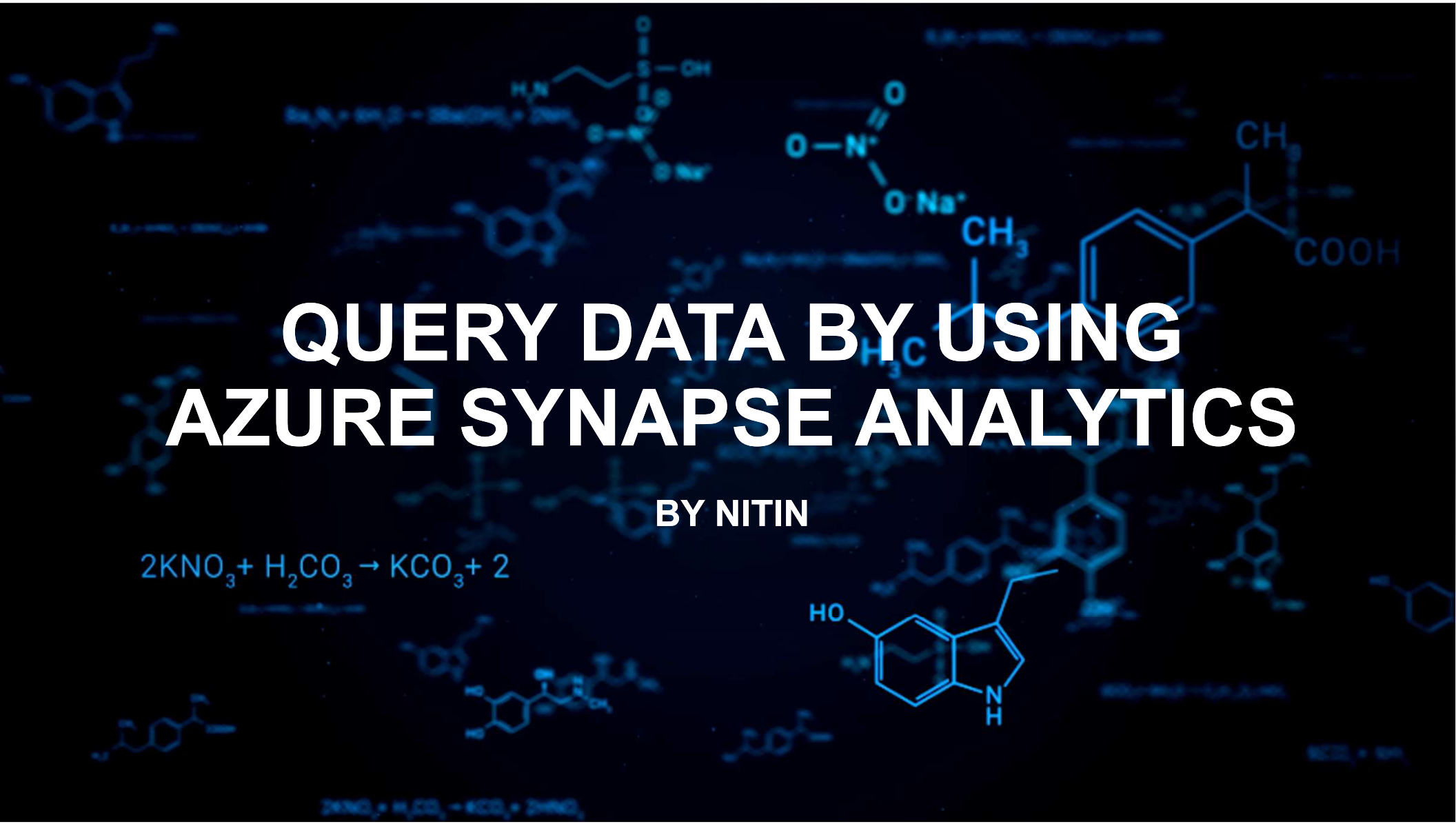
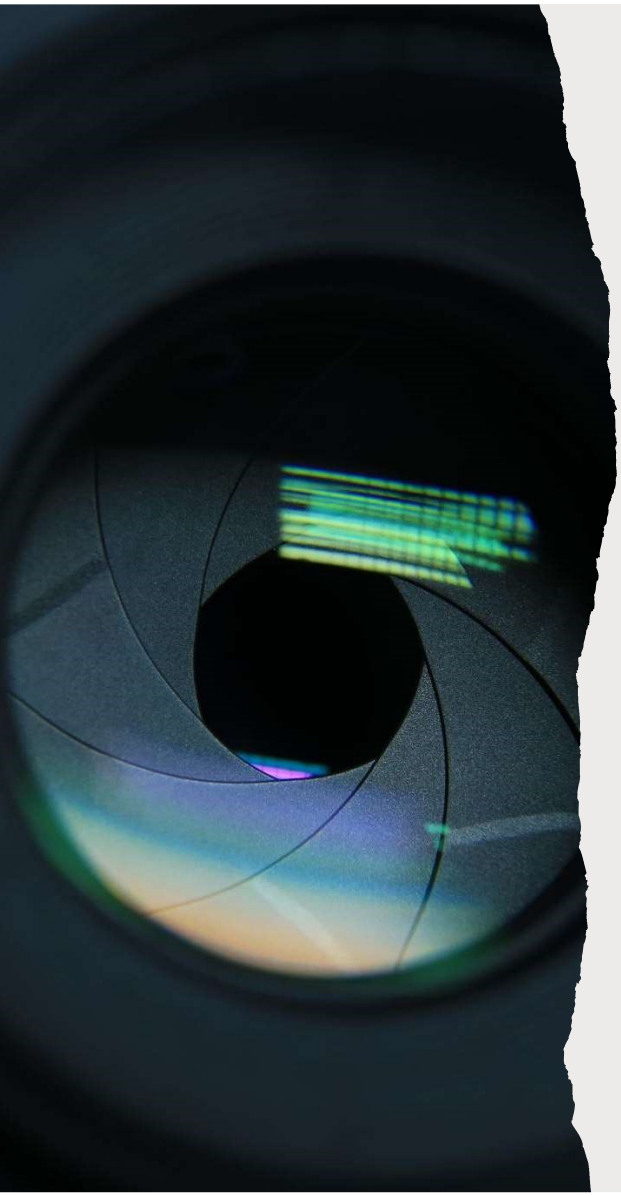

QUERY DATA BY USING
AZURE SYNAPSE ANALYTICS
BY NITIN
2KNO3+ H2CO3 → KCO3+ 2



OBJECTIVE

- I learned how to define external data sources using PolyBase to reference data lake storage for more advanced and reusable query scenarios.

