# Coding Challenge – 5

## Azure Synapse ETL Pipeline

**Name:** Aathirainathan P

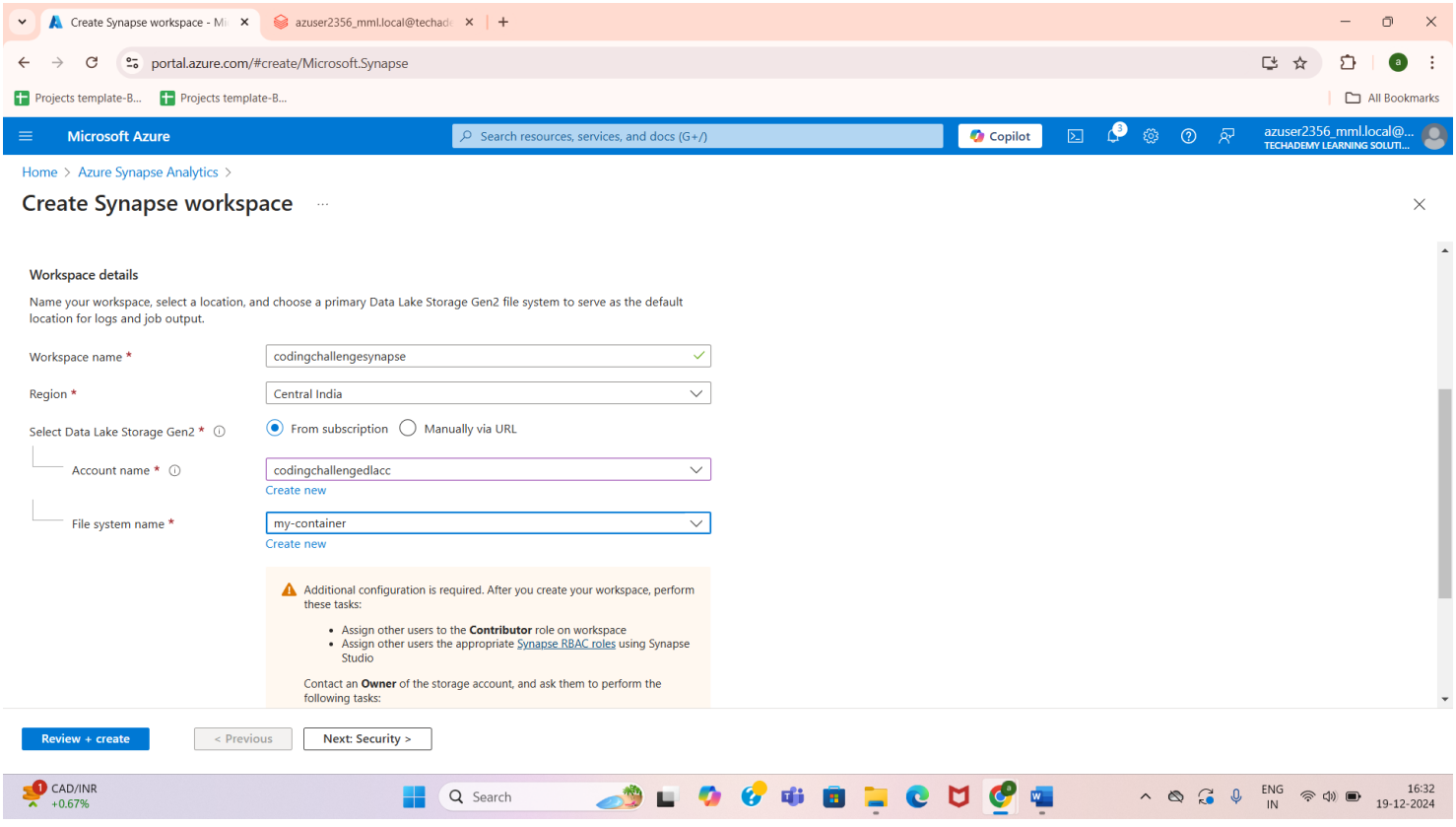
**Date:** 19-12-2024

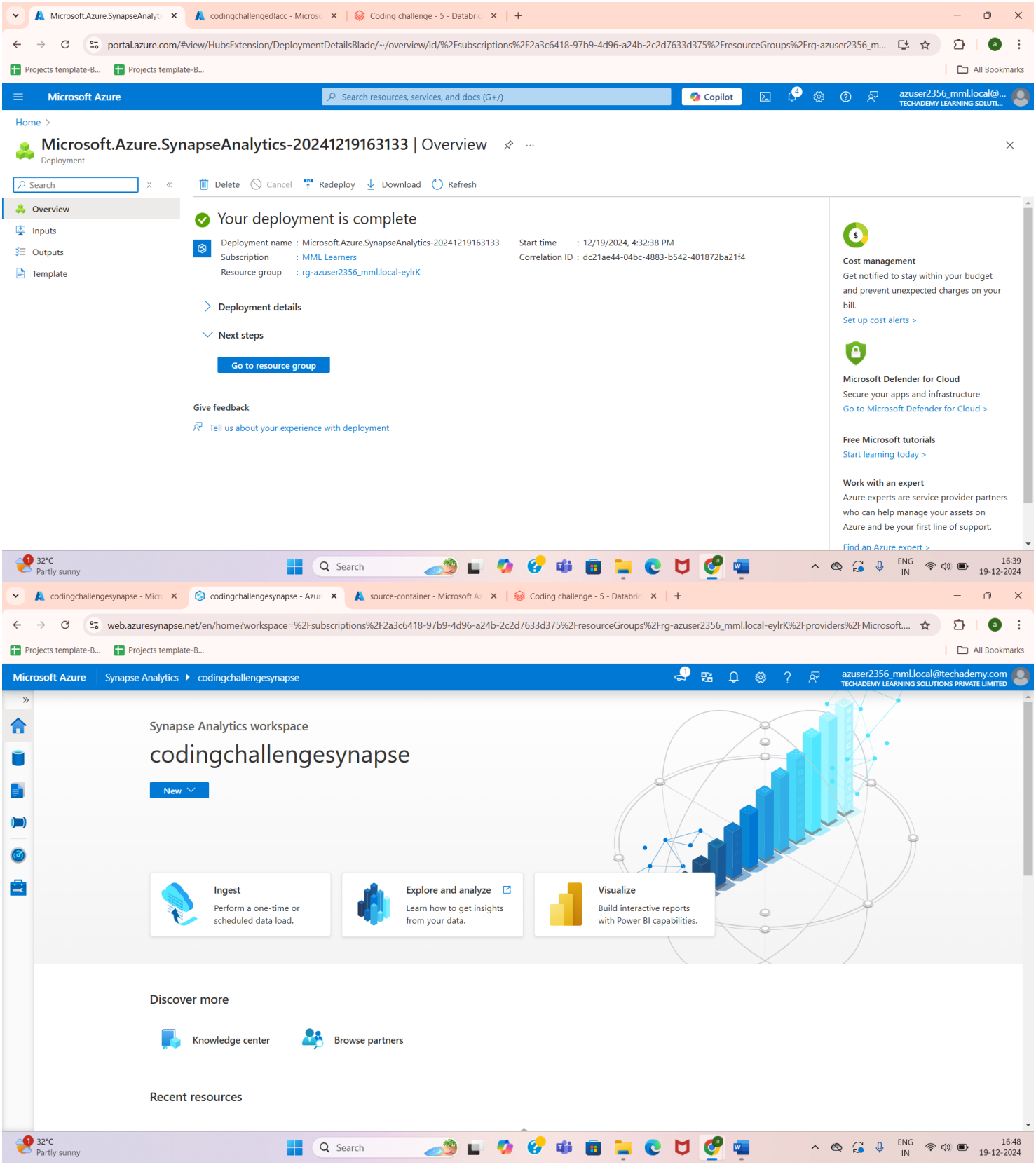
---

## 1) Build an ETL pipline with azure synapse with dataflow running on it.

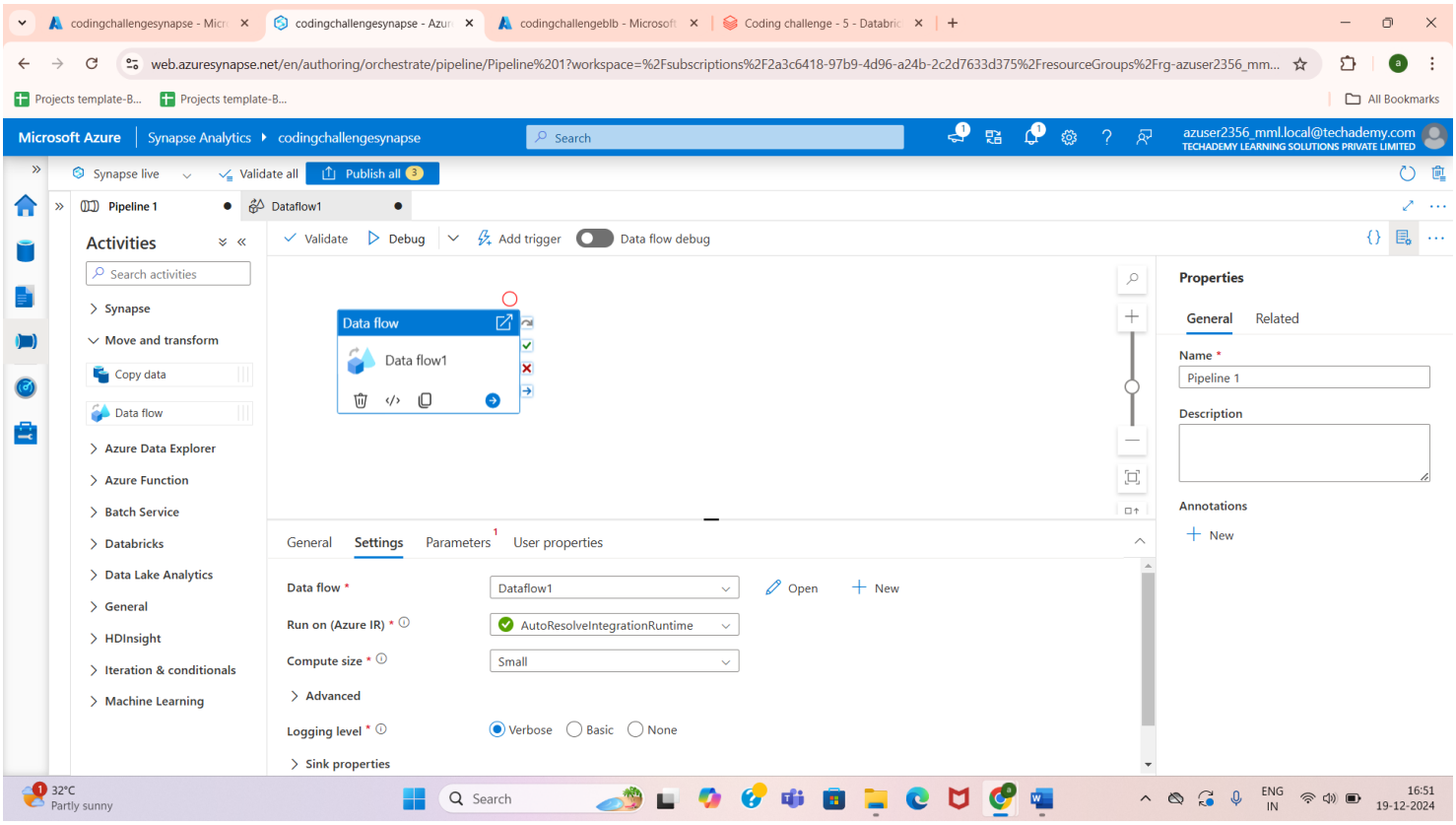
## Tasks Done With Steps:

