Analyze Data in a Data Lake with Spark

SUBMITTED BY

NITIN

OBJECTIVES

I am demonstrating data analysis skills in a data lake using Spark, focusing on data management, querying, and visualization for large-scale tasks. This includes setting up Azure Synapse Analytics, performing data manipulations, and using Python libraries like Matplotlib and Seaborn for visualizations. The goal is to enhance practical experience in cloud resource management, complex data operations, and insightful data presentations.

labs.xtremelabs.io/LabViewerConnection/DetachLabManual...

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

Instance ID:6747987
XtremeLabs: DP-203T00-A-CEP [DP-203T00-A-M05-CEP] Module 05: Analyze data in a data lake with Spark

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

Paste Content
Paste Content

```
cd dp-203/Allfiles/labs/05
./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 Apache Spark in Azure Synapse Analytics article in the Azure Synapse Analytics documentation.

Previous Exercise Next Exercise

Page: 3/13

Support

Home - Microsoft Azure

portal.azure.com/#home

Microsoft Azure Search resources, services, and docs (G+/) Copilot

Azure services

Create a resource | Quickstart Center | Azure AI services | Kubernetes services | Virtual machines | App Services | Storage accounts | SQL databases

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

```
2021.csv

ICloudBlob                          : Microsoft.Azure.Storage.Blob.CloudBlockBlob
BlobType                            : BlockBlob
Length                              : 2869399
IsDeleted                           : False
BlobClient                          : Azure.Storage.Blobs.BlobClient
BlobBaseClient                      : Azure.Storage.Blobs.Specialized.BlockBlobClient
BlobProperties                      : Azure.Storage.Blobs.Models.BlobProperties
RemainingDaysBeforePermanentDelete  :
ContentType                         : application/octet-stream
LastModified                        : 11/4/2024 10:52:53 PM +00:00
SnapshotTime                        :
ContinuationToken                   :
VersionId                           :
IsLatestVersion                     :
AccessTier                          : Hot
TagCount                            : 0
Tags                                :
ListBlobProperties                  :
Context                             : Microsoft.WindowsAzure.Commands.Storage.Common.AzureStorageContext
Name                                : sales/orders/2021.csv

Script completed at 11/04/2024 22:52:54
PS /home/xlab-job-901/dp-203/Allfiles/labs/05>
```

Search ENG IN 3:53 PM 11/4/2024

labs.xtremelabs.io/LabViewerConnection/DetachL...

labs.xtremelabs.io/LabViewerConnection/DetachLabMa...

Instance ID:6748012
XtremeLabs: DP-203T00-A-CEP [DP-203T00-A-M05-CEP] Module 05: Analyze data in a data lake with Spark

- [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 **Manage** page, select the **Apache Spark pools** tab and note that a Spark pool with a name similar to **spark*xxxxxxx*** has been provisioned in the workspace. Later you will use this Spark pool to load and analyze data from files in the data lake storage for the workspace.
- [x] 5. 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] 6. Expand your storage account and verify that it contains a file system container named **files**.
- [x] 7. Select the **files** container, and note that it contains folders named **sales** and **synapse**. The **synapse** folder is used by Azure Synapse, and the **sales** folder contains the data files you are going to query.
- [x] 8. Open the **sales** folder and the **orders** folder it contains, and observe that the **orders** folder contains .csv files for three years of sales data.
- [ ] 9. Right-click any of the files and select **Preview** to see the data it contains. Note that the files do

Page: 4/13

Support

synapseclhz81s - Microsoft Azu

synapseclhz81s - Azure Synapse

web.azuresynapse.net/en/authoring/explore/linked/storageaccounts/synapseclhz81s-WorkspaceDefaultStorage-datalake...

Microsoft Azure | Synapse Analytics ▸ synap

Search

XLab-MUR-945@xtremelabs.us
LAB DIRECTORY

Synapse live | Validate all | Publish all

Data

Workspace | Linked

Filter resources by name

Azure Data Lake Storage Gen2 2

synapseclhz81s (Primary - datalake...

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 notebook | New data flow | More

files › sales › orders

Name	Last Modified	Content Type	Size
2019.csv	11/4/2024, 4:12:31 PM		119.8 KB
2020.csv	11/4/2024, 4:12:33 PM		276.8 KB
2021.csv	11/4/2024, 4:12:33 PM		2.7 MB

Showing 1 to 3 of 3 cached items

Search

ENG
IN

4:15 PM
11/4/2024

labs.xtremelabs.io/LabViewerConnection/DetachL...

labs.xtremelabs.io/LabViewerConnection/DetachLabMa...

Instance ID:6748012
XtremeLabs: DP-203T00-A-CEP [DP-203T00-A-M05-CEP] Module 05: Analyze data in a data lake with Spark

approach to try to determine appropriate data types for the columns based on the data they contain, and if a header row is present in a text file it can be used to identify the column names (by specifying a **header**=**True** parameter in the **load** function). Alternatively, you can define an explicit schema for the dataframe.

7. Modify the code as follows (replacing *datalakexxxxxxx*), to define an explicit schema for the dataframe that includes the column names and data types. Rerun the code in the cell.

Paste Content
Paste Content

```
%%pyspark
from pyspark.sql.types import *
from pyspark.sql.functions import *

orderSchema = StructType([
    StructField("SalesOrderNumber", String
Type()),
    StructField("SalesOrderLineNumber", In
tegerType()),
    StructField("OrderDate", DateType()),
    StructField("CustomerName", StringType
()),
    StructField("Email", StringType()),
    StructField("Item", StringType()),
    StructField("Quantity", IntegerType
()),
    StructField("UnitPrice", FloatType()),
    StructField("Tax", FloatType())
    ])

df = spark.read.load('abfss://files@datala
kexxxxxxx.dfs.core.windows.net/sales/order
s/*.csv', format='csv', schema=orderSchem
a)
```

Page: 5/13

Support

synapseclhz81s - Microsoft Azu

synapseclhz81s - Azure Synapse

web.azuresynapse.net/en/authoring/explore/linked/notebooks/Notebook%201?workspace=%2Fsubscriptions%2F49b69...

Microsoft Azure | Synapse Analytics | synap | Search | XLab-MUR-945@xtremelabs.us LAB DIRECTORY

Synapse live | Validate all | Publish all 1

files | Notebook 1

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 all | Undo | Publish | Outline | Attach to sparkclhz81s | Language PySpark (Python)

Ready