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labI...

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstance...

Instance ID:6779219

- [x] 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.
- [x] 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*)**.
- [x] 5. Expand your storage account and verify that it contains a file system container named **files**.
- [x] 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 do not contain a header row, so you can unselect the option to display column headers.
- [ ] 9. Close the preview, and then use the ↑ button to

On VM: LON-CL1 Page: 4/

Support

Microsoft Azure Portal | Micros... | synapse0gknypd - Microsoft Az... | synapse0gknypd - Azure Synap...

web.azuresynapse.net/en/authoring/explore/linked/storageaccounts/synapse0gknypd-WorkspaceDefaultSto...

Microsoft Azure | Synapse Analytic | Search | XLab-Rfm-108@xtremelabs.us LAB DIRECTORY

Synapse live | Validate all | Publish all

Data

Workspace | Linked

Filter resources by name

Azure Data Lake Storage Gen2 2

synapse0gknypd (Primary - datalak...

files (Primary)

(Attached Containers)

files

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.

New SQL script | New data flow | New integration dataset | Upload | More

files

Name	Last Modified	Content Type	Size
sales	12/5/2024, 5:37:35 PM	Folder	

Showing 1 to 1 of 1 cached items

Search

ENG IN 5:41 PM 12/5/2024

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanc...

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanceG...

Instance ID:6779219

37% Completed

Lab Time Left: 01:30:11

Lab Actions

. Module: Model, query, and explore data in Azure Synapse

Use SQL to query CSV files:

- [x] 1. Select the **csv** folder, and then in the **New SQL script** list on the toolbar, select **Select TOP 100 ro**
- [x] 2. In the **File type** list, select **Text format**, and then apply the settings to open a new SQL script that
- [x] 3. In the **Properties** pane for **SQL Script 1** that is created, change the name to **Sales CSV query**, and **rows**. Then in the toolbar, select **Publish** to save the script and use the **Properties** button (which toolbar to hide the **Properties** pane.
- [x] 4. Review the SQL code that has been generated, which should be similar to this:

Paste Content

```
-- This is auto-generated code
SELECT
    TOP 100 *
FROM
    OPENROWSET(
        BULK 'https://datalakexxxxxxx.dfs.core.windows.net/files/sales/csv**',
        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 the **Connect to** list, ensure **Built-in** is selected - this represents the built-in SQL Pool that was c
- [ ] 6. On the toolbar, use the ▷ **Run** button to run the SQL code, and review the results, which should

C1	C2	C3	C4	C5	C6	C7	C8
SO43701	1	2019-07-01	Chirsty Zhu	clarence35@adventure-works.com	Mountain-100	1	3399.99

On VM: LON-CL1

Page: 5/

Support

Microsoft Azure Port | synapse0gknypd - M | synapse0gknypd - A | New Tab

web.azuresynapse.net/en/authoring/explore/linked/sqlscripts/SQL%20script%201?workspace=%2Fsub...

synapse0gknypd | Search | XLab-Rfm-108@xtremelabs.us LAB DIRECTORY

Synapse live | Validate all | Publish all

files | Sales CSV query

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.

Run | Undo | Publish | Query plan | Connect to: Built-in

```
1  -- This is auto-generated code
2  SELECT
3      TOP 100 *
4  FROM
5      OPENROWSET(
6          BULK 'https://datalake0gknypd.dfs.core.windows.net/files/sales
7          FORMAT = 'CSV',
8          PARSER_VERSION = '2.0'
9      ) AS [result]
10
```

Results | Messages

View: Table | Chart | Export results

Search

C1	C2	C3	C4	C5
SO49171	1	2021-01-01	Mariah Foster	mariah2
SO49172	1	2021-01-01	Brian Howard	brian23@

00:00:12 Query executed successfully.

Properties

General | Related (0)

Name *: Sales CSV query

Description:

Type: .sql script

Size: 228 bytes

Results settings per query:
- (•) First 5000 rows (default)
- ( ) All rows

Search | ENG IN | 5:46 PM 12/5/2024

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanc...

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanceG...

Instance ID:6779219

7. Note the results consist of columns named C1, C2, and so on. In this example, the CSV files do not include the column headers. While it's pc generic column names that have been assigned, or by ordinal position, it will be easier to understand the data if you define a tabular schem the OPENROWSET function as shown here (replacing *datalakexxxxxxx* with the name of your data lake storage account), and then rerun the

Paste Content

```
SELECT
    TOP 100 *
FROM
    OPENROWSET(
        BULK 'https://datalakexxxxxxx.dfs.core.windows.net/files/sales/csv/**',
        FORMAT = 'CSV',
        PARSER_VERSION='2.0'
    )
    WITH (
        SalesOrderNumber VARCHAR(10) COLLATE Latin1_General_100_BIN2_UTF8,
        SalesOrderLineNumber INT,
        OrderDate DATE,
        CustomerName VARCHAR(25) COLLATE Latin1_General_100_BIN2_UTF8,
        EmailAddress VARCHAR(50) COLLATE Latin1_General_100_BIN2_UTF8,
        Item VARCHAR(30) COLLATE Latin1_General_100_BIN2_UTF8,
        Quantity INT,
        UnitPrice DECIMAL(18,2),
        TaxAmount DECIMAL (18,2)
    ) AS [result]
```

Now the results look like this:

SalesOrderNumber	SalesOrderLineNumber	OrderDate	CustomerName	EmailAddress	Item
SO43701	1	2019-07-01	Christy Zhu	clarence35@adventure-works.com	Mountain-100
...	...	...	...	...	...

8. Publish the changes to your script, and then close the script pane.

Previous Next

On VM: LON-CL1 Page: 5/

Support

Microsoft Azure Port | synapse0gknypd - M | synapse0gknypd - A | New Tab

web.azuresynapse.net/en/authoring/explore/linked/sqlscripts/SQL%20script%201?workspace=%2Fsub...

synapse0gknypd Search XLab-Rfm-108@xtremelabs.us LAB DIRECTORY

Synapse live Validate all Publish all 1

files Sales CSV query

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.

Run Undo Publish Query plan Connect to Built-in