### *Create an Azure Synapse Workspace:*

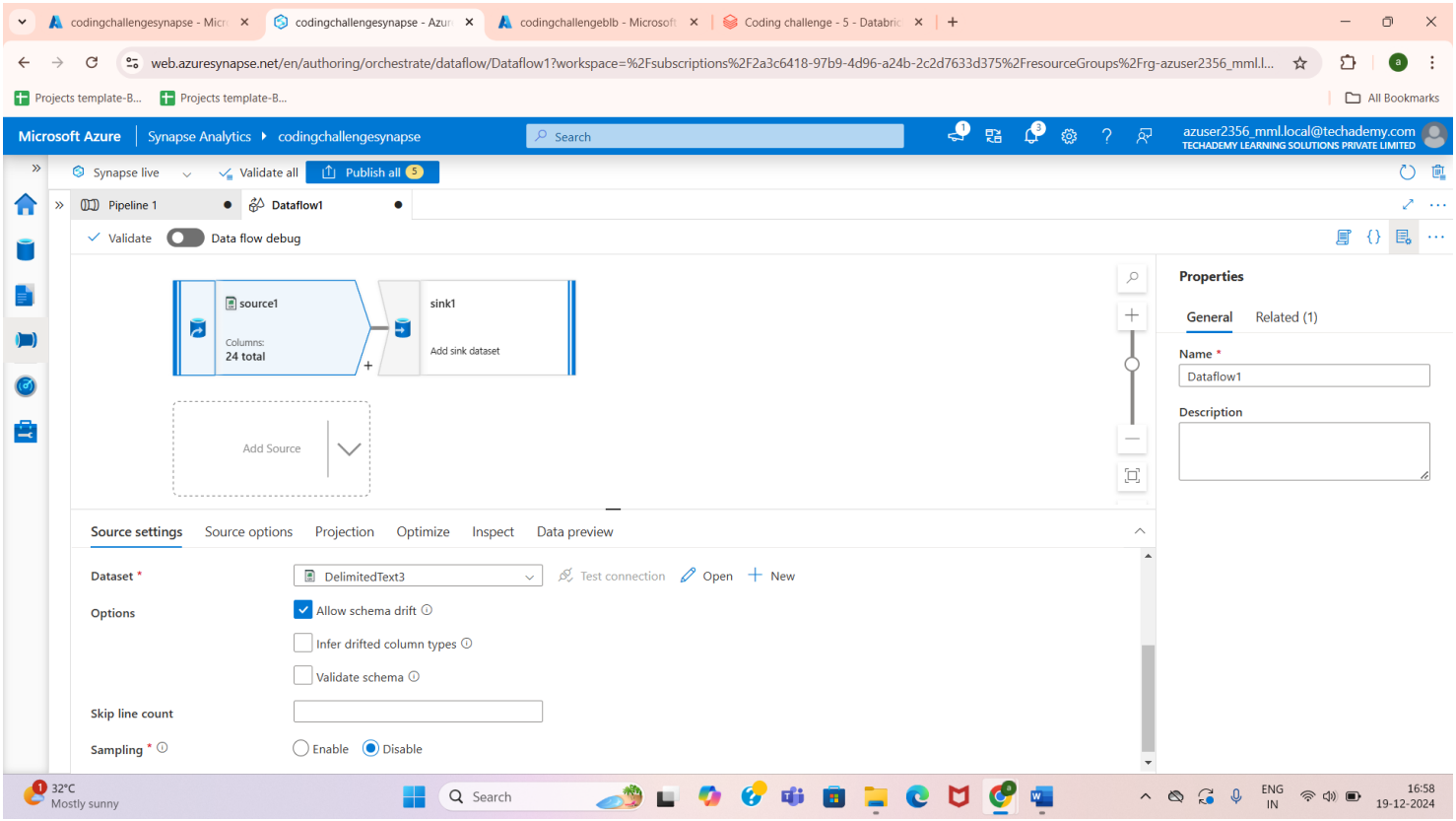


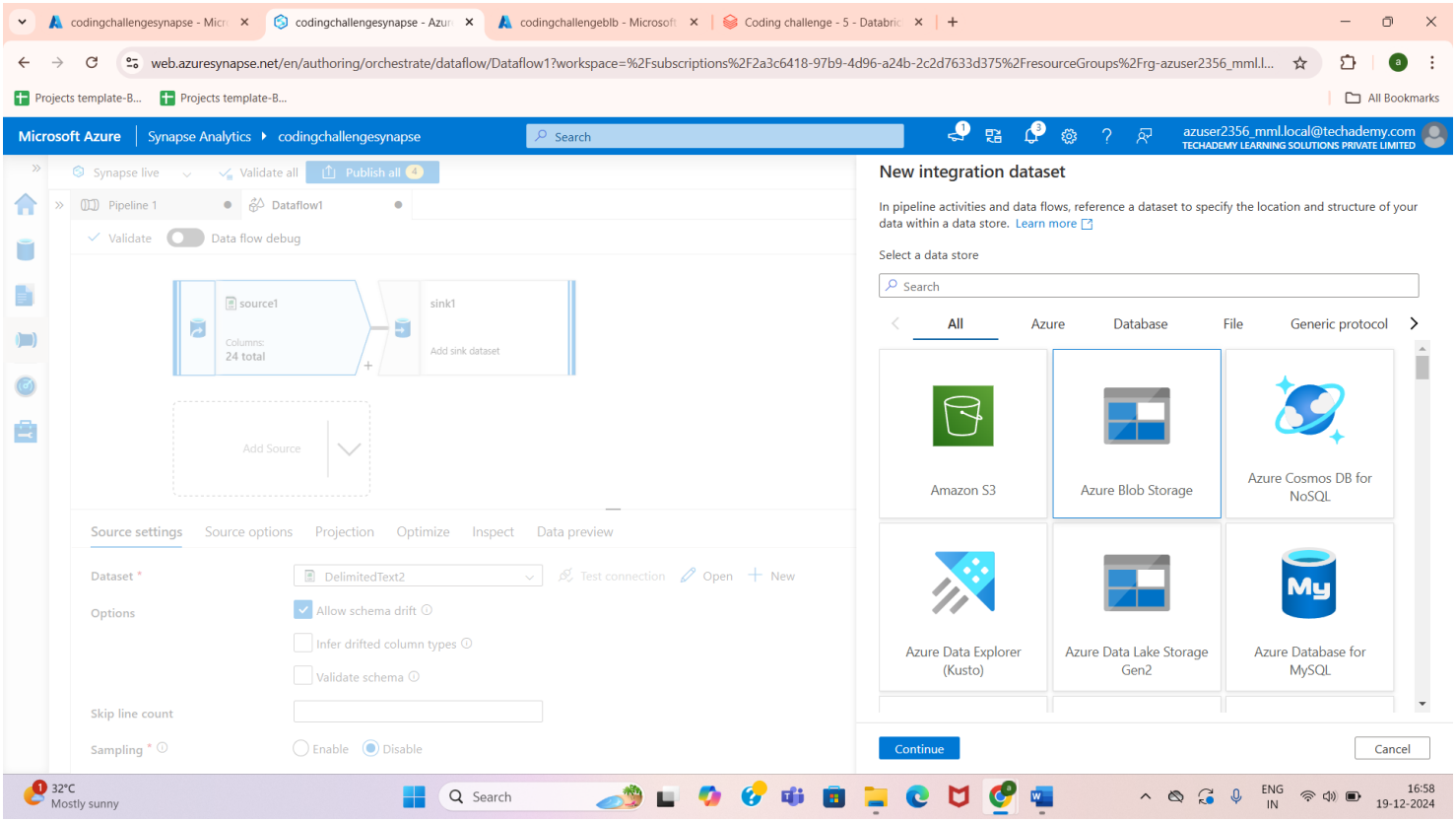

Microsoft.Azure.SynapseAnalyti
codingchallengedlacc - Micros
Coding challenge - 5 - Databric
portal.azure.com/#view/HubsExtension/DeploymentDetailsBlade/~/overview/id/%2Fsubscriptions%2F2a3c6418-97b9-4d96-a24b-2c2d7633d375%2FresourceGroups%2Frg-azuser2356_m...
Projects template-B...
Projects template-B...
All Bookmarks
Microsoft Azure
Search resources, services, and docs (G+/)
Copilot
azuser2356_mml.local@...
TECHADEMY LEARNING SOLUTI...
Home
Microsoft.Azure.SynapseAnalytics-20241219163133 | Overview
Deployment
Search
Overview
Inputs
Outputs
Template
Delete
Cancel
Redeploy
Download
Refresh
Your deployment is complete
Deployment name : Microsoft.Azure.SynapseAnalytics-20241219163133
Subscription : MML Learners
Resource group : rg-azuser2356_mml.local-eylrK
Start time : 12/19/2024, 4:32:38 PM
Correlation ID : dc21ae44-04bc-4883-b542-401872ba21f4
Deployment details
Next steps
Go to resource group
Give feedback
Tell us about your experience with deployment
Cost management
Get notified to stay within your budget and prevent unexpected charges on your bill.
Set up cost alerts >
Microsoft Defender for Cloud
Secure your apps and infrastructure
Go to Microsoft Defender for Cloud >
Free Microsoft tutorials
Start learning today >
Work with an expert
Azure experts are service provider partners who can help manage your assets on Azure and be your first line of support.
Find an Azure expert >
32°C
Partly sunny
Search
ENG
IN
16:39
19-12-2024
codingchallengesynapse - Micr
codingchallengesynapse - Azur
source-container - Microsoft A
Coding challenge - 5 - Databric
web.azuresynapse.net/en/home?workspace=%2Fsubscriptions%2F2a3c6418-97b9-4d96-a24b-2c2d7633d375%2FresourceGroups%2Frg-azuser2356_mml.local-eylrK%2Fproviders%2FMicrosoft....
Projects template-B...
Projects template-B...
All Bookmarks
Microsoft Azure
Synapse Analytics
codingchallengesynapse
azuser2356_mml.local@techademy.com
TECHADEMY LEARNING SOLUTIONS PRIVATE LIMITED
Synapse Analytics workspace
codingchallengesynapse
New
Ingest
Perform a one-time or scheduled data load.
Explore and analyze
Learn how to get insights from your data.
Visualize
Build interactive reports with Power BI capabilities.
Discover more
Knowledge center
Browse partners
Recent resources
32°C
Partly sunny
Search
ENG
IN
16:48
19-12-2024