```
1  %%pyspark
2  df = spark.read.load('abfss://files@datalakeclhz81s.dfs.core.windows.net/sales/orders/2019.csv', format='csv'
3  )
4  display(df.limit(100))
```

[3] 2 sec - Command executed in 1 sec 892 ms by XLab-MUR-945 on 4:21:57 PM, 11/04/24

Job execution Succeeded **Spark** 2 executors 8 cores View in monitoring Open Spark UI

View: Table | Chart | Export results

_c0	_c1	_c2	_c3
SO43701	1	2019-07-01	Christy Zhu
SO43704	1	2019-07-01	Julio Ruiz
SO43705	1	2019-07-01	Curtis Lu
SO43700	1	2019-07-01	Ruben Prasad
SO43703	1	2019-07-01	Albert Alvarez
SO43697	1	2019-07-01	Cole Watson
SO43699	1	2019-07-01	Sydney Wright
SO43702	1	2019-07-01	Colin Anand

Search

ENG IN 4:22 PM 11/4/2024

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

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

Instance ID:6748147

XtremeLabs: DP-203T00-A-CEP [DP-203T00-A-M05-CEP] Module 05: Analyze data in a data lake with Spark

Paste Content

```
%%pyspark
df = spark.read.load('abfss://files@datalake
xxxxxxx.dfs.core.windows.net/sales/orders/20
19.csv', format='csv'
## If header exists uncomment line below
##, header=True
)
display(df.limit(10))
```

4. When the code has finished running, and then review the output beneath the cell in the notebook. It shows the first ten rows in the file you selected, with automatic column names in the form **_c0**, **_c1**, **_c2**, and so on.

5. Modify the code so that the **spark.read.load** function reads data from all of the CSV files in the folder, and the **display** function shows the first 100 rows. Your code should look like this (with *datalakexxxxxxx* matching the name of your data lake store):

Paste Content

Paste Content

```
%%pyspark
df = spark.read.load('abfss://files@datalake
xxxxxxx.dfs.core.windows.net/sales/orders/*.
csv', format='csv'
)
display(df.limit(100))
```

Page: 5/13

Support

synapse8w7bzrq - Microsoft Az

synapse8w7bzrq - Azure Synaps

web.azuresynapse.net/en/authoring/explore/linked/notebooks/Notebook%201?workspace=%2Fsubscript...

Microsoft Azure | Synapse Analytics | synapse8w7b | Search | XLab-OcW-933@xtremelabs.us LAB DIRECTORY

Synapse live | Validate all | Publish all 1

files | Notebook 1

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 all | Undo | Publish | Outline | Attach to spark8w7bzrq | Language PySpark (Python) | Variables

Ready

```
1  %%pyspark
2  df = spark.read.load('abfss://files@datalake8w7bzrq.dfs.core.windows.net/sales/orders/*.csv', format='csv'
3  )
4  display(df.limit(100))
```

[2] 6 sec - Command executed in 6 sec 8 ms by XLab-OcW-933 on 7:08:16 PM, 11/04/24

Job execution Succeeded **Spark** 2 executors 8 cores View in monitoring Open Spark UI

View: Table | Chart | Export results

_c0	_c1	_c2	_c3	_c4
SO49171	1	2021-01-01	Mariah Foster	mariah
SO49172	1	2021-01-01	Brian Howard	brian2
SO49173	1	2021-01-01	Linda Alvarez	linda1
SO49174	1	2021-01-01	Gina Hernandez	gina4
SO49178	1	2021-01-01	Beth Ruiz	beth4
SO49179	1	2021-01-01	Evan Ward	evan1
SO49175	1	2021-01-01	Margaret Guo	marga
SO49180	1	2021-01-01	Mitchell Yuan	mitch
SO49176	1	2021-01-01	Shawn Sharma	shawn
SO49177	1	2021-01-01	Barbara Chande	barba

Search

ENG IN 7:09 PM 11/4/2024

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

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

Instance ID:6748147

XtremeLabs: DP-203T00-A-CEP [DP-203T00-A-M05-CEP] Module 05: Analyze data in a data lake with Spark

```
pe()),
    StructField("SalesOrderLineNumber", Inte
gerType()),
    StructField("OrderDate", DateType()),
    StructField("CustomerName", StringType
()),
    StructField("Email", StringType()),
    StructField("Item", StringType()),
    StructField("Quantity", IntegerType()),
    StructField("UnitPrice", FloatType()),
    StructField("Tax", FloatType())
    ])

df = spark.read.load('abfss://files@datalake
xxxxxxx.dfs.core.windows.net/sales/orders/*.
csv', format='csv', schema=orderSchema)
display(df.limit(100))
```

8. Under the results, use the + **Code** button to add a new code cell to the notebook. Then in the new cell, add the following code to display the dataframe's schema:

Paste Content

Paste Content

```
df.printSchema()
```

9. Run the new cell and verify that the dataframe schema matches the **orderSchema** you defined. The **printSchema** function can be useful when using a dataframe with an automatically inferred

Page: 5/13

Support

synapse8w7bzrq - Microsoft Az

synapse8w7bzrq - Azure Synaps

web.azuresynapse.net/en/authoring/explore/linked/notebooks/Notebook%201?workspace=%2Fsubscript...

Microsoft Azure | Synapse Analytics | synapse8w7b | Search | XLab-OcW-933@xtremelabs.us LAB DIRECTORY

Synapse live | Validate all | Publish all 1

files | Notebook 1

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 all | Undo | Publish | Outline | Attach to spark8w7bzrq | Language PySpark (Python) | Variables

Ready

```
9       StructField("CustomerName", StringType()),
10      StructField("Email", StringType()),
11      StructField("Item", StringType()),
12      StructField("Quantity", IntegerType()),
13      StructField("UnitPrice", FloatType()),
14      StructField("Tax", FloatType())
15      ])
16
17  df = spark.read.load('abfss://files@datalake8w7bzrq.dfs.core.windows.net/sales/orders/*.csv', format='csv', schema=or
18  display(df.limit(100))
```

[3] 2 sec - Command executed in 1 sec 876 ms by XLab-OcW-933 on 7:10:42 PM, 11/04/24

Job execution Succeeded **Spark** 2 executors 8 cores View in monitoring Open Spark UI

View: Table | Chart | Export results

SalesOrderNumber	SalesOrderLineNumber	OrderDate	CustomerName	Email
SO49171	1	2021-01-01	Mariah Foster	mariah
SO49172	1	2021-01-01	Brian Howard	brian2
SO49173	1	2021-01-01	Linda Alvarez	linda1
SO49174	1	2021-01-01	Gina Hernandez	gina4(
SO49178	1	2021-01-01	Beth Ruiz	beth4(
SO49179	1	2021-01-01	Evan Ward	evan1
SO49175	1	2021-01-01	Margaret Guo	marga
SO49180	1	2021-01-01	Mitchell Yuan	mitche

Search

ENG IN 7:10 PM 11/4/2024

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

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

Instance ID:6748147
XtremeLabs: DP-203T00-A-CEP [DP-203T00-A-M05-CEP] Module 05: Analyze data in a data lake with Spark

43% Completed

Lab Time Left: 01:21:20

Lab Actions

6. Module 5: Analyze data in a data lake with Spark:

Analyze data in a dataframe:

The **dataframe** object in Spark is similar to a Pandas dataframe in Python, and includes a wide range of functions that you can use to manipulate, filter, group, and otherwise analyze the data it contains.

Filter a dataframe

1. Add a new code cell to the notebook, and enter the following code in it:

Paste Content

Paste Content