```
SELECT
    TOP 100 *
FROM
    OPENROWSET(
        BULK 'https://datalake0gknypd.dfs.core.windows.net/files/sale
        FORMAT='CSV',
        PARSER_VERSION='2.0'
    )
    WITH (
        SalesOrderNumber VARCHAR(10) COLLATE Latin1_General_100_BIN2_U
        SalesOrderLineNumber INT,
        OrderDate DATE,
        CustomerName VARCHAR(25) COLLATE Latin1_General_100_BIN2_UTF8
        EmailAddress VARCHAR(50) COLLATE Latin1_General_100_BIN2_UTF8,
```

Results Messages

View Table Chart Export results

Search

SalesOrderNu...	SalesOrderLine...	OrderDate	CustomerName	EmailAd
SO49171	1	2021-01-01	Mariah Foster	mariah2
SO49172	1	2021-01-01	Brian Howard	brian23@

00:00:02 Query executed successfully.

Properties

General Related (0)

Name * Sales CSV query

Description

Type .sql script

Size 228 bytes

Results settings per query

First 5000 rows (default)

All rows

Search ENG IN 5:48 PM 12/5/2024

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanc...

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanceG...

Instance ID:6779219

1. In the **files** tab contaning the file system for your data lake, return to the **sales** folder so you can see the **csv**, **json**, and **parquet** folders.

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

3. In the **File type** list, select **Parquet format**, and then apply the settings to open a new SQL script that queries the data in the folder. The script should look similar to this:

Paste Content

```
-- This is auto-generated code
SELECT
    TOP 100 *
FROM
    OPENROWSET(
        BULK 'https://datalakexxxxxxx.dfs.core.windows.net/files/sales/
parquet/**',
        FORMAT = 'PARQUET'
    ) AS [result]
```

4. Run the code, and note that it returns sales order data in the same schema as the CSV files you explored earlier. The schema information is embedded in the parquet file, so the appropriate column names are shown in the results.

5. Modify the code as follows (replacing *datalakexxxxxxx* with the name of your data lake storage account) and then run it.

Paste Content

```
SELECT YEAR(OrderDate) AS OrderYear,
       COUNT(*) AS OrdredItems
FROM
    OPENROWSET(
        BULK 'https://datalakexxxxxxx.dfs.core.windows.net/files/sales/
parquet/**',
        FORMAT = 'PARQUET'
    ) AS [result]
GROUP BY YEAR(OrderDate)
ORDER BY OrderYear
```

6. Note that the results include order counts for all three years - the wildcard used in the BULK path causes the query to return data from all subfolders.
The subfolders reflect *partitions* in the parquet data, which is a technique often used to optimize performance for

On VM: LON-CL1 Page: 6/

Support

Microsoft Azure Port | synapse0gknypd - M | synapse0gknypd - A | New Tab

web.azuresynapse.net/en/authoring/explore/linked/sqlscripts/SQL%20script%201?workspace=%2Fsub...

synapse0gknypd Search XLab-Rfm-108@xtremelabs.us LAB DIRECTORY

Synapse live | Validate all | Publish all 1

files | Sales CSV query | SQL script 1

Run | Undo | Publish | Query plan | Connect to Built-in

```
1  -- This is auto-generated code
2  SELECT
3      TOP 100 *
4  FROM
5      OPENROWSET(
6          BULK 'https://datalake0gknypd.dfs.core.windows.net/files/sales
7          FORMAT = 'PARQUET'
8      ) AS [result]
9
```

Results | Messages

View: Table | Chart | Export results

Search

SalesOrderNu...	SalesOrderLine...	OrderDate	CustomerName	EmailAd
SO43701	1	2019-07-01	Christy Zhu	christy12
SO43704	1	2019-07-01	Julio Ruiz	julio1@a

00:00:02 Query executed successfully.

Properties

General | Related (0)

Name *
SQL script 1

Description

Type
.sql script

Size
203 bytes

Results settings per query
First 5000 rows (default)
All rows

Search ENG IN 5:54 PM 12/5/2024

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanc...

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanceG...

Instance ID:6779219

5. Modify the code as follows (replacing *datalakexxxxxxx* with the name of your data lake storage account) and then run it.

Paste Content

```
SELECT YEAR(OrderDate) AS OrderYear,
       COUNT(*) AS OrdredItems
FROM
    OPENROWSET(
        BULK 'https://datalakexxxxxxx.dfs.core.windows.net/files/sales/
parquet/**',
        FORMAT = 'PARQUET'
    ) AS [result]
GROUP BY YEAR(OrderDate)
ORDER BY OrderYear
```

6. Note that the results include order counts for all three years - the wildcard used in the BULK path causes the query to return data from all subfolders.
The subfolders reflect *partitions* in the parquet data, which is a technique often used to optimize performance for systems that can process multiple partitions of data in parallel. You can also use partitions to filter the data.

7. Modify the code as follows (replacing *datalakexxxxxxx* with the name of your data lake storage account) and then run it.

Paste Content

```
SELECT YEAR(OrderDate) AS OrderYear,
       COUNT(*) AS OrdredItems
FROM
    OPENROWSET(
        BULK 'https://datalakexxxxxxx.dfs.core.windows.net/files/sales/
parquet/year=*/**',
        FORMAT = 'PARQUET'
    ) AS [result]
WHERE [result].filepath(1) IN ('2019', '2020')
GROUP BY YEAR(OrderDate)
ORDER BY OrderYear
```

8. Review the results and note that they include only the sales counts for 2019 and 2020. This filtering is achieved by inclusing a wildcard for the partition folder value in the BULK path (*year=**) and a WHERE clause based on the *filepath* property of the results returned by OPENROWSET (which in this case has the alias *[result]*).

9. Name your script **Sales Parquet query**, and publish it. Then close the script pane.

On VM: LON-CL1 Page: 6/

Support

Microsoft Azure Port | synapse0gknypd - M | synapse0gknypd - A | New Tab

web.azuresynapse.net/en/authoring/explore/linked/sqlscripts/SQL%20script%201?workspace=%2Fsub...

synapse0gknypd Search XLab-Rfm-108@xtremelabs.us LAB DIRECTORY

Synapse live Validate all Publish all 1

files Sales CSV query SQL script 1

Run Undo Publish Query plan Connect to Built-in