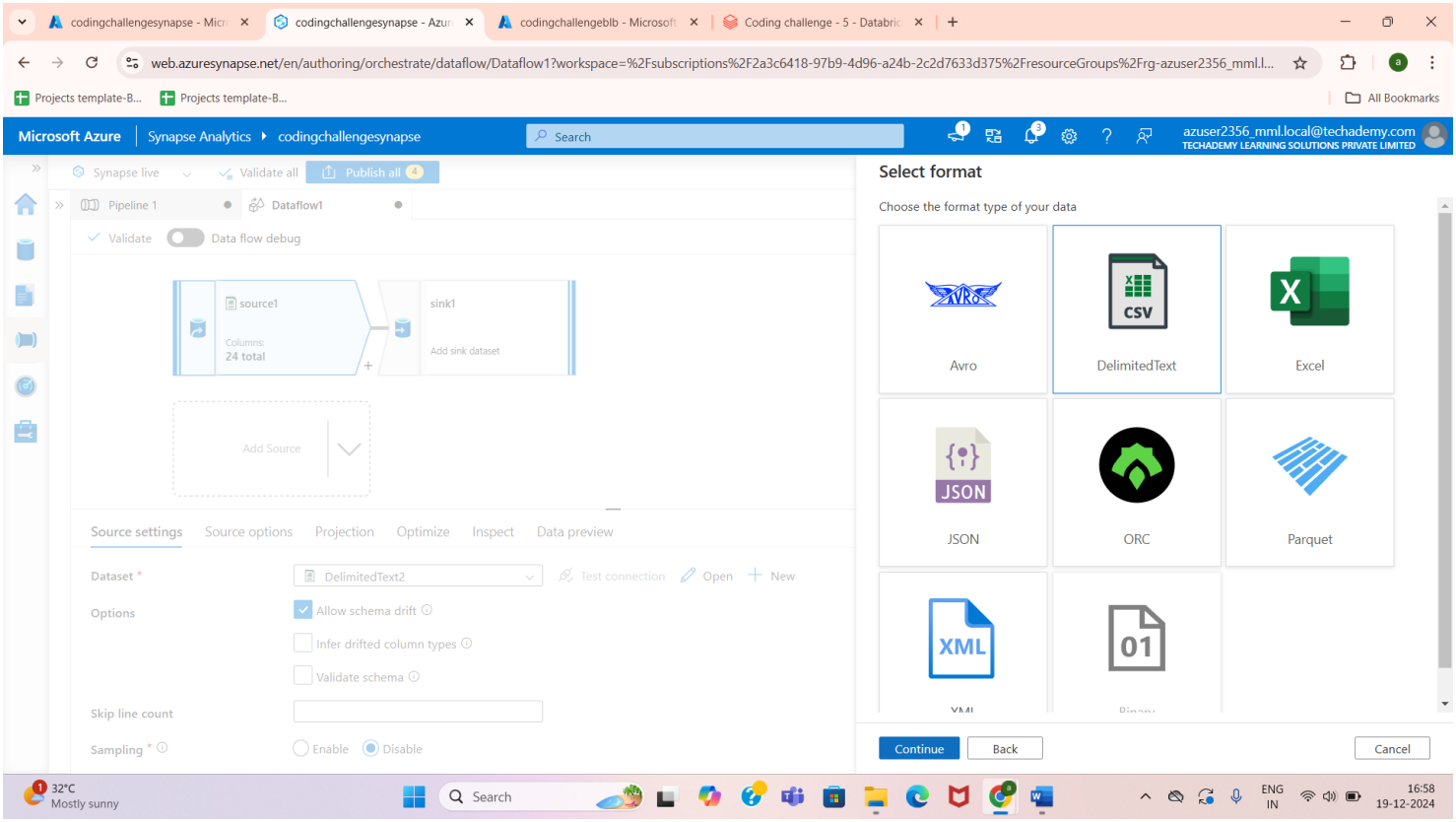
## *Create data flow activity:*

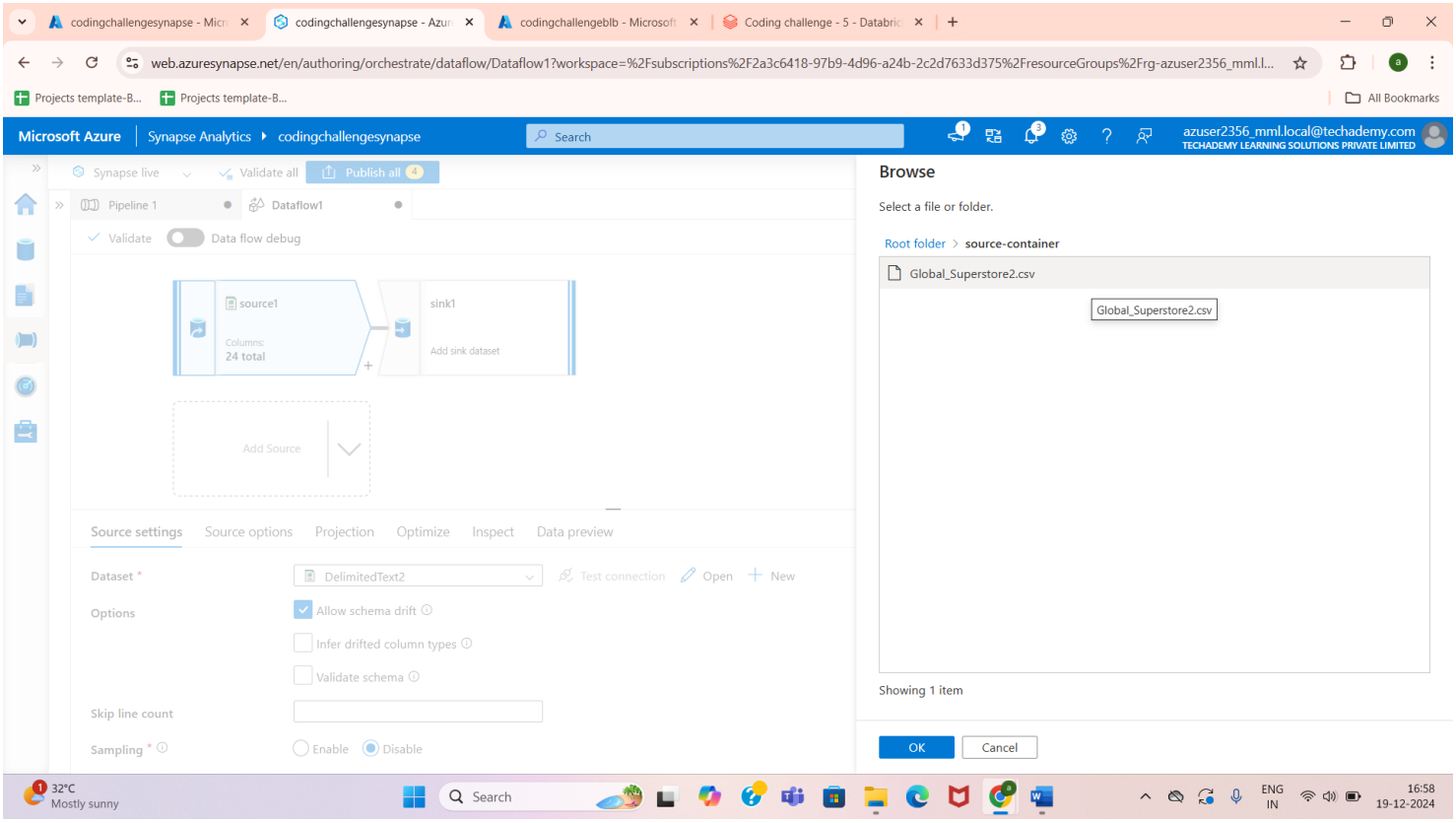



## *Create a source and sink and Configure the source:*

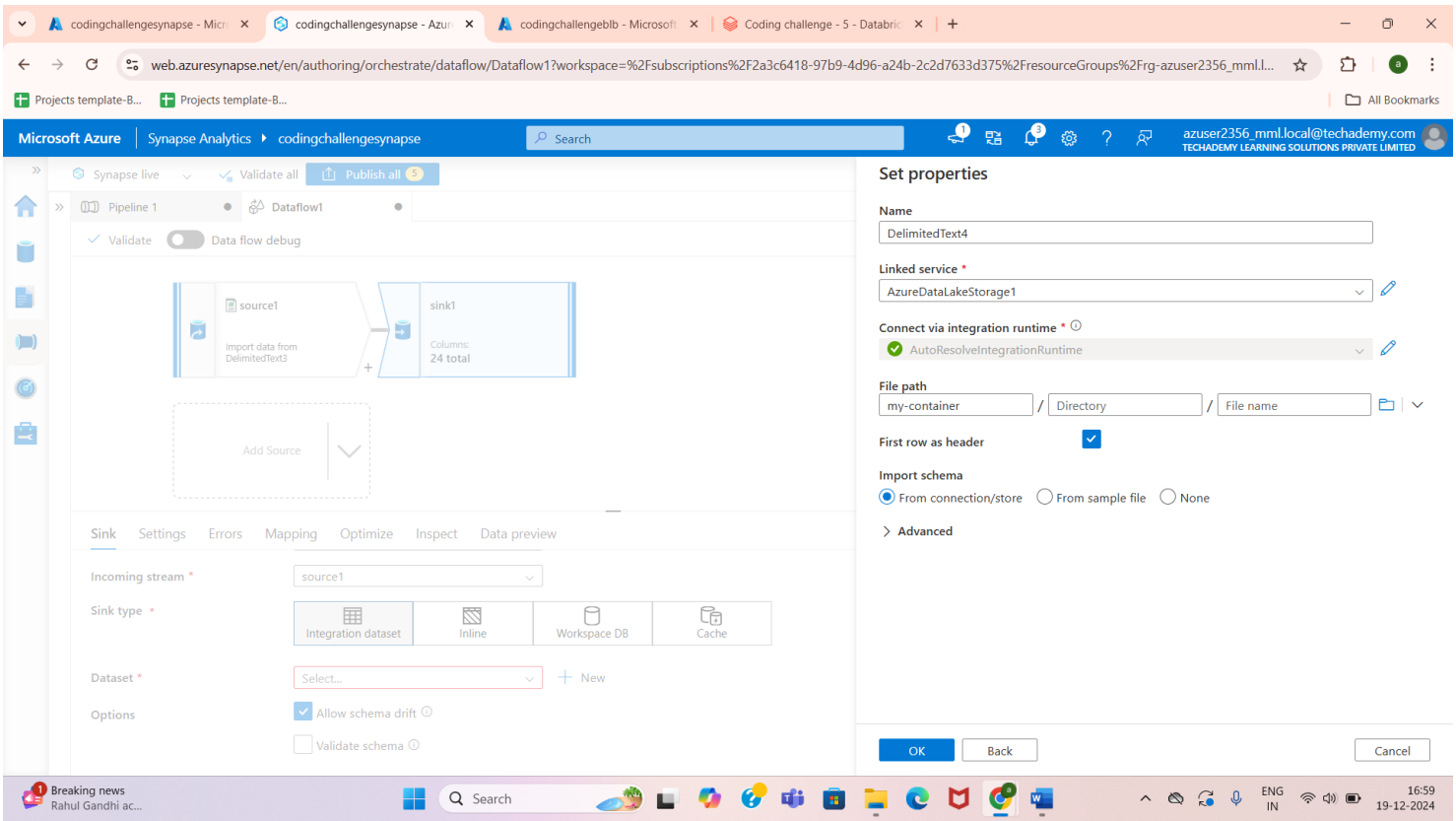



codingchallengesynapse - Micr
codingchallengesynapse - Azur
codingchallengeblb - Microsoft
Coding challenge - 5 - Databric
web.azuresynapse.net/en/authoring/orchestrate/dataflow/Dataflow1?workspace=%2Fsubscriptions%2F2a3c6418-97b9-4d96-a24b-2c2d7633d375%2FresourceGroups%2Frg-azuser2356_mml.l...
Projects template-B...
Projects template-B...
All Bookmarks
Microsoft Azure
Synapse Analytics
codingchallengesynapse
Search
azuser2356_mml.local@techademy.com
TECHADEMY LEARNING SOLUTIONS PRIVATE LIMITED
Synapse live
Validate all
Publish all 4
Pipeline 1
Dataflow1
Validate
Data flow debug
source1
Columns:
24 total
sink1
Add sink dataset
Add Source
Source settings
Source options
Projection
Optimize
Inspect
Data preview
Dataset *
DelimitedText2
Test connection
Open
New
Options
Allow schema drift
Infer drifted column types
Validate schema
Skip line count
Sampling *
Enable
Disable
New integration dataset
In pipeline activities and data flows, reference a dataset to specify the location and structure of your data within a data store. Learn more
Select a data store
Search
All
Azure
Database
File
Generic protocol
Amazon S3
Azure Blob Storage
Azure Cosmos DB for NoSQL
Azure Data Explorer (Kusto)
Azure Data Lake Storage Gen2
Azure Database for MySQL
Continue
Cancel
32°C
Mostly sunny
Search
ENG
IN
16:58
19-12-2024