```
customers = df['CustomerName', 'Email']
print(customers.count())
print(customers.distinct().count())
display(customers.distinct())
```

2. Run the new code cell, and review the results. Observe the following details:
   - When you perform an operation on a dataframe, the result is a new dataframe (in this case, a new **customers** dataframe is

Page: 6/13

Support

synapse8w7bzrq - Microsoft Az

synapse8w7bzrq - Azure Synaps

web.azuresynapse.net/en/authoring/explore/linked/notebooks/Notebook%201?workspace=%2Fsubscript...

Microsoft Azure | Synapse Analytics | synapse8w7b | Search | XLab-OcW-933@xtremelabs.us LAB DIRECTORY

Synapse live | Validate all | Publish all 1

files | Notebook 1

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 all | Undo | Publish | Outline | Attach to spark8w7bzrq | Language PySpark (Python) | Variables

Ready

```
1  customers = df['CustomerName', 'Email']
2  print(customers.count())
3  print(customers.distinct().count())
4  display(customers.distinct())
```

[5] 8 sec - Command executed in 8 sec 697 ms by XLab-OcW-933 on 7:12:17 PM, 11/04/24

Job execution Succeeded **Spark** 2 executors 8 cores View in monitoring Open Spark UI

32718
12427

View: Table | Chart | Export results

CustomerName	Email
Bridget Andersen	bridget15@adventure-works.com
Mya Butler	mya14@adventure-works.com
Deanna Hernandez	deanna29@adventure-works.com
Ricky Navarro	ricky10@adventure-works.com
Omar Ye	omar9@adventure-works.com
Kellie Gutierrez	kellie9@adventure-works.com
Raymond Rana	raymond13@adventure-works.c...
Derrick Moreno	derrick6@adventure-works.com
Megan Walker	megan25@adventure-works.com
Edward Jackson	edward34@adventure-works.com

Search

ENG IN 7:12 PM 11/4/2024

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

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

Instance ID:6748147

XtremeLabs: DP-203T00-A-CEP [DP-203T00-A-M05-CEP] Module 05: Analyze data in a data lake with Spark

and filter the data they contain.

- The `dataframe['Field1', 'Field2', ...]` syntax is a shorthand way of defining a subset of column. You can also use **select** method, so the first line of the code above could be written as `customers = df.select("CustomerName", "Email")`

3. Modify the code as follows:

Paste Content

Paste Content

```
customers = df.select("CustomerName", "Email").where(df['Item']=='Road-250 Red, 52')
print(customers.count())
print(customers.distinct().count())
display(customers.distinct())
```

4. Run the modified code to view the customers who have purchased the *Road-250 Red, 52* product. Note that you can "chain" multiple functions together so that the output of one function becomes the input for the next - in this case, the dataframe created by the **select** method is the source dataframe for the **where** method that is used to apply filtering criteria.

Page: 6/13

Support

synapse8w7bzrq - Microsoft Az

synapse8w7bzrq - Azure Synaps

web.azuresynapse.net/en/authoring/explore/linked/notebooks/Notebook%201?workspace=%2Fsubscript...

Microsoft Azure | Synapse Analytics | synapse8w7b | Search | XLab-OcW-933@xtremelabs.us LAB DIRECTORY

Synapse live | Validate all | Publish all 1

files | Notebook 1

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 all | Undo | Publish | Outline | Attach to spark8w7bzrq | Language PySpark (Python) | Variables

Ready

```
1 customers = df.select("CustomerName", "Email").where(df['Item']=='Road-250 Red, 52')
2 print(customers.count())
3 print(customers.distinct().count())
4 display(customers.distinct())
```

[6] 4 sec - Command executed in 3 sec 965 ms by XLab-OcW-933 on 7:12:56 PM, 11/04/24

Job execution Succeeded **Spark** 2 executors 8 cores View in monitoring Open Spark UI

133
133

View: Table | Chart | Export results

CustomerName	Email
Briana Ashe	briana4@adventure-works.com
Ann Madan	ann13@adventure-works.com
Colleen Andersen	colleen36@adventure-works.com
Barbara Shen	barbara33@adventure-works.com
Evelyn Subram	evelyn14@adventure-works.com
Carolyn Rodriguez	carolyn19@adventure-works.com
Joan Hernandez	joan15@adventure-works.com
Brandon Jai	brandon28@adventure-works.com
Isaiah Howard	isaiah10@adventure-works.com
Robyn Gomez	robyn0@adventure-works.com

Search

ENG IN 7:13 PM 11/4/2024

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

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

Instance ID:6748147

XtremeLabs: DP-203T00-A-CEP [DP-203T00-A-M05-CEP] Module 05: Analyze data in a data lake with Spark

columns (in this case, *Quantity*)

3. Add another new code cell to the notebook, and enter the following code in it:

Paste Content

Paste Content

```
yearlySales = df.select(year("OrderDate").alias("Year")).groupBy("Year").count().orderBy("Year")
display(yearlySales)
```

4. Run the code cell you added, and note that the results show the number of sales orders per year. Note that the **select** method includes a SQL **year** function to extract the year component of the *OrderDate* field, and then an **alias** method is used to assign a columm name to the extracted year value. The data is then grouped by the derived *Year* column and the count of rows in each group is calculated before finally the **orderBy** method is used to sort the resulting dataframe.

Page: 7/13

Support

synapse8w7bzrq - Microsoft Az

synapse8w7bzrq - Azure Synaps

web.azuresynapse.net/en/authoring/explore/linked/notebooks/Notebook%201?workspace=%2Fsubscript...

Microsoft Azure | Synapse Analytics | synapse8w7b | Search | XLab-OcW-933@xtremelabs.us LAB DIRECTORY

Synapse live | Validate all | Publish all 1

files | Notebook 1

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 all | Undo | Publish | Outline | Attach to spark8w7bzrq | Language PySpark (Python) | Variables

Ready

Mountain-100 Black, 38	49
Road-650 Black, 60	76