```
1
2  SELECT YEAR(OrderDate) AS OrderYear,
3         COUNT(*) AS OrdredItems
4  FROM
5      OPENROWSET(
6          BULK 'https://datalake0gknypd.dfs.core.windows.net/files/sales
7          FORMAT = 'PARQUET'
8      ) AS [result]
9  GROUP BY YEAR(OrderDate)
10 ORDER BY OrderYear
```

Results Messages

View Table Chart Export results

Search

OrderYear	OrdredItems
2019	1201
2020	2733
2021	28784

00:00:03 Query executed successfully.

Properties

General Related (0)

Name * SQL script 1

Description

Type .sql script

Size 203 bytes

Results settings per query
First 5000 rows (default)
All rows

Search ENG IN 5:55 PM 12/5/2024

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanc...

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanceG...

Instance ID:6779219

```
parquet/**',
        FORMAT = 'PARQUET'
    ) AS [result]
GROUP BY YEAR(OrderDate)
ORDER BY OrderYear
```

6. Note that the results include order counts for all three years - the wildcard used in the BULK path causes the query to return data from all subfolders.
   The subfolders reflect *partitions* in the parquet data, which is a technique often used to optimize performance for systems that can process multiple partitions of data in parallel. You can also use partitions to filter the data.

7. Modify the code as follows (replacing *datalakexxxxxxx* with the name of your data lake storage account) and then run it.

Paste Content

```
SELECT YEAR(OrderDate) AS OrderYear,
       COUNT(*) AS OrdredItems
FROM
    OPENROWSET(
        BULK 'https://datalakexxxxxxx.dfs.core.windows.net/files/sales/
parquet/year=*/**',
        FORMAT = 'PARQUET'
    ) AS [result]
WHERE [result].filepath(1) IN ('2019', '2020')
GROUP BY YEAR(OrderDate)
ORDER BY OrderYear
```

8. Review the results and note that they include only the sales counts for 2019 and 2020. This filtering is achieved by inclusing a wildcard for the partition folder value in the BULK path (*year=**) and a WHERE clause based on the *filepath* property of the results returned by OPENROWSET (which in this case has the alias *[result]*).

9. Name your script **Sales Parquet query**, and publish it. Then close the script pane.

Previous Next

On VM: LON-CL1 Page: 6/

Support

Microsoft Azure Port | synapse0gknypd - M | synapse0gknypd - A | New Tab

web.azuresynapse.net/en/authoring/explore/linked/sqlscripts/SQL%20script%201?workspace=%2Fsub...

synapse0gknypd Search XLab-Rfm-108@xtremelabs.us LAB DIRECTORY

Synapse live Validate all Publish all 1

files Sales CSV query SQL script 1

Run Undo Publish Query plan Connect to Built-in

```
1
2  SELECT YEAR(OrderDate) AS OrderYear,
3         COUNT(*) AS OrdredItems
4  FROM
5      OPENROWSET(
6          BULK 'https://datalake0gknypd.dfs.core.windows.net/files/sales
7          FORMAT = 'PARQUET'
8      ) AS [result]
9  WHERE [result].filepath(1) IN ('2019', '2020')
10 GROUP BY YEAR(OrderDate)
11 ORDER BY OrderYear
```

Results Messages

View Table Chart Export results

Search

OrderYear	OrdredItems
2019	1201
2020	2733

00:00:01 Query executed successfully.

Properties

General Related (0)

Name * SQL script 1

Description

Type .sql script

Size 203 bytes

Results settings per query
- First 5000 rows (default)
- All rows

Search ENG IN 5:56 PM 12/5/2024

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanc...

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanceG...

Instance ID:6779219

```
parquet/**',
        FORMAT = 'PARQUET'
    ) AS [result]
GROUP BY YEAR(OrderDate)
ORDER BY OrderYear
```

6. Note that the results include order counts for all three years - the wildcard used in the BULK path causes the query to return data from all subfolders.
   The subfolders reflect *partitions* in the parquet data, which is a technique often used to optimize performance for systems that can process multiple partitions of data in parallel. You can also use partitions to filter the data.

7. Modify the code as follows (replacing *datalakexxxxxxx* with the name of your data lake storage account) and then run it.

   Paste Content

```
SELECT YEAR(OrderDate) AS OrderYear,
       COUNT(*) AS OrdredItems
FROM
    OPENROWSET(
        BULK 'https://datalakexxxxxxx.dfs.core.windows.net/files/sales/
parquet/year=*/**',
        FORMAT = 'PARQUET'
    ) AS [result]
WHERE [result].filepath(1) IN ('2019', '2020')
GROUP BY YEAR(OrderDate)
ORDER BY OrderYear
```

8. Review the results and note that they include only the sales counts for 2019 and 2020. This filtering is achieved by inclusing a wildcard for the partition folder value in the BULK path (*year=**) and a WHERE clause based on the *filepath* property of the results returned by OPENROWSET (which in this case has the alias *[result]*).

9. Name your script **Sales Parquet query**, and publish it. Then close the script pane.

Previous Next

On VM: LON-CL1 Page: 6/

Support

Microsoft Azure Port | synapse0gknypd - M | synapse0gknypd - A | New Tab

web.azuresynapse.net/en/authoring/explore/linked/sqlscripts/SQL%20script%201?workspace=%2Fsub...

synapse0gknypd Search XLab-Rfm-108@xtremelabs.us LAB DIRECTORY

Synapse live Validate all Publish all

files Sales CSV query Sales Parquet query

Publishing completed
Successfully published

Run Undo Publish Query plan Connect to Built-in

```
1
2  SELECT YEAR(OrderDate) AS OrderYear,
3         COUNT(*) AS OrdredItems
4  FROM
5      OPENROWSET(
6          BULK 'https://datalake0gknypd.dfs.core.windows.net/files/sales
7          FORMAT = 'PARQUET'
8      ) AS [result]
9  WHERE [result].filepath(1) IN ('2019', '2020')
10 GROUP BY YEAR(OrderDate)
11 ORDER BY OrderYear
```

Results Messages

View Table Chart Export results

Search

OrderYear	OrdredItems
2019	1201
2020	2733

00:00:01 Query executed successfully.

Properties

General Related (0)

Name *
Sales Parquet query

Description

Type
.sql script

Size
203 bytes

Results settings per query
First 5000 rows (default)
All rows

Search ENG IN 5:56 PM 12/5/2024

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanc...

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanceG...

Instance ID:6779219

the character code *0x0b*.

- Format the results as a single field containing the JSON row of data as an NVARCHAR(MAX) string.

Paste Content