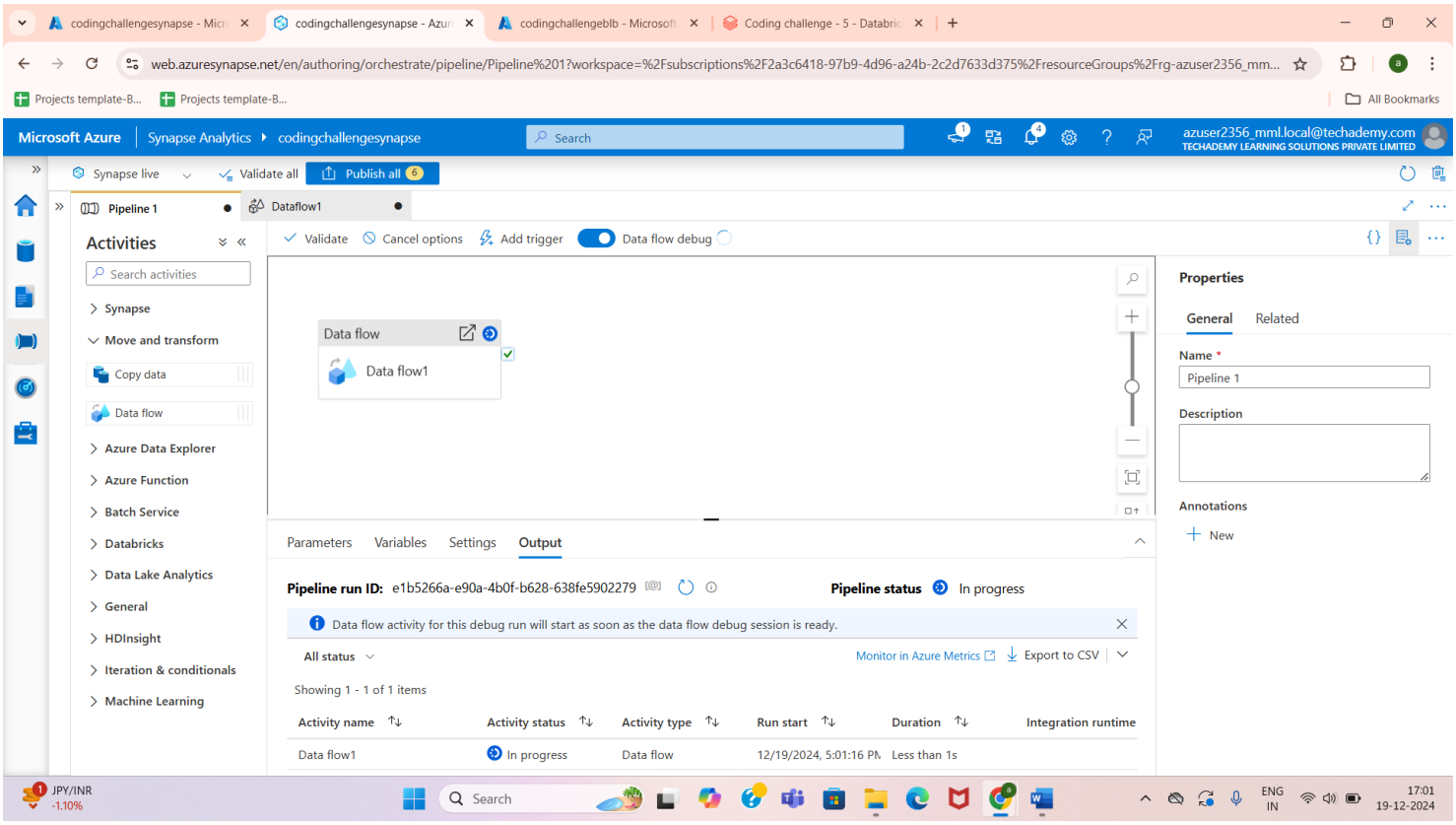

codingchallengesynapse - Micr
codingchallengesynapse - Azur
codingchallengeblb - Microsoft
Coding challenge - 5 - Databric
web.azuresynapse.net/en/authoring/orchestrate/dataflow/Dataflow1?workspace=%2Fsubscriptions%2F2a3c6418-97b9-4d96-a24b-2c2d7633d375%2FresourceGroups%2Frg-azuser2356_mml.l...
Projects template-B...
Projects template-B...
All Bookmarks
Microsoft Azure
Synapse Analytics
codingchallengesynapse
Search
azuser2356_mml.local@techademy.com
TECHADEMY LEARNING SOLUTIONS PRIVATE LIMITED
Synapse live
Validate all
Publish all 4
Pipeline 1
Dataflow1
Validate
Data flow debug
source1
Columns:
24 total
sink1
Add sink dataset
Add Source
Source settings
Source options
Projection
Optimize
Inspect
Data preview
Dataset *
DelimitedText2
Test connection
Open
New
Options
Allow schema drift
Infer drifted column types
Validate schema
Skip line count
Sampling *
Enable
Disable
Select format
Choose the format type of your data
Avro
DelimitedText
Excel
JSON
ORC
Parquet
XML
Continue
Back
Cancel
32°C
Mostly sunny
Search
ENG
IN
16:58
19-12-2024

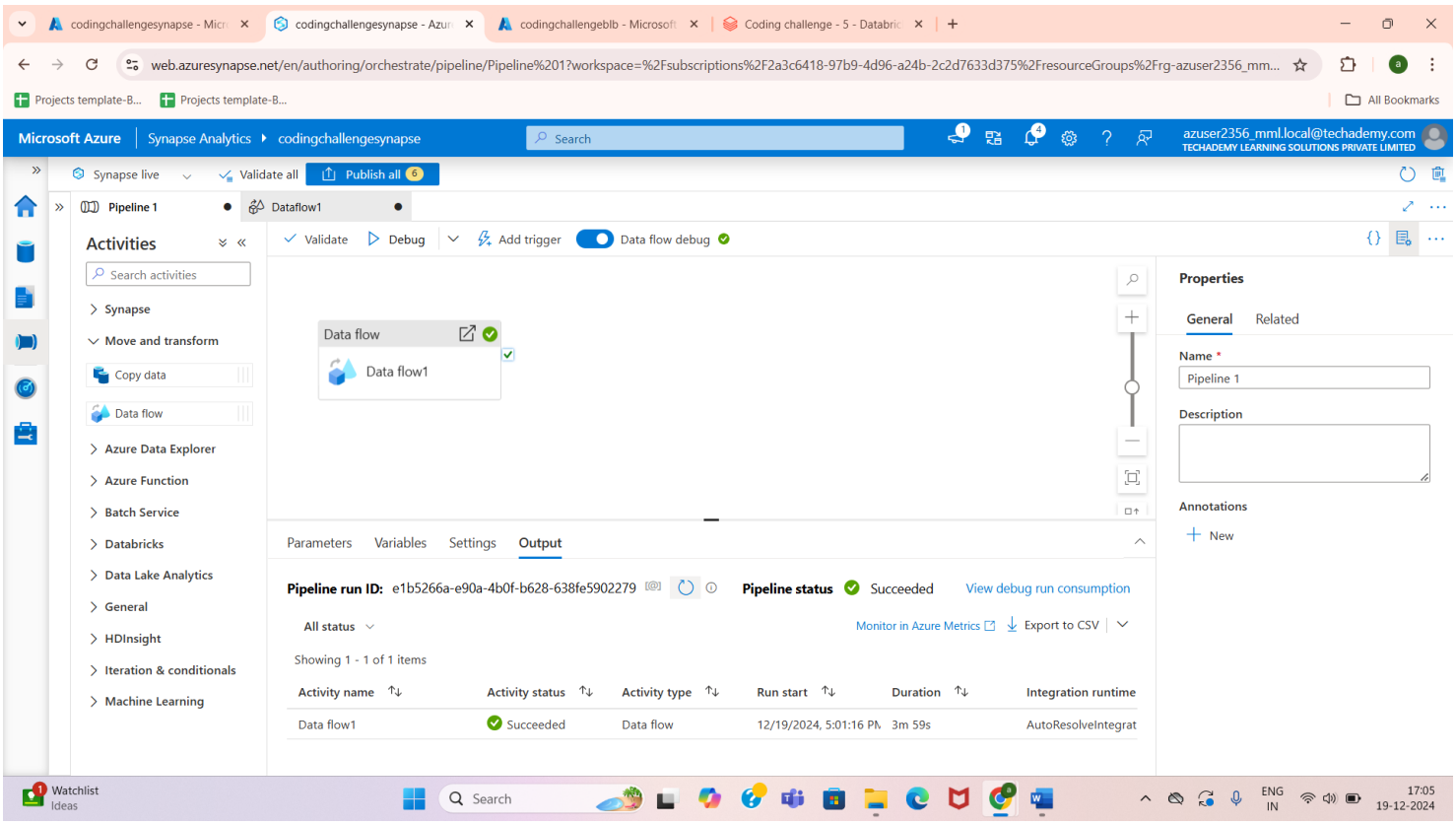



## *Configure the sink:*



## *Validate and Debug the pipeline:*



## *Extract the data from the delta lake storage and do the transformations and loading:*

04:44 PM (15s) 1 Python

```
%python
# Fetching the csv file from the blob storage
storage_account_name = "codingchallengedlacc"
container_name = "my-container"
storage_account_key = "hMW9uKwa01lepiUgz0tGQd9P9UKbeVq29mCY5wCz2GIPpsMELvdjGaAcJMmDjsiX8RHCznioSigW+AStcpQIgQ=="

# Unmount the directory if it is already mounted
if any(mount.mountPoint == "/mnt/superstore" for mount in dbutils.fs.mounts()):
    dbutils.fs.unmount("/mnt/superstore")

# Mount dl Storage
dbutils.fs.mount(
    source=f"wasbs://{container_name}@{storage_account_name}.blob.core.windows.net",
    mount_point="/mnt/superstore",
    extra_configs={
        f"fs.azure.account.key.{storage_account_name}.blob.core.windows.net": storage_account_key
    }
)
```

```
True
```

## *Verifying the mount:*

04:44 PM (6s) 2

```
#verifying the mount
display(dbutils.fs.ls("/mnt/superstore"))
```

(2) Spark Jobs

Table

| | path | name | size | modificationTime |
|---|---|---|---|---|
| 1 | dbfs:/mnt/superstore/Global_Superstore2.c... | Global_Superstore2.c... | 12089916 | 1734606845000 |

1 row | 5.79 seconds runtime

Refreshed 41 minutes ago

## *Load the dataset:*

04:45 PM (6s) 3 Python

```
from pyspark.sql import SparkSession
spark = SparkSession.builder.appName("SuperstoreETL").getOrCreate()

# Load the dataset
file_path = "/mnt/superstore/Global_Superstore2.csv"
df = spark.read.csv(file_path, header=True, inferSchema=True)
df.show(5)
```