```
1  yearlySales = df.select(year("OrderDate").alias("Year")).groupBy("Year").count().orderBy("Year")
2  display(yearlySales)
```

[8] 3 sec - Command executed in 2 sec 754 ms by XLab-OcW-933 on 7:14:07 PM, 11/04/24

Job execution Succeeded **Spark** 2 executors 8 cores View in monitoring Open Spark UI

View: Table | Chart | Export results

Year	count
2019	1201
2020	2733
2021	28784

Search

ENG IN 7:14 PM 11/4/2024

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

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

Instance ID:6748147

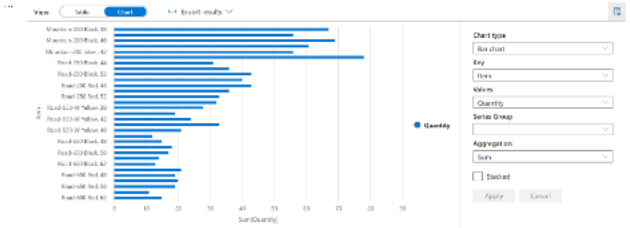
XtremeLabs: DP-203T00-A-CEP [DP-203T00-A-M05-CEP] Module 05: Analyze data in a data lake with Spark

2. Run the code and observe that it returns the data from the **salesorders** view you created previously.

3. In the results section beneath the cell, change the **View** option from **Table** to **Chart**.

4. Use the **View options** button at the top right of the chart to display the options pane for the chart. Then set the options as follows and select **Apply**:
   - **Chart type**: Bar chart
   - **Key**: Item
   - **Values**: Quantity
   - **Series Group**: *leave blank*
   - **Aggregation**: Sum
   - **Stacked**: *Unselected*

5. Verify that the chart looks similar to this:



Page: 10/13

Support

synapse8w7bzrq - Microsoft Az

synapse8w7bzrq - Azure Synaps

web.azuresynapse.net/en/authoring/explore/linked/notebooks/Notebook%201?workspace=%2Fsubscript...

Microsoft Azure | Synapse Analytics | synapse8w7b | Search

XLab-OcW-933@xtremelabs.us
LAB DIRECTORY

Synapse live | Validate all | Publish all 1

files | Notebook 1

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 all | Undo | Publish | Outline | Attach to spark8w7bzrq | Language PySpark (Python) | Variables

Ready

```
2  SELECT * FROM salesorders
```

[11] 2 sec - Command executed in 1 sec 870 ms by XLab-OcW-933 on 7:15:47 PM, 11/04/24

Job execution Succeeded **Spark** 2 executors 8 cores View in monitoring Open Spark UI

View: Table | Chart | Export results

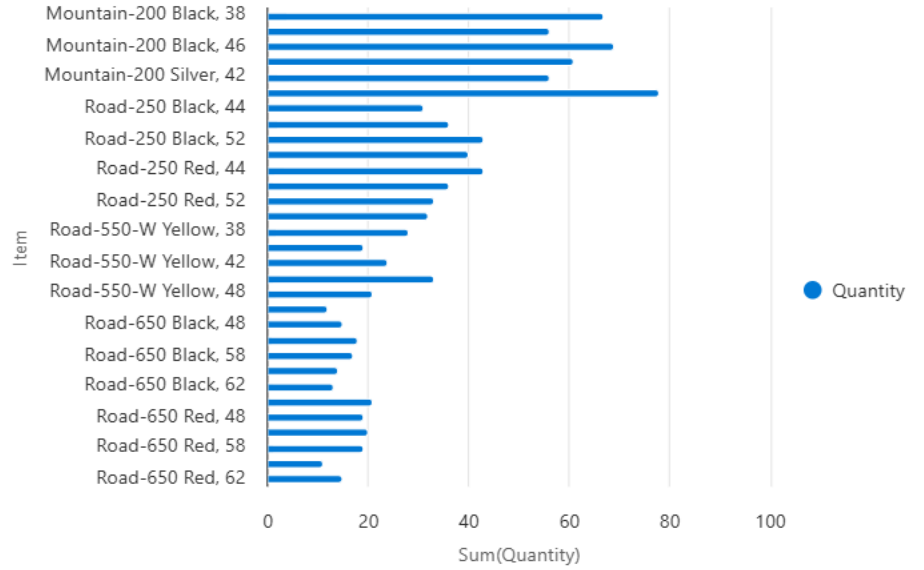


Chart type: Bar chart

Key: Item

Values: Quantity

Series Group:

Aggregation: Sum

Stacked

Apply | Cancel

Search

ENG IN

7:16 PM 11/4/2024

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

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

Instance ID:6748147
XtremeLabs: DP-203T00-A-CEP [DP-203T00-A-M05-CEP] Module 05: Analyze data in a data lake with Spark

72% Completed

Lab Time Left: 01:16:09

Lab Actions

11. Module 5: Analyze data in a data lake with Spark:

Get started with `matplotlib`:

1. Add a new code cell to the notebook, and enter the following code in it:

Paste Content
Paste Content

```
sqlQuery = "SELECT CAST(YEAR(OrderDate) AS C
HAR(4)) AS OrderYear, \
                SUM((UnitPrice * Quantity) +
Tax) AS GrossRevenue \
            FROM salesorders \
            GROUP BY CAST(YEAR(OrderDate) AS
CHAR(4)) \
            ORDER BY OrderYear"
df_spark = spark.sql(sqlQuery)
df_spark.show()
```

2. Run the code and observe that it returns a Spark dataframe containing the yearly revenue.
To visualize the data as a chart, we'll start by using the **matplotlib** Python library. This library is the core plotting library on which many others are based, and provides a great deal of flexibility in creating charts.

Page: 11/13

Support

synapse8w7bzrq - Microsoft Az

synapse8w7bzrq - Azure Synaps

web.azuresynapse.net/en/authoring/explore/linked/notebooks/Notebook%201?workspace=%2Fsubscript...

Microsoft Azure | Synapse Analytics | synapse8w7b | Search | XLab-OcW-933@xtremelabs.us LAB DIRECTORY

Synapse live | Validate all | Publish all 1

files | Notebook 1 ●

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 all | Undo | Publish | Outline | Attach to spark8w7bzrq | Language PySpark (Python) | Variables

Ready

```
1  sqlQuery = "SELECT CAST(YEAR(OrderDate) AS CHAR(4)) AS OrderYear, \
2                SUM((UnitPrice * Quantity) + Tax) AS GrossRevenue \
3            FROM salesorders \
4            GROUP BY CAST(YEAR(OrderDate) AS CHAR(4)) \
5            ORDER BY OrderYear"
6  df_spark = spark.sql(sqlQuery)
7  df_spark.show()
```

[12] ✓ 1 sec - Command executed in 1 sec 825 ms by XLab-OcW-933 on 7:17:22 PM, 11/04/24

Job execution Succeeded **Spark** 2 executors 8 cores View in monitoring Open Spark UI

```
+---------+-------------------+
|OrderYear|       GrossRevenue|
+---------+-------------------+
|     2019|  4172169.969970703|
|     2020|  6882259.268127441|
|     2021|1.1547835291696548E7|
+---------+-------------------+
```

Search

ENG IN 7:17 PM 11/4/2024

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

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

Instance ID:6748147

XtremeLabs: DP-203T00-A-CEP [DP-203T00-A-M05-CEP] Module 05: Analyze data in a data lake with Spark