```
SELECT
    TOP 100 *
FROM
    OPENROWSET(
        BULK 'https://datalakexxxxxxx.dfs.core.windows.net/files/sales/
json/**',
        FORMAT = 'CSV',
        FIELDTERMINATOR ='0x0b',
        FIELDQUOTE = '0x0b',
        ROWTERMINATOR = '0x0b'
    ) WITH (Doc NVARCHAR(MAX)) as rows
```

- [x] 5. Run the modified code and observe that the results include a JSON document for each order.

- [ ] 6. Modify the query as follows (replacing *datalakexxxxxxx* with the name of your data lake storage account) so that it uses the JSON_VALUE function to extract individual field values from the JSON data.

Paste Content

```
SELECT JSON_VALUE(Doc, '$.SalesOrderNumber') AS OrderNumber,
       JSON_VALUE(Doc, '$.CustomerName') AS Customer,
       Doc
FROM
    OPENROWSET(
        BULK 'https://datalakexxxxxxx.dfs.core.windows.net/files/sales/
json/**',
        FORMAT = 'CSV',
        FIELDTERMINATOR ='0x0b',
        FIELDQUOTE = '0x0b',
        ROWTERMINATOR = '0x0b'
    ) WITH (Doc NVARCHAR(MAX)) as rows
```

- [ ] 7. Name your script **Sales JSON query**, and publish it. Then close the script pane.

Previous Next

On VM: LON-CL1 Page: 7/

Support

Microsoft Azure Port | synapse0gknypd - M | synapse0gknypd - A | New Tab

web.azuresynapse.net/en/authoring/explore/linked/sqlscripts/SQL%20script%201?workspace=%2Fsub...

synapse0gknypd Search XLab-Rfm-108@xtremelabs.us LAB DIRECTORY

Synapse live Validate all Publish all 1

files | Sales CSV query | Sales Parquet query | SQL script 1

Run Undo Publish Query plan Connect to Built-in

```
1  SELECT
2      TOP 100 *
3  FROM
4      OPENROWSET(
5          BULK 'https://datalake0gknypd.dfs.core.windows.net/files/sales
6          FORMAT = 'CSV',
7          FIELDTERMINATOR ='0x0b',
8          FIELDQUOTE = '0x0b',
9          ROWTERMINATOR = '0x0b'
10     ) WITH (Doc NVARCHAR(MAX)) as rows
```

Results Messages

View Table Chart Export results

Search

Doc
{ "SalesOrderNumber": "SO43703", "SalesOrderLineItem": 1, "OrderDate": "2019-07-01", "...
{ "SalesOrderNumber": "SO43701", "SalesOrderLineItem": 1, "OrderDate": "2019-07-01", "...
{ "SalesOrderNumber": "SO43704", "SalesOrderLineItem": 1, "OrderDate": "2019-07-01", "...

00:00:02 Query executed successfully.

Properties

General Related (0)

Name * SQL script 1

Description

Type .sql script

Size 229 bytes

Results settings per query

- (•) First 5000 rows (default)
- ( ) All rows

Search ENG IN 6:01 PM 12/5/2024

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanc...

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanceG...

Instance ID:6779219

the character code *0x0b*.

- Format the results as a single field containing the JSON row of data as an NVARCHAR(MAX) string.

Paste Content

```
SELECT
    TOP 100 *
FROM
    OPENROWSET(
        BULK 'https://datalakexxxxxxx.dfs.core.windows.net/files/sales/
json/**',
        FORMAT = 'CSV',
        FIELDTERMINATOR ='0x0b',
        FIELDQUOTE = '0x0b',
        ROWTERMINATOR = '0x0b'
    ) WITH (Doc NVARCHAR(MAX)) as rows
```

5. Run the modified code and observe that the results include a JSON document for each order.

6. Modify the query as follows (replacing *datalakexxxxxxx* with the name of your data lake storage account) so that it uses the JSON_VALUE function to extract individual field values from the JSON data.

Paste Content

```
SELECT JSON_VALUE(Doc, '$.SalesOrderNumber') AS OrderNumber,
       JSON_VALUE(Doc, '$.CustomerName') AS Customer,
       Doc
FROM
    OPENROWSET(
        BULK 'https://datalakexxxxxxx.dfs.core.windows.net/files/sales/
json/**',
        FORMAT = 'CSV',
        FIELDTERMINATOR ='0x0b',
        FIELDQUOTE = '0x0b',
        ROWTERMINATOR = '0x0b'
    ) WITH (Doc NVARCHAR(MAX)) as rows
```

7. Name your script **Sales JSON query**, and publish it. Then close the script pane.

Previous Next

On VM: LON-CL1 Page: 7/

Support

Microsoft Azure Port | synapse0gknypd - M | synapse0gknypd - A | New Tab

web.azuresynapse.net/en/authoring/explore/linked/sqlscripts/SQL%20script%201?workspace=%2Fsub...

synapse0gknypd Search XLab-Rfm-108@xtremelabs.us LAB DIRECTORY

Synapse live Validate all Publish all

files Sales CSV query Sales Parquet query

Publishing completed
Successfully published

Run Undo Publish Query plan Connect to Built-in

```
1  SELECT JSON_VALUE(Doc, '$.SalesOrderNumber') AS OrderNumber,
2         JSON_VALUE(Doc, '$.CustomerName') AS Customer,
3         Doc
4  FROM
5      OPENROWSET(
6          BULK 'https://datalake0gknypd.dfs.core.windows.net/files/sales
7          FORMAT = 'CSV',
8          FIELDTERMINATOR ='0x0b',
9          FIELDQUOTE = '0x0b',
10         ROWTERMINATOR = '0x0b'
11     ) WITH (Doc NVARCHAR(MAX)) as rows
```

Results Messages

View Table Chart Export results

Search

OrderNumber	Customer	Doc
SO43703	Albert Alvarez	{ "SalesOrderN...
SO43701	Christy Zhu	{ "SalesOrderN...
SO43704	Julio Ruiz	{ "SalesOrderN...

00:00:02 Query executed successfully.

Properties

General Related (0)

Name *
Sales JSON query

Description

Type
.sql script

Size
229 bytes

Results settings per query
First 5000 rows (default)
All rows

Search ENG IN 6:06 PM 12/5/2024

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanc...

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanceG...

Instance ID:6779219

1. In Synapse Studio, on the **Develop** page, in the **+** menu, select **SQL script**.
2. In the new script pane, add the following code (replacing *datalakexxxxxxx* with the name of your data lake storage account) to create a new database and add an external data source to it.

Paste Content