(3) Spark Jobs

df: pyspark.sql.dataframe.DataFrame = [Row ID: integer, Order ID: string ... 22 more fields]

```
---------+
| 32298| CA-2012-124891|2012-07-31|2012-07-31|    Same Day|   RH-19495|     Rick Hansen|   Consumer|New York City|       New York|United States|
10024|    US|   East| TEC-AC-10003033|Technology| Accessories|Plantronics CS510...| 2309.65|       7|       0|762.1845|       933.57|      Criti
cal|
| 26341|  IN-2013-77878|2013-02-05|2013-02-07|Second Class|   JR-16210|   Justin Ritter|  Corporate|   Wollongong|New South Wales|    Australia|
NULL|  APAC|Oceania| FUR-CH-10003950| Furniture|      Chairs|Novimex Executive...|3709.395|       9|     0.1|-288.765|       923.63|      Critic
al|
| 25330|  IN-2013-71249|2013-10-17|2013-10-18| First Class|   CR-12730|    Craig Reiter|   Consumer|     Brisbane|     Queensland|    Australia|
NULL|  APAC|Oceania| TEC-PH-10004664|Technology|      Phones|Nokia Smart Phone...|5175.171|       9|     0.1| 919.971|       915.49|        Medi
um|
| 13524|ES-2013-1579342|2013-01-28|2013-01-30| First Class|   KM-16375|Katherine Murray|Home Office|       Berlin|         Berlin|      Germany|
NULL|    EU|Central| TEC-PH-10004583|Technology|      Phones|Motorola Smart Ph...| 2892.51|       5|     0.1|  -96.54|       910.16|        Medi
um|
| 47221|   SG-2013-4320|2013-11-05|2013-11-06|    Same Day|    RH-9495|     Rick Hansen|   Consumer|        Dakar|          Dakar|      Senegal|
NULL|Africa| Africa|TEC-SHA-10000501|Technology|     Copiers|Sharp Wireless Fa...| 2832.96|       8|       0|  311.52|       903.04|      Critic
```

## *Add Total Cost Column:*

04:45 PM (1s) 4

```
#Add Total Cost Column
from pyspark.sql.functions import col

df_transformed = df.withColumn("TotalCost", col("Sales") - col("Profit"))
df.show(5)
```

(1) Spark Jobs

df_transformed: pyspark.sql.dataframe.DataFrame = [Row ID: integer, Order ID: string ... 23 more fields]

```
---------+
| 32298| CA-2012-124891|2012-07-31|2012-07-31|    Same Day|   RH-19495|     Rick Hansen|   Consumer|New York City|       New York|United States|
10024|    US|   East| TEC-AC-10003033|Technology| Accessories|Plantronics CS510...| 2309.65|       7|       0|762.1845|       933.57|      Criti
cal|
| 26341|  IN-2013-77878|2013-02-05|2013-02-07|Second Class|   JR-16210|   Justin Ritter|  Corporate|   Wollongong|New South Wales|    Australia|
NULL|  APAC|Oceania| FUR-CH-10003950| Furniture|      Chairs|Novimex Executive...|3709.395|       9|     0.1|-288.765|       923.63|      Critic
al|
| 25330|  IN-2013-71249|2013-10-17|2013-10-18| First Class|   CR-12730|    Craig Reiter|   Consumer|     Brisbane|     Queensland|    Australia|
NULL|  APAC|Oceania| TEC-PH-10004664|Technology|      Phones|Nokia Smart Phone...|5175.171|       9|     0.1| 919.971|       915.49|        Medi
um|
| 13524|ES-2013-1579342|2013-01-28|2013-01-30| First Class|   KM-16375|Katherine Murray|Home Office|       Berlin|         Berlin|      Germany|
NULL|    EU|Central| TEC-PH-10004583|Technology|      Phones|Motorola Smart Ph...| 2892.51|       5|     0.1|  -96.54|       910.16|        Medi
um|
| 47221|   SG-2013-4320|2013-11-05|2013-11-06|    Same Day|    RH-9495|     Rick Hansen|   Consumer|        Dakar|          Dakar|      Senegal|
NULL|Africa| Africa|TEC-SHA-10000501|Technology|     Copiers|Sharp Wireless Fa...| 2832.96|       8|       0|  311.52|       903.04|      Critic
al|
+------+---------------+----------+----------+------------+-----------+----------------+-----------+-------------+---------------+-------------+
-----------+------+-------+----------------+----------+------------+--------------------+--------+--------+--------+--------+-------------+-----
---------+
```

## *Convert Sales and Profit to Float:*

04:45 PM (1s) 5 Python

```
#Convert Sales and Profit to Float

df_transformed = df_transformed.withColumn("Sales", col("Sales").cast("float"))
df_transformed = df_transformed.withColumn("Profit", col("Profit").cast("float"))

df_transformed.show(5)
```

(1) Spark Jobs

df_transformed: pyspark.sql.dataframe.DataFrame = [Row ID: integer, Order ID: string ... 23 more fields]

```
+------+---------------+----------+----------+------------+-----------+----------------+-----------+-------------+---------------+-------------+
-----------+------+-------+----------------+----------+------------+--------------------+--------+--------+--------+--------+-------------+-----
---------+------------------+
|Row ID|       Order ID|Order Date| Ship Date|   Ship Mode|Customer ID|   Customer Name|    Segment|         City|          State|      Country|
Postal Code|Market| Region|      Product ID|  Category|Sub-Category|        Product Name|   Sales|Quantity|Discount|  Profit|Shipping Cost|Order
Priority|         TotalCost|
+------+---------------+----------+----------+------------+-----------+----------------+-----------+-------------+---------------+-------------+
-----------+------+-------+----------------+----------+------------+--------------------+--------+--------+--------+--------+-------------+-----
---------+------------------+
| 32298| CA-2012-124891|2012-07-31|2012-07-31|    Same Day|   RH-19495|     Rick Hansen|   Consumer|New York City|       New York|United States|
10024|    US|   East| TEC-AC-10003033|Technology| Accessories|Plantronics CS510...| 2309.65|       7|       0|762.1845|       933.57|      Criti
cal|1547.4655000000002|
| 26341|  IN-2013-77878|2013-02-05|2013-02-07|Second Class|   JR-16210|   Justin Ritter|  Corporate|   Wollongong|New South Wales|    Australia|
NULL|  APAC|Oceania| FUR-CH-10003950| Furniture|      Chairs|Novimex Executive...|3709.395|       9|     0.1|-288.765|       923.63|      Critic
```

## *Filter Rows with Zero or Negative Profit:*

04:45 PM (1s) 6 Python

```
#Filter Rows with Zero or Negative Profit

df_transformed = df_transformed.filter(col("Profit") > 0)
df_transformed.show(5)
```

(1) Spark Jobs

df_transformed: pyspark.sql.dataframe.DataFrame = [Row ID: integer, Order ID: string ... 23 more fields]

```
---+------------------+
| 32298|CA-2012-124891|2012-07-31|2012-07-31|    Same Day|   RH-19495|  Rick Hansen| Consumer|New York City|       New York|United States|
10024|    US|    East| TEC-AC-10003033|Technology| Accessories|Plantronics CS510...| 2309.65|       7|       0|762.1845|       933.57|      Criti
cal|1547.4655000000002|
| 25330| IN-2013-71249|2013-10-17|2013-10-18| First Class|   CR-12730| Craig Reiter| Consumer|     Brisbane|     Queensland|    Australia|
NULL|  APAC|Oceania| TEC-PH-10004664|Technology|      Phones|Nokia Smart Phone...|5175.171|       9|     0.1| 919.971|       915.49|        Medi
um| 4255.200000000001|
| 47221|  SG-2013-4320|2013-11-05|2013-11-06|    Same Day|    RH-9495|  Rick Hansen| Consumer|        Dakar|          Dakar|      Senegal|
NULL|Africa| Africa|TEC-SHA-10000501|Technology|     Copiers|Sharp Wireless Fa...| 2832.96|       8|       0|  311.52|       903.04|      Critic
al|           2521.44|
| 22732| IN-2013-42360|2013-06-28|2013-07-01|Second Class|   JM-15655|  Jim Mitchum|Corporate|       Sydney|New South Wales|    Australia|
NULL|  APAC|Oceania| TEC-PH-10000030|Technology|      Phones|Samsung Smart Pho...|2862.675|       5|     0.1| 763.275|       897.35|      Critic
al|            2099.4|
| 30570| IN-2011-81826|2011-11-07|2011-11-09| First Class|   TS-21340|Toby Swindell| Consumer|      Porirua|     Wellington|  New Zealand|
NULL|  APAC|Oceania| FUR-CH-10004050| Furniture|      Chairs|Novimex Executive...| 1822.08|       4|       0|  564.84|       894.77|      Critic
al|1257.2399999999998|
+------+--------------+----------+----------+------------+-----------+-------------+---------+-------------+---------------+-------------+------
------+------+-------+----------------+----------+------------+--------------------+--------+--------+--------+--------+-------------+-----------
---+------------------+
```

```
# Step 5: Show the transformed data
df_transformed.show(5)
```