```
t a Spark one
df_sales = df_spark.toPandas()

# Create a bar plot of revenue by year
plt.bar(x=df_sales['OrderYear'], height=df_s
ales['GrossRevenue'])

# Display the plot
plt.show()
```

4. Run the cell and review the results, which consist of a column chart with the total gross revenue for each year. Note the following features of the code used to produce this chart:
   - The **matplotlib** library requires a *Pandas* dataframe, so you need to convert the *Spark* dataframe returned by the Spark SQL query to this format.
   - At the core of the **matplotlib** library is the **pyplot** object. This is the foundation for most plotting functionality.
   - The default settings result in a usable chart, but there's considerable scope to customize it

5. Modify the code to plot the chart as follows:

Paste Content

Paste Content

```
# Cl
```

Page: 11/13

Support

synapse8w7bzrq - Microsoft Az

synapse8w7bzrq - Azure Synaps

web.azuresynapse.net/en/authoring/explore/linked/notebooks/Notebook%201?workspace=%2Fsubscript...

Microsoft Azure | Synapse Analytics ▸ synapse8w7bzrq

Search

XLab-OcW-933@xtremelabs.us
LAB DIRECTORY

Synapse live | Validate all | Publish all 1

files | Notebook 1

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.

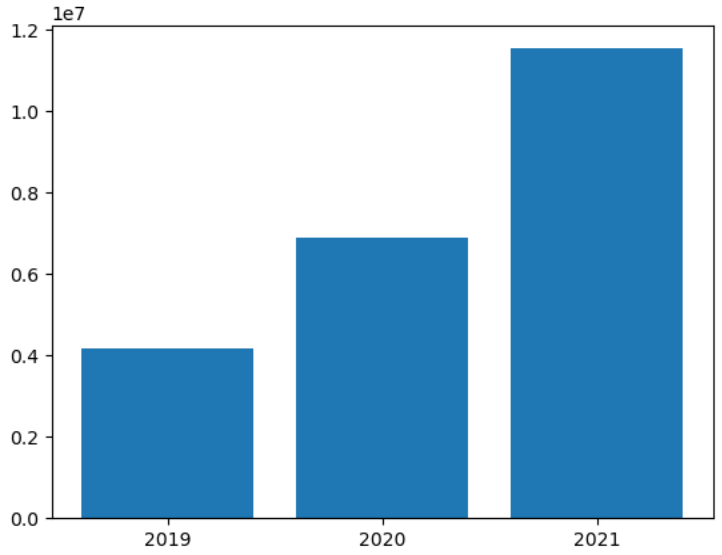
Run all | Undo | Publish | Outline | Attach to spark8w7bzrq | Language PySpark (Python) | Variables

Ready

```
2
3  # matplotlib requires a Pandas dataframe, not a Spark one
4  df_sales = df_spark.toPandas()
5
6  # Create a bar plot of revenue by year
7  plt.bar(x=df_sales['OrderYear'], height=df_sales['GrossRevenue'])
8
9  # Display the plot
10 plt.show()
```

[13] 4 sec - Command executed in 3 sec 933 ms by XLab-OcW-933 on 7:18:14 PM, 11/04/24

Job execution Succeeded **Spark** 2 executors 8 cores View in monitoring Open Spark UI



Search

ENG
IN

7:18 PM
11/4/2024

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

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

Instance ID:6748147
XtremeLabs: DP-203T00-A-CEP [DP-203T00-A-M05-CEP] Module 05: Analyze data in a data lake with Spark

5. Modify the code to plot the chart as follows:
Paste Content
Paste Content

```
# Clear the plot area
plt.clf()

# Create a bar plot of revenue by year
plt.bar(x=df_sales['OrderYear'], height=df_s
ales['GrossRevenue'], color='orange')

# Customize the chart
plt.title('Revenue by Year')
plt.xlabel('Year')
plt.ylabel('Revenue')
plt.grid(color='#95a5a6', linestyle='--', li
newidth=2, axis='y', alpha=0.7)
plt.xticks(rotation=45)

# Show the figure
plt.show()
```

6. Re-run the code cell and view the results. The chart now includes a little more information.
A plot is technically contained with a **Figure**. In the previous examples, the figure was created implicitly for you; but you can create it explicitly.

7. Modify the code to plot the chart as follows:
Paste Content
Paste Content

Page: 11/13

Support

synapse8w7bzrq - Microsoft Az
synapse8w7bzrq - Azure Synaps

web.azuresynapse.net/en/authoring/explore/linked/notebooks/Notebook%201?workspace=%2Fsubscript...

Microsoft Azure | Synapse Analytics ▸ synapse8w7bzrq | Search | XLab-OcW-933@xtremelabs.us LAB DIRECTORY

Synapse live | Validate all | Publish all 1

files | Notebook 1

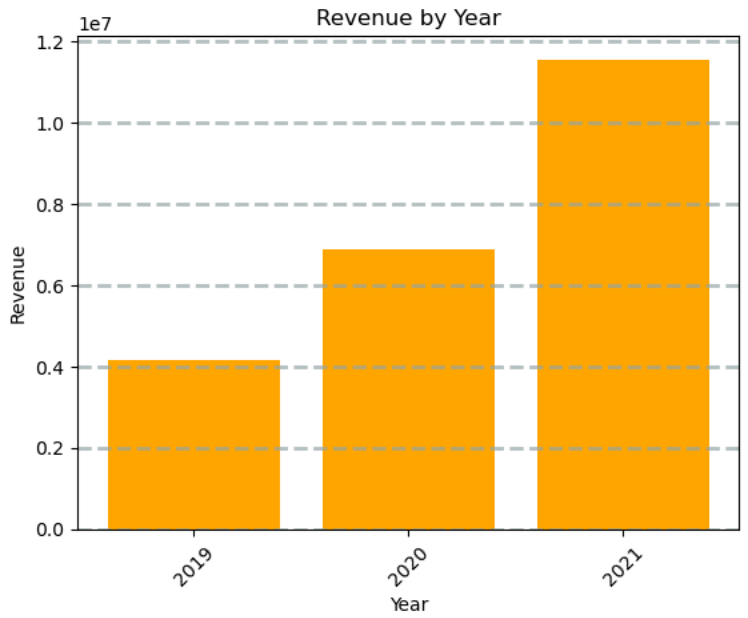
Other users in your workspace may have access to modify this item. Do not use this item unless you trust all users who may have access to the workspace.

Run all | Undo | Publish | Outline | Attach to spark8w7bzrq | Language PySpark (Python) | Variables

Ready