```
CREATE DATABASE Sales
  COLLATE Latin1_General_100_BIN2_UTF8;
GO;

Use Sales;
GO;

CREATE EXTERNAL DATA SOURCE sales_data WITH (
    LOCATION = 'https://datalakexxxxxxx.dfs.core.windows.net/files/sale
s/'
);
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 ↻ 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 no 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.
7. In the **...** menu for the **Sales** database, select **New SQL script** > **Empty script**. Then in the new script pane, enter and run the following query:

Paste Content

```
SELECT *
FROM
    OPENROWSET(
        BULK 'csv/*.csv',
        DATA_SOURCE = 'sales_data',
        FORMAT = 'CSV',
        PARSER_VERSION = '2.0'
```

On VM: LON-CL1

Page: 8/

Support

Microsoft Azure Port | synapse0gknypd - M | synapse0gknypd - A | New Tab

web.azuresynapse.net/en/authoring/explore/workspace/storageaccounts/synapse0gknypd-Workspace...

synapse0gknypd

Search

XLab-Rfm-108@xtremelabs.us
LAB DIRECTORY

Synapse live | Validate all | Publish all

Data

Workspace | Linked

Filter resources by name

SQL database 1

Sales (SQL)

- External tables
- External resources
- Views
- Schemas
- Security

Sales Parquet query | Sales JSON query | files

New SQL script | New data flow | New integration dataset | More

files > sales

Name	Last Modified	Content Type	Size
csv	12/5/2024, 5:37:35 PM	Folder	
json	12/5/2024, 5:37:54 PM	Folder	
parquet	12/5/2024, 5:37:41 PM	Folder	

Showing 1 to 3 of 3 cached items

Search

ENG
IN

6:08 PM
12/5/2024

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanc...

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanceG...

Instance ID:6779219

to refresh the page. Then view the **Workspace** tab in the **Data** pane, where a **SQL database** list is no 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.

7. In the **...** menu for the **Sales** database, select **New SQL script** > **Empty script**. Then in the new script pane, enter and run the following query:

Paste Content

```
SELECT *
FROM
    OPENROWSET(
        BULK 'csv/*.csv',
        DATA_SOURCE = 'sales_data',
        FORMAT = 'CSV',
        PARSER_VERSION = '2.0'
    ) AS orders
```

The query uses the external data source to connect to the data lake, and the OPENROWSET function now only need to reference the relative path to the .csv files.

8. Modify the code as follows to query the parquet files using the data source.

Paste Content

```
SELECT *
FROM
    OPENROWSET(
        BULK 'parquet/year=*/*.snappy.parquet',
        DATA_SOURCE = 'sales_data',
        FORMAT='PARQUET'
    ) AS orders
WHERE orders.filepath(1) = '2019'
```

Previous Next

On VM: LON-CL1 Page: 8/

Support

Microsoft Azure Port | synapse0gknypd - M | synapse0gknypd - A | New Tab

web.azuresynapse.net/en/authoring/explore/workspace/sqlscripts/SQL%20script%201?workspace=%2...

synapse0gknypd Search XLab-Rfm-108@xtremelabs.us LAB DIRECTORY

Synapse live Validate all Publish all 1

s CSV query | Sales Parquet query | Sales JSON query | files | SQL script 1

Run Undo Publish Query plan Connect to Built-in

```
1  SELECT *
2  FROM
3      OPENROWSET(
4          BULK 'csv/*.csv',
5          DATA_SOURCE = 'sales_data',
6          FORMAT = 'CSV',
7          PARSER_VERSION = '2.0'
8      ) AS orders
```

Results Messages

View Table Chart Export results

Search

C1	C2	C3	C4	C5
SO49171	1	2021-01-01	Mariah Foster	mariah2
SO49172	1	2021-01-01	Brian Howard	brian23@

00:00:04 Query executed successfully.

Properties

General Related (0)

Name * SQL script 1

Description

Type .sql script

Size 0 bytes

Results settings per query

First 5000 rows (default)

All rows

Search ENG IN 6:10 PM 12/5/2024

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanc...

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanceG...

Instance ID:6779219

```
        FORMAT_TYPE = DELIMITEDTEXT,
        FORMAT_OPTIONS(
        FIELD_TERMINATOR = ',',
        STRING_DELIMITER = '"'
        )
    );
GO;

CREATE EXTERNAL TABLE dbo.orders
(
    SalesOrderNumber VARCHAR(10),
    SalesOrderLineNumber INT,
    OrderDate DATE,
    CustomerName VARCHAR(25),
    EmailAddress VARCHAR(50),
    Item VARCHAR(30),
    Quantity INT,
    UnitPrice DECIMAL(18,2),
    TaxAmount DECIMAL (18,2)
)
WITH
(
    DATA_SOURCE =sales_data,
    LOCATION = 'csv/*.csv',
    FILE_FORMAT = CsvFormat
);
GO
```

- [ ] 2. Refresh and expand the **External tables** folder in the **Data** pane and confirm that a table named **dbo.orders** has been created in the **Sales** database.
- [ ] 3. In the **...** menu for the **dbo.orders** table, select **New SQL script** > **Select TOP 100 rows.**
- [ ] 4. Run the SELECT script that has been generated, and verify that it retrieves the first 100 rows of data from the table, which in turn references the files in the data lake.

Note: You should always choose the method that best fits your specific needs and use case. For more detailed information, you can check the How to use OPENROWSET using serverless SQL pool in Azure Synapse Analytics https://learn.microsoft.com/en-us/azure/synapse-analytics/sql/develop-openrowset and Access external storage using serverless SQL pool in Azure Synapse Analytics https://learn.microsoft.com/en-us/azure/synapse-analytics/sql/develop-storage-files-overview?tabs=impersonation articles.

On VM: LON-CL1 Page: 9/

Support

Microsoft Azure Port | synapse0gknypd - M | synapse0gknypd - A | New Tab

web.azuresynapse.net/en/authoring/explore/workspace/sqlscripts/SQL%20script%201?workspace=%2...

synapse0gknypd Search XLab-Rfm-108@xtremelabs.us LAB DIRECTORY

Synapse live | Validate all | Publish all 1

Data

Workspace Linked

Filter resources by name

SQL database 1

Sales (SQL)

External tables

dbo.orders

External resources

Views

Schemas

Security

s CSV query | Sales Parquet query | Sales JSON query | files

Run | Undo | Publish | Query plan

```
15      OrderDate DATE,
16      CustomerName VARCHAR(25),
17      EmailAddress VARCHAR(50),
18      Item VARCHAR(30),
19      Quantity INT,
20      UnitPrice DECIMAL(18,2),
21      TaxAmount DECIMAL (18,2)
22  )
23  WITH
24  (
25      DATA_SOURCE =sales_data,
26      LOCATION = 'csv/*.csv',
27      FILE_FORMAT = CsvFormat
28  );
29  GO
```

Results Messages

No results to show

Your query yielded no displayable results

00:00:01 Query executed successfully.