(2) Spark Jobs

df_clean: pyspark.sql.dataframe.DataFrame = [Row ID: integer, Order ID: string ... 22 more fields]

df_transformed: pyspark.sql.dataframe.DataFrame = [Category: string, Region: string ... 4 more fields]

```
+----------+------+------------------+------------------+-------------+-----------------+
|  Category|Region|        TotalSales|       TotalProfit|TotalQuantity|    TotalDiscount|
+----------+------+------------------+------------------+-------------+-----------------+
| Furniture|Canada|10595.279964447021|           2613.24|           78|              0.0|
|Technology|  EMEA|  300854.583026886|17494.443000000036|         2259|189.1000056862831|
| Furniture|  East| 205540.3473367691|         2501.8162|         2151| 90.6000018119812|
|Technology|Africa| 322367.0430994034|  44129.4930000001|         2031|143.1999975964427|
|Technology|  East| 264872.0816922188| 47439.55759999996|         1927|76.30000080168247|
+----------+------+------------------+------------------+-------------+-----------------+
only showing top 5 rows
```

## *Add a Profit Margin Column:*

04:45 PM (2s) 8 Python

```
from pyspark.sql.functions import col

# Add a Profit Margin Column
df_transformed = df_transformed.withColumn("ProfitMargin", (col("TotalProfit") / col("TotalSales")) * 100)

# Show the result
df_transformed.show(5)
```

(2) Spark Jobs

df_transformed: pyspark.sql.dataframe.DataFrame = [Category: string, Region: string ... 5 more fields]

```
+----------+------+------------------+------------------+-------------+-----------------+------------------+
|  Category|Region|        TotalSales|       TotalProfit|TotalQuantity|    TotalDiscount|      ProfitMargin|
+----------+------+------------------+------------------+-------------+-----------------+------------------+
| Furniture|Canada|10595.279964447021|           2613.24|           78|              0.0|24.664190174953884|
|Technology|  EMEA| 300854.583026886|17494.443000000036|         2259|189.1000056862831| 5.814916569988444|
| Furniture|  East| 205540.3473367691|         2501.8162|         2151| 90.6000018119812|1.2171898278934408|
|Technology|Africa| 322367.0430994034|  44129.4930000001|         2031|143.1999975964427|13.689207363046899|
|Technology|  East| 264872.0816922188| 47439.55759999996|         1927|76.30000080168247|17.910365372189244|
+----------+------+------------------+------------------+-------------+-----------------+------------------+
only showing top 5 rows
```

## *Remove Duplicate Rows based on 'Category' and 'Region':*

04:45 PM (1s) 9 Python

```
# Remove Duplicate Rows based on 'Category' and 'Region'
df_transformed = df_transformed.dropDuplicates(["Category", "Region"])

df_transformed.show(5)
```

(2) Spark Jobs

df_transformed: pyspark.sql.dataframe.DataFrame = [Category: string, Region: string ... 5 more fields]

```
+----------+------+------------------+------------------+-------------+-----------------+------------------+
|  Category|Region|        TotalSales|       TotalProfit|TotalQuantity|    TotalDiscount|      ProfitMargin|
+----------+------+------------------+------------------+-------------+-----------------+------------------+
| Furniture|Canada|10595.279964447021|           2613.24|           78|              0.0|24.664190174953884|
|Technology|  EMEA| 300854.583026886|17494.443000000036|         2259|189.1000056862831| 5.814916569988444|
| Furniture|  East| 205540.3473367691|         2501.8162|         2151| 90.6000018119812|1.2171898278934408|
|Technology|Africa| 322367.0430994034|  44129.4930000001|         2031|143.1999975964427|13.689207363046899|
|Technology|  East| 264872.0816922188| 47439.55759999996|         1927|76.30000080168247|17.910365372189244|
+----------+------+------------------+------------------+-------------+-----------------+------------------+
only showing top 5 rows
```

## *Rename Columns for Clarity:*

04:45 PM (1s) 10 Python

```
#Rename Columns for Clarity
df_transformed = df_transformed.withColumnRenamed("Sales", "TotalSales") \
                               .withColumnRenamed("Profit", "TotalProfit")

df_transformed.show(5)
```

(2) Spark Jobs

df_transformed: pyspark.sql.dataframe.DataFrame = [Category: string, Region: string ... 5 more fields]

```
+----------+------+------------------+------------------+-------------+-----------------+------------------+
|  Category|Region|        TotalSales|       TotalProfit|TotalQuantity|    TotalDiscount|      ProfitMargin|
+----------+------+------------------+------------------+-------------+-----------------+------------------+
| Furniture|Canada|10595.279964447021|           2613.24|           78|              0.0|24.664190174953884|
|Technology|  EMEA| 300854.583026886|17494.443000000036|         2259|189.1000056862831| 5.814916569988444|
| Furniture|  East| 205540.3473367691|         2501.8162|         2151| 90.6000018119812|1.2171898278934408|
|Technology|Africa| 322367.0430994034|  44129.4930000001|         2031|143.1999975964427|13.689207363046899|
|Technology|  East| 264872.0816922188| 47439.55759999996|         1927|76.30000080168247|17.910365372189244|
+----------+------+------------------+------------------+-------------+-----------------+------------------+
only showing top 5 rows
```

## *Add a Year Column:*

04:45 PM (1s) 11 Python

```
from pyspark.sql.functions import col, year, to_date

# Add a Year Column
df = df.withColumn("Year", year(to_date(col("Order Date"), "MM/dd/yyyy")))

# Show the result
df_transformed.show(5)
```

(2) Spark Jobs

df: pyspark.sql.dataframe.DataFrame = [Row ID: integer, Order ID: string ... 23 more fields]

```
+----------+------+------------------+------------------+-------------+-----------------+------------------+
|  Category|Region|        TotalSales|       TotalProfit|TotalQuantity|    TotalDiscount|      ProfitMargin|
+----------+------+------------------+------------------+-------------+-----------------+------------------+
| Furniture|Canada|10595.279964447021|           2613.24|           78|              0.0|24.664190174953884|
|Technology|  EMEA| 300854.583026886|17494.443000000036|         2259|189.1000056862831| 5.814916569988444|
| Furniture|  East| 205540.3473367691|         2501.8162|         2151| 90.6000018119812|1.2171898278934408|
|Technology|Africa| 322367.0430994034|  44129.4930000001|         2031|143.1999975964427|13.689207363046899|
|Technology|  East| 264872.0816922188| 47439.55759999996|         1927|76.30000080168247|17.910365372189244|
+----------+------+------------------+------------------+-------------+-----------------+------------------+
only showing top 5 rows
```

## *Filter the Data for a Specific Year:*

04:45 PM (2s) 12 Python

```
from pyspark.sql.functions import col, year, to_date, sum
#Filter the Data for a Specific Year

# Step 1: Add the Year Column to the Original DataFrame
df = df.withColumn("Year", year(to_date(col("Order Date"), "MM/dd/yyyy")))

# Step 2: Perform the aggregation including 'Year'
df_transformed = df.groupBy("Category", "Region", "Year") \
    .agg(
        sum("Sales").alias("TotalSales"),
        sum("Profit").alias("TotalProfit"),
        sum("Quantity").alias("TotalQuantity"),
        sum("Discount").alias("TotalDiscount")
    )

# Step 3: Filter the Data for a Specific Year
df_transformed = df_transformed.filter(col("Year") == 2012)

# Show the result
df_transformed.show(5)
```

(2) Spark Jobs

df: pyspark.sql.dataframe.DataFrame = [Row ID: integer, Order ID: string ... 23 more fields]

df_transformed: pyspark.sql.dataframe.DataFrame = [Category: string, Region: string ... 5 more fields]

```
+----------+-------+----+------------------+------------------+-------------+------------------+
|  Category| Region|Year|        TotalSales|       TotalProfit|TotalQuantity|     TotalDiscount|
+----------+-------+----+------------------+------------------+-------------+------------------+
| Furniture|Oceania|2012|100519.00800000002|          8623.818|        607.0| 26.19999999999998|
|Technology|  North|2012|126353.56300000004|25098.862999999998|        809.0|             7.378|
|Technology| Africa|2012|64734.582000000024| 6320.742000000001|        404.0| 36.69999999999999|
| Furniture| Canada|2012|           1600.68|            290.19|         16.0|               0.0|
|Technology|Oceania|2012| 89761.76700000005|         14203.827|        688.0|23.499999999999986|
+----------+-------+----+------------------+------------------+-------------+------------------+
only showing top 5 rows
```

## *Sort Data by Total Sales:*

04:46 PM (1s) 13 Python

```
#Sort Data by Total Sales
df_transformed = df_transformed.orderBy(col("TotalSales").desc())
df_transformed.show()
```