```
8  plt.title('Revenue by Year')
9  plt.xlabel('Year')
10 plt.ylabel('Revenue')
11 plt.grid(color='#95a5a6', linestyle='--', linewidth=2, axis='y', alpha=0.7)
12 plt.xticks(rotation=45)
13
14 # Show the figure
15 plt.show()
```

[14] <1 sec - Command executed in 552 ms by XLab-OcW-933 on 7:18:59 PM, 11/04/24



Search | ENG IN | 7:19 PM 11/4/2024

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

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

Instance ID:6748147

XtremeLabs: DP-203T00-A-CEP [DP-203T00-A-M05-CEP] Module 05: Analyze data in a data lake with Spark

```
# Create a pie chart of yearly order counts
on the second axis
yearly_counts = df_sales['OrderYear'].value_
counts()
ax[1].pie(yearly_counts)
ax[1].set_title('Orders per Year')
ax[1].legend(yearly_counts.keys().tolist())

# Add a title to the Figure
fig.suptitle('Sales Data')

# Show the figure
plt.show()
```

10. Re-run the code cell and view the results. The figure contains the subplots specified in the code.

Note: To learn more about plotting with matplotlib, see the matplotlib documentation .

Page: 11/13

Support

synapse8w7bzrq - Microsoft Az

synapse8w7bzrq - Azure Synaps

web.azuresynapse.net/en/authoring/explore/linked/notebooks/Notebook%201?workspace=%2Fsubscript...

Microsoft Azure | Synapse Analytics ▸ synapse8w7bzrq

Search

XLab-OcW-933@xtremelabs.us
LAB DIRECTORY

Synapse live | Validate all | Publish all 1

files | Notebook 1

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.

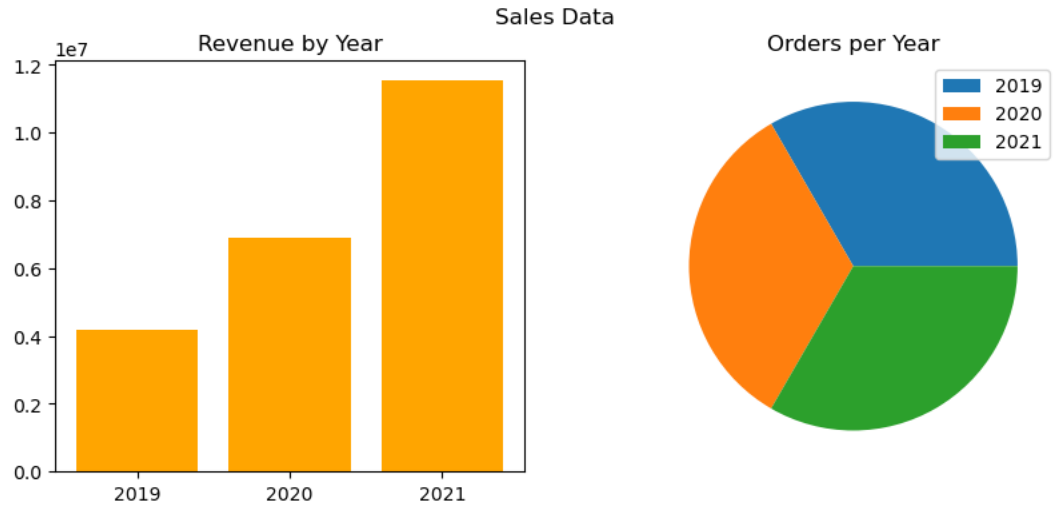
Run all | Undo | Publish | Outline | Attach to spark8w7bzrq | Language PySpark (Python) | Variables

Ready

```
7  # Create a bar plot of revenue by year on the first axis
8  ax[0].bar(x=df_sales['OrderYear'], height=df_sales['GrossRevenue'], color='orange')
9  ax[0].set_title('Revenue by Year')
10
11 # Create a pie chart of yearly order counts on the second axis
12 yearly_counts = df_sales['OrderYear'].value_counts()
13 ax[1].pie(yearly_counts)
14 ax[1].set_title('Orders per Year')
15 ax[1].legend(yearly_counts.keys().tolist())
16
17 # Add a title to the Figure
18 fig.suptitle('Sales Data')
19
20 # Show the figure
21 plt.show()
```

[16] ✓ <1 sec - Command executed in 548 ms by XLab-OcW-933 on 7:20:17 PM, 11/04/24

```
<Figure size 640x480 with 0 Axes>
```



Search

ENG
IN

7:20 PM
11/4/2024

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

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

Instance ID:6748147

XtremeLabs: DP-203T00-A-CEP [DP-203T00-A-M05-CEP] Module 05: Analyze data in a data lake with Spark

1. Add a new code cell to the notebook, and enter the following code in it:

Paste Content

Paste Content

```
import seaborn as sns

# Clear the plot area
plt.clf()

# Create a bar chart
ax = sns.barplot(x="OrderYear", y="GrossReve
nue", data=df_sales)
plt.show()
```

2. Run the code and observe that it displays a bar chart using the seaborn library.

3. Add a new code cell to the notebook, and enter the following code in it:

Paste Content

Paste Content

```
# Clear the plot area
plt.clf()

# Set the visual theme for seaborn
sns.set_theme(style="whitegrid")

# Create a bar chart
ax = sns.barplot(x="OrderYear", y="GrossReve
```

Page: 12/13

Support

synapse8w7bzrq - Microsoft Az

synapse8w7bzrq - Azure Synaps

web.azuresynapse.net/en/authoring/explore/linked/notebooks/Notebook%201?workspace=%2Fsubscript...

Microsoft Azure | Synapse Analytics ▸ synapse8w7bzrq

Search

XLab-OcW-933@xtremelabs.us
LAB DIRECTORY

Synapse live | Validate all | Publish all 1

files | Notebook 1

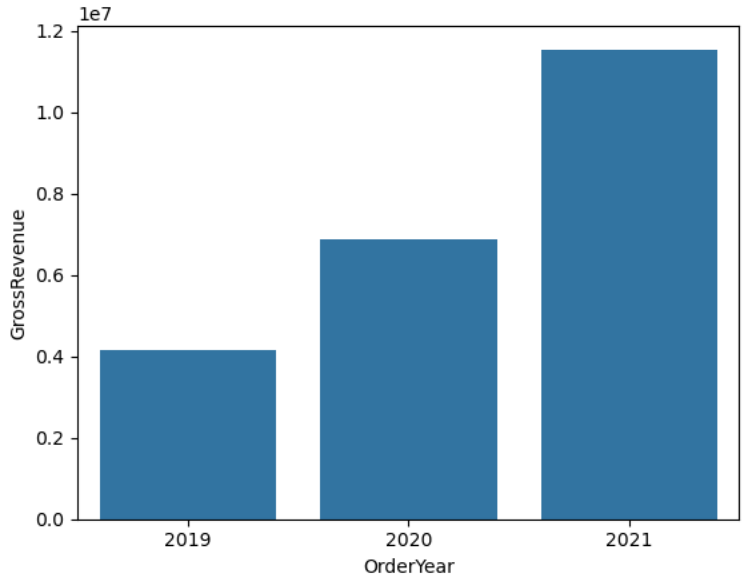
Other users in your workspace may have access to modify this item. Do not use this item unless you trust all users who may have access to the workspace.

Run all | Undo | Publish | Outline | Attach to spark8w7bzrq | Language PySpark (Python) | Variables

Ready