Properties

General Related (0)

Name * SQL script 1

Description

Type .sql script

Size 0 bytes

Results settings per query

First 5000 rows (default)

All rows

Search ENG IN 6:18 PM 12/5/2024

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanc...

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanceG...

Instance ID:6779219

83% Completed

Lab Time Left: 00:58:05

Lab Actions

10. Module: Model, query, and explore data in Azure Synapse

Visualize query results:

Now that you've explored various ways to query files in the data lake by using SQL queries, you can analyze the results of these queries to gain insights into the data. Often, insights are easier to uncover by visualizing the query results in a chart; which you can easily do by using the integrated charting functionality in the Synapse Studio query editor.

1. On the **Develop** page, create a new empty SQL query.
2. Ensure that the script is connected to the **Built-in** SQL pool and the **Sales** database.
3. Enter the following SQL code:

Paste Content

```
SELECT YEAR(OrderDate) AS OrderYear,
       SUM((UnitPrice * Quantity) + TaxAmount) AS GrossRevenue
FROM dbo.orders
GROUP BY YEAR(OrderDate)
ORDER BY OrderYear;
```

4. In the **Results** pane, select **Chart** and view the chart that is created for you; which should be a line chart.
5. Change the **Category column** to **OrderYear** so that the line chart shows the revenue trend over the three year period from 2019 to 2021:
6. Switch the **Chart type** to **Column** to see the yearly revenue as a column chart:
7. Experiment with the charting functionality in the query editor. It offers some basic charting capabilities that you can use while interactively exploring data, and you can save charts as images to include in reports. However, functionality is limited compared to enterprise data visualization tools such as Microsoft Power BI.

Previous Next

On VM: LON-CL1 Page: 10/

Support

Microsoft Azure Port | synapse0gknypd - M | synapse0gknypd - A | New Tab

web.azuresynapse.net/en/authoring/explore/workspace/sqlscripts/SQL%20script%202?workspace=%2...

synapse0gknypd Search XLab-Rfm-108@xtremelabs.us LAB DIRECTORY

Synapse live Validate all Publish all 2

s Parquet query | Sales JSON query | files | SQL script 1 | SQL script 2

Run Undo Publish Query plan Connect to Built-in

```
1  SELECT TOP (100) [SalesOrderNumber]
2  ,[SalesOrderLineNumber]
3  ,[OrderDate]
4  ,[CustomerName]
5  ,[EmailAddress]
6  ,[Item]
7  ,[Quantity]
8  ,[UnitPrice]
9  ,[TaxAmount]
10 FROM [dbo].[orders]
```

Results Messages

View Table Chart Export results

Search

SalesOrderNu...	SalesOrderLine...	OrderDate	CustomerName	EmailAd
SO49171	1	2021-01-01	Mariah Foster	mariah2
SO49172	1	2021-01-01	Brian Howard	brian23@

00:00:01 Query executed successfully.

Properties

General Related (0)

Name * SQL script 2

Description

Type .sql script

Size 171 bytes

Results settings per query

First 5000 rows (default)

All rows

Search ENG IN 6:18 PM 12/5/2024

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanc...

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanceG...

Instance ID:6779219

92% Completed

Lab Time Left: 00:56:22

Lab Actions

10. Module: Model, query, and explore data in Azure Synapse

Visualize query results:

- [x] Now that you've explored various ways to query files in the data lake by using SQL queries, you can analyze the results of these queries to gain insights into the data. Often, insights are easier to uncover by visualizing the query results in a chart; which you can easily do by using the integrated charting functionality in the Synapse Studio query editor.
- [x] 1. On the **Develop** page, create a new empty SQL query.
- [x] 2. Ensure that the script is connected to the **Built-in** SQL pool and the **Sales** database.
- [x] 3. Enter the following SQL code:

Paste Content

```
SELECT YEAR(OrderDate) AS OrderYear,
       SUM((UnitPrice * Quantity) + TaxAmount) AS GrossRevenue
FROM dbo.orders
GROUP BY YEAR(OrderDate)
ORDER BY OrderYear;
```

- [x] 4. In the **Results** pane, select **Chart** and view the chart that is created for you; which should be a line chart.
- [x] 5. Change the **Category column** to **OrderYear** so that the line chart shows the revenue trend over the three year period from 2019 to 2021:
- [x] 6. Switch the **Chart type** to **Column** to see the yearly revenue as a column chart:
- [x] 7. Experiment with the charting functionality in the query editor. It offers some basic charting capabilities that you can use while interactively exploring data, and you can save charts as images to include in reports. However, functionality is limited compared to enterprise data visualization tools such as Microsoft Power BI.

Previous Next

On VM: LON-CL1 Page: 10/

Support

Microsoft Azure Port | synapse0gknypd - M | synapse0gknypd - A | New Tab

web.azuresynapse.net/en/authoring/analyze/sqlscripts/SQL%20script%203?workspace=%2Fsubscriptio...

synapse0gknypd Search XLab-Rfm-108@xtremelabs.us LAB DIRECTORY

Synapse live Validate all Publish all 3

s JSON query | files | SQL script 1 | SQL script 2 | SQL script 3

Run Undo Publish Query plan Connect to Built-in

```
1  SELECT YEAR(OrderDate) AS OrderYear,
2         SUM((UnitPrice * Quantity) + TaxAmount) AS GrossRevenue
3  FROM dbo.orders
4  GROUP BY YEAR(OrderDate)
5  ORDER BY OrderYear;
```

Results Messages

View Table Chart Save as image

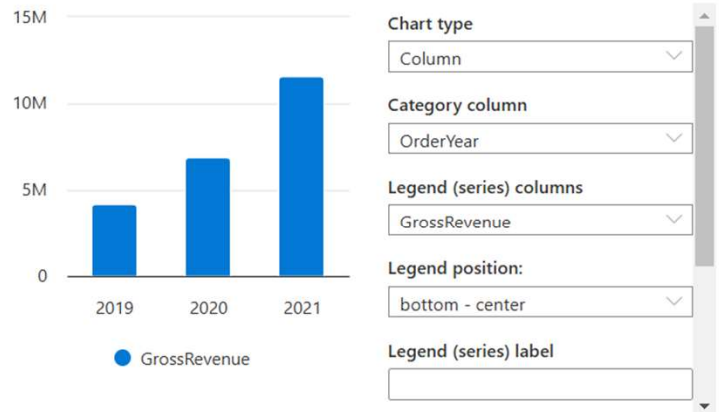


Chart type: Column

Category column: OrderYear

Legend (series) columns: GrossRevenue

Legend position: bottom - center

Legend (series) label:

00:00:02 Query executed successfully.