```
1  import seaborn as sns
2
3  # Clear the plot area
4  plt.clf()
5
6  # Create a bar chart
7  ax = sns.barplot(x="OrderYear", y="GrossRevenue", data=df_sales)
8  plt.show()
```

[17] 7 sec - Command executed in 6 sec 875 ms by XLab-OcW-933 on 7:21:11 PM, 11/04/24



Search

ENG
IN

7:21 PM
11/4/2024

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

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

Instance ID:6748147

XtremeLabs: DP-203T00-A-CEP [DP-203T00-A-M05-CEP] Module 05: Analyze data in a data lake with Spark

3. Add a new code cell to the notebook, and enter the following code in it:

Paste Content

Paste Content

```
# Clear the plot area
plt.clf()

# Set the visual theme for seaborn
sns.set_theme(style="whitegrid")

# Create a bar chart
ax = sns.barplot(x="OrderYear", y="GrossReve
nue", data=df_sales)
plt.show()
```

4. Run the code and note that seaborn enables you to set a consistent color theme for your plots.

5. Add a new code cell to the notebook, and enter the following code in it:

Paste Content

Paste Content

```
# Clear the plot area
plt.clf()

# Create a bar chart
ax = sns.lineplot(x="OrderYear", y="GrossRev
enue", data=df_sales)
plt.show()
```


Support

synapse8w7bzrq - Microsoft Az

synapse8w7bzrq - Azure Synaps

web.azuresynapse.net/en/authoring/explore/linked/notebooks/Notebook%201?workspace=%2Fsubscript...

Microsoft Azure | Synapse Analytics ▸ synapse8w7bzrq | Search | XLab-OcW-933@xtremelabs.us LAB DIRECTORY

Synapse live | Validate all | Publish all 1

files | Notebook 1

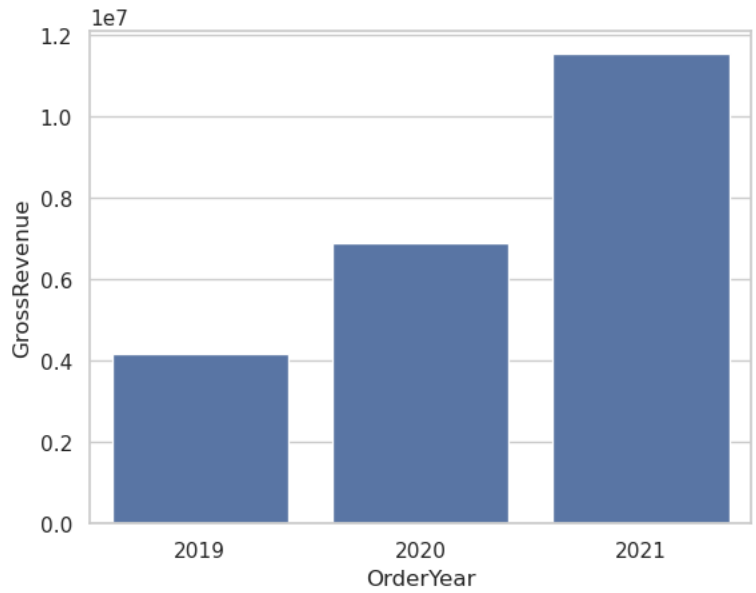
Other users in your workspace may have access to modify this item. Do not use this item unless you trust all users who may have access to the workspace.

Run all | Undo | Publish | Outline | Attach to spark8w7bzrq | Language PySpark (Python) | Variables

Ready

```
1  # Clear the plot area
2  plt.clf()
3
4  # Set the visual theme for seaborn
5  sns.set_theme(style="whitegrid")
6
7  # Create a bar chart
8  ax = sns.barplot(x="OrderYear", y="GrossRevenue", data=df_sales)
9  plt.show()
```

[19] <1 sec - Command executed in 521 ms by XLab-OcW-933 on 7:21:52 PM, 11/04/24



Search

ENG IN

7:21 PM 11/4/2024

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

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

Instance ID:6748147

XtremeLabs: DP-203T00-A-CEP [DP-203T00-A-M05-CEP] Module 05: Analyze data in a data lake with Spark

```
# Create a bar chart
ax = sns.barplot(x="OrderYear", y="GrossReve
nue", data=df_sales)
plt.show()
```

4. Run the code and note that seaborn enables you to set a consistent color theme for your plots.

5. Add a new code cell to the notebook, and enter the following code in it:

Paste Content

Paste Content

```
# Clear the plot area
plt.clf()

# Create a bar chart
ax = sns.lineplot(x="OrderYear", y="GrossRev
enue", data=df_sales)
plt.show()
```

6. Run the code to view the yearly revenue as a line chart.

Note: To learn more about plotting with seaborn, see the seaborn documentation .

Support

synapse8w7bzrq - Microsoft Az

synapse8w7bzrq - Azure Synaps

web.azuresynapse.net/en/authoring/explore/linked/notebooks/Notebook%201?workspace=%2Fsubscript...

Microsoft Azure | Synapse Analytics ▸ synapse8w7bzrq | Search | XLab-OcW-933@xtremelabs.us LAB DIRECTORY

Synapse live | Validate all | Publish all 1

files | Notebook 1

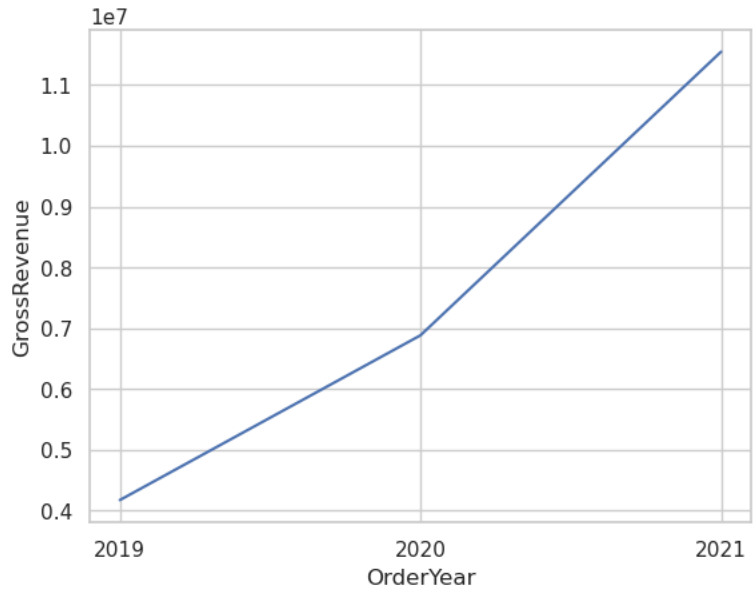
Other users in your workspace may have access to modify this item. Do not use this item unless you trust all users who may have access to the workspace.

Run all | Undo | Publish | Outline | Attach to spark8w7bzrq | Language PySpark (Python) | Variables

Ready