Properties

General Related (0)

Name * SQL script 3

Description

Type .sql script

Size 0 bytes

Results settings per query

- (•) First 5000 rows (default)
- ( ) All rows

Search ENG IN 6:20 PM 12/5/2024

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanc...

labs.xtremelabs.io/LabViewerConnection/DetachLabManual?labInstanceG...

Instance ID:6779219

92% Completed

Lab Time Left: 00:56:22

Lab Actions

10. Module: Model, query, and explore data in Azure Synapse

Visualize query results:

- [x] Now that you've explored various ways to query files in the data lake by using SQL queries, you can analyze the results of these queries to gain insights into the data. Often, insights are easier to uncover by visualizing the query results in a chart; which you can easily do by using the integrated charting functionality in the Synapse Studio query editor.
- [x] 1. On the **Develop** page, create a new empty SQL query.
- [x] 2. Ensure that the script is connected to the **Built-in** SQL pool and the **Sales** database.
- [x] 3. Enter the following SQL code:

Paste Content

```
SELECT YEAR(OrderDate) AS OrderYear,
       SUM((UnitPrice * Quantity) + TaxAmount) AS GrossRevenue
FROM dbo.orders
GROUP BY YEAR(OrderDate)
ORDER BY OrderYear;
```

- [x] 4. In the **Results** pane, select **Chart** and view the chart that is created for you; which should be a line chart.
- [x] 5. Change the **Category column** to **OrderYear** so that the line chart shows the revenue trend over the three year period from 2019 to 2021:
- [x] 6. Switch the **Chart type** to **Column** to see the yearly revenue as a column chart:
- [x] 7. Experiment with the charting functionality in the query editor. It offers some basic charting capabilities that you can use while interactively exploring data, and you can save charts as images to include in reports. However, functionality is limited compared to enterprise data visualization tools such as Microsoft Power BI.

Previous Next

On VM: LON-CL1 Page: 10/

Support

Microsoft Azure Port | synapse0gknypd - M | synapse0gknypd - A | New Tab

web.azuresynapse.net/en/authoring/analyze/sqlscripts/SQL%20script%203?workspace=%2Fsubscriptio...

synapse0gknypd Search XLab-Rfm-108@xtremelabs.us LAB DIRECTORY

Synapse live Validate all Publish all 3

s JSON query | files | SQL script 1 | SQL script 2 | SQL script 3

Run Undo Publish Query plan Connect to Built-in

```
1  SELECT YEAR(OrderDate) AS OrderYear,
2         SUM((UnitPrice * Quantity) + TaxAmount) AS GrossRevenue
3  FROM dbo.orders
4  GROUP BY YEAR(OrderDate)
5  ORDER BY OrderYear;
```

Results Messages

View Table Chart Save as image

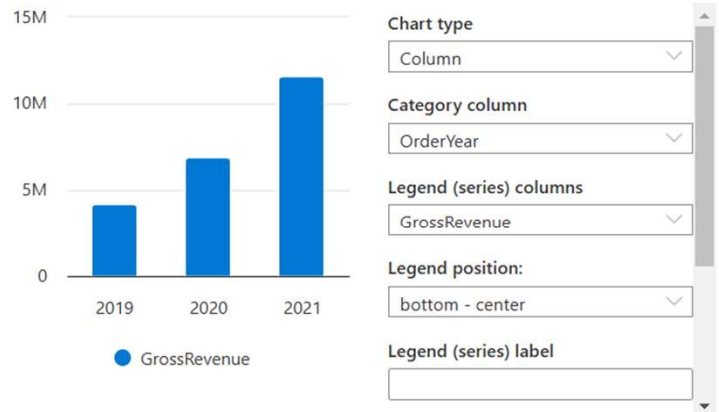


Chart type: Column

Category column: OrderYear

Legend (series) columns: GrossRevenue

Legend position: bottom - center

Legend (series) label:

00:00:02 Query executed successfully.

Properties

General Related (0)

Name * SQL script 3

Description

Type .sql script

Size 0 bytes

Results settings per query

- (•) First 5000 rows (default)
- ( ) All rows

Search ENG IN 6:20 PM 12/5/2024

Assignments - Non-Credit 2024 | XtremeLabs - Labs

labs.xtremelabs.io/LabMenu/?track=DP-500T00-A-CEP

This training session brought to you by Southern Alberta Institute of Technology (XTP) Powered by XTREMELABS

Nitin Nitin NN

View Labs

Access Codes

History

Support

FAQs

Search for courses

DP-203T00-A-CEP

DP-500T00-A-CEP

DP-900T00-A-CEP

Course DP-500T00: Designing and Implementing Enterprise-Scale Analytics Solutions Using Microsoft Azure and Microsoft Power BI

Go To Lesson

Lab Title: [DP-500T00-A-CEP-MOD-01] Query Data by using Azure Synapse Analytics
Duration: 120 minutes
Status : **Completed**
Take Lab

Lab Title: [DP-500T00-A-CEP-MOD-02] Explore data by using native visuals in Spark notebooks
Duration: 120 minutes
Status : Not Initiated
Take Lab

Lab Title: [DP-500T00-A-CEP-MOD-03] Analyze data in a relational data warehouse
Duration: 120 minutes
Status : Not Initiated
Take Lab

Lab Title: [DP-500T00-A-CEP-MOD-04] Create a star schema model
Duration: 120 minutes
Status : Not Initiated
Take Lab

Lab Title: [DP-500T00-A-CEP-MOD-05]

Lab Title: [DP-500T00-A-CEP-MOD-06] Work

Classroom Chat

Search

ENG IN

6:30 PM 12/5/2024