```
1  # Clear the plot area
2  plt.clf()
3
4  # Create a bar chart
5  ax = sns.lineplot(x="OrderYear", y="GrossRevenue", data=df_sales)
6  plt.show()
```

[20] <1 sec - Command executed in 553 ms by XLab-OcW-933 on 7:22:36 PM, 11/04/24



Search

ENG IN

7:22 PM 11/4/2024

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

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

Instance ID:6748147

XtremeLabs: DP-203T00-A-CEP [DP-203T00-A-M05-CEP] Module 05: Analyze data in a data lake with Spark

97% Completed

Lab Time Left: 01:09:03

Lab Actions

13. Module 5: Analyze data in a data lake with Spark:

Delete Azure resources:

- [x] If you've finished exploring Azure Synapse Analytics, you should delete the resources you've created to avoid unnecessary Azure costs.
- [x] 1. Close the Synapse Studio browser tab and return to the Azure portal.
- [x] 2. On the Azure portal, on the **Home** page, select **Resource groups**.
- [x] 3. Select the **dp000-*xxxxxxx*** resource group for your Synapse Analytics workspace (not the managed resource group), and verify that it contains the Synapse workspace, storage account, and Spark pool for your workspace.
- [ ] 4. At the top of the **Overview** page for your resource group, select **Delete resource group**.
- [ ] 5. Enter the **dp000-*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.

Page: 13/13

Support

dp000-8w7bzrq - Microsoft Azu

portal.azure.com/#@xtremelabs.onmicrosoft.com/resource/subscriptions/ce5e3ccc-35fe-4387-b1b6-56c9...

Microsoft Azure

Search resources, services, and docs (G+/)

Copilot

Home > Resource groups >

Deleting resource group dp000-8w7bzrq

Deleting resource group dp000-8w7bzrq

dp000-8w7bzrq

Resource group

Search

Overview

Activity log

Access control (IAM)

Tags

Resource visualizer

Events

Settings

Cost Management

Monitoring

Automation

Help

Create | Manage view | Delete resource group | Refresh | Export to CSV | Open query

Essentials

JSON View

Resources | Recommendations

Filter for any field... | Type equals **all** | Location equals **all** | Add filter

Showing 1 to 3 of 3 records. | Show hidden types | No grouping

List view

Name	Type	Location
datalake8w7bzrq	Storage account	West US
spark8w7bzrq (synapse8w7bzrq/spark8w7bzrq)	Apache Spark pool	West US
synapse8w7bzrq	Synapse workspace	West US

< Page 1 of 1 >

Give feedback

Search

ENG IN

7:24 PM 11/4/2024

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

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

Instance ID:6748147

XtremeLabs: DP-203T00-A-CEP [DP-203T00-A-M05-CEP] Module 05: Analyze data in a data lake with Spark

100% Completed

Lab Time Left: 01:07:02

Lab Actions

13. Module 5: Analyze data in a data lake with Spark:

Delete Azure resources:

- [x] If you've finished exploring Azure Synapse Analytics, you should delete the resources you've created to avoid unnecessary Azure costs.
- [x] 1. Close the Synapse Studio browser tab and return to the Azure portal.
- [x] 2. On the Azure portal, on the **Home** page, select **Resource groups**.
- [x] 3. Select the **dp000-*xxxxxxx*** resource group for your Synapse Analytics workspace (not the managed resource group), and verify that it contains the Synapse workspace, storage account, and Spark pool for your workspace.
- [x] 4. At the top of the **Overview** page for your resource group, select **Delete resource group**.
- [x] 5. Enter the **dp000-*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.

Page: 13/13

Support

dp000-8w7bzrq - Microsoft Azu

portal.azure.com/#@xtremelabs.onmicrosoft.com/resource/subscriptions/ce5e3ccc-35fe-4387-b1b6-56c9...

Microsoft Azure | Search resources, services, and docs (G+/) | Copilot

Home > Resource groups >

dp000-8w7bzrq

Resource group

Deleted resource group dp000-8w7bzrq

Deleted resource group dp000-8w7bzrq

Search

- Overview
- Activity log
- Access control (IAM)
- Tags
- Resource visualizer
- Events
- Settings
- Cost Management
- Monitoring
- Automation
- Help

Create | Manage view | Delete resource group | Refresh | Export to CSV | Open query

Essentials | JSON View

Resources | Recommendations

Filter for any field... | Type equals **all** | Location equals **all** | Add filter

Showing 0 to 0 of 0 records. | Show hidden types | No grouping

List view

Name	Type	Location

No resources match your filters

Try changing or clearing your filters.

Create resources | Clear filters

Give feedback

Search

ENG IN

7:26 PM 11/4/2024

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

Welcome to XtremeLabs, Nitin Nitin NN

View Labs
Access Codes
History
Support
FAQs

Search for courses

DP-203T00-A-CEP
DP-900T00-A-CEP

Classroom Chat

Lab Title: [DP-203T00-A-M01-CEP] Module 01: Explore Azure Synapse Analytics
Duration: 240 minutes
Status : **Completed and Relaunched**
Take Lab

Lab Title: [DP-203T00-A-M02-CEP] Module 02: Query files using a serverless SQL pool
Duration: 120 minutes
Status : **Completed and Relaunched**
Take Lab

Lab Title: [DP-203T00-A-M03-CEP] Module 03: Transform data using a serverless SQL pool
Duration: 120 minutes
Status : **Completed**
Take Lab

Lab Title: [DP-203T00-A-M04-CEP] Module 04: Analyze data in a lake database
Duration: 120 minutes
Status : **Completed**
Take Lab

Lab Title: [DP-203T00-A-M05-CEP] Module 05: Analyze data in a data lake with Spark
Duration: 120 minutes
Status : **Completed and Relaunched**
Take Lab

Lab Title: [DP-203T00-A-M06-CEP] Module 06: Transform data using Spark in Synapse Analytics
Duration: 120 minutes
Status : Not Initiated
Take Lab

Lab Title: [DP-203T00-A-M07-CEP] Module

Lab Title: [DP-203T00-A-M08-CEP] Module