(2) Spark Jobs

df_transformed: pyspark.sql.dataframe.DataFrame = [Category: string, Region: string ... 5 more fields]

| Category | Region | Year | TotalSales | TotalProfit | TotalQuantity | TotalDiscount |
|---|---|---|---|---|---|---|
| Technology | Central | 2012 | 237291.81162000002 | 31373.854319999966 | 1791.0 | 61.14199999999993 |
| Furniture | Central | 2012 | 183778.99949999998 | 5231.0848000000005 | 1876.94 | 85.14000000000004 |
| Office Supplies | Central | 2012 | 180068.29200000002 | 26988.422600000013 | 6003.084 | 198.4000000000001 |
| Technology | North | 2012 | 126353.56300000004 | 25098.862999999998 | 809.0 | 7.378 |
| Office Supplies | South | 2012 | 116962.67200000002 | 10905.974199999997 | 3831.186 | 151.70000000000016 |
| Furniture | South | 2012 | 106962.4015 | 8656.693399999998 | 1108.0 | 38.94999999999999 |
| Technology | South | 2012 | 103154.04796 | 11673.361959999993 | 1034.824 | 38.896 |
| Furniture | Oceania | 2012 | 100519.00800000002 | 8623.818 | 607.0 | 26.19999999999998 |
| Office Supplies | North | 2012 | 91426.151 | 17629.370999999996 | 2648.0 | 36.899999999999984 |
| Technology | Oceania | 2012 | 89761.76700000005 | 14203.827 | 688.0 | 23.499999999999986 |
| Furniture | North | 2012 | 82451.25800000002 | 8177.708000000001 | 839.0 | 50.49999999999996 |
| Technology | Southeast Asia | 2012 | 77886.98879999996 | 6068.458799999999 | 534.0 | 32.09000000000001 |
| Technology | Central Asia | 2012 | 69458.76000000001 | 13756.859999999999 | 394.0 | 5.5 |
| Technology | North Asia | 2012 | 64934.62500000001 | 13026.855000000001 | 359.0 | 4.0 |
| Technology | Africa | 2012 | 64734.582000000024 | 6320.742000000001 | 404.0 | 36.69999999999999 |

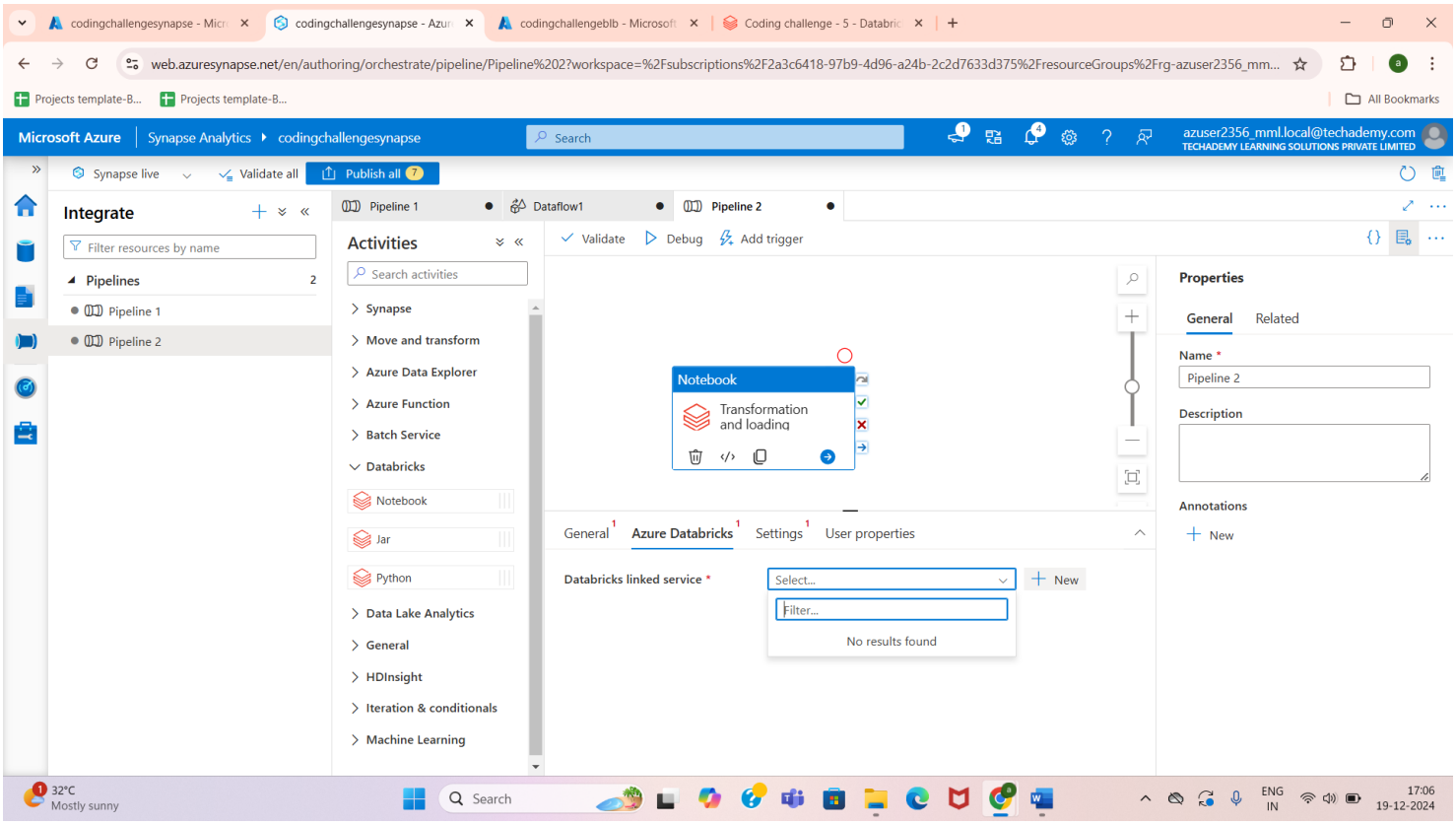
## *Save the transformed data in DBFS:*

04:46 PM (14s) 14 Python

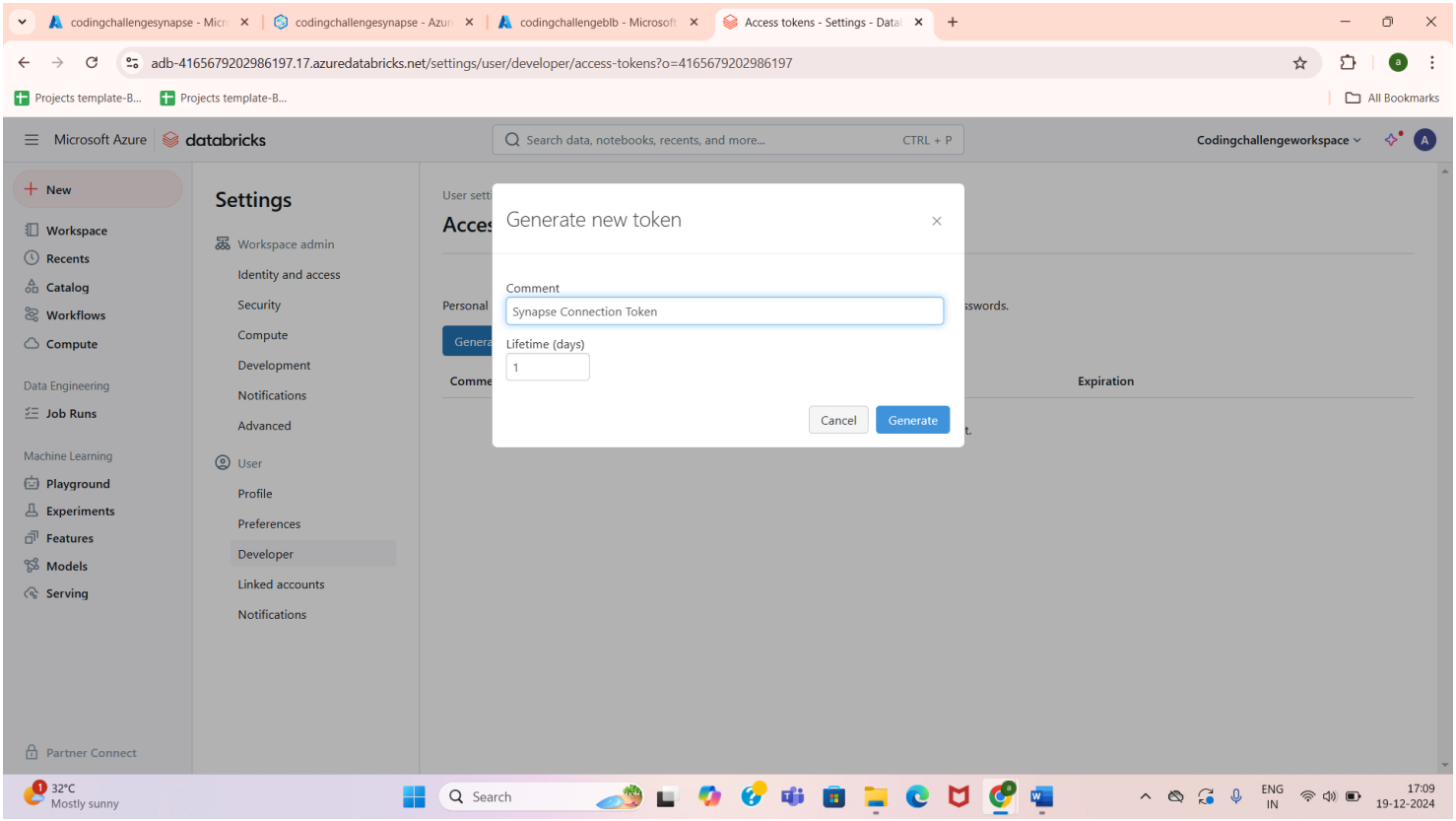
```
# Save the transformed data in DBFS
df_transformed.write.format("delta").mode("overwrite").saveAsTable("transformed_data")
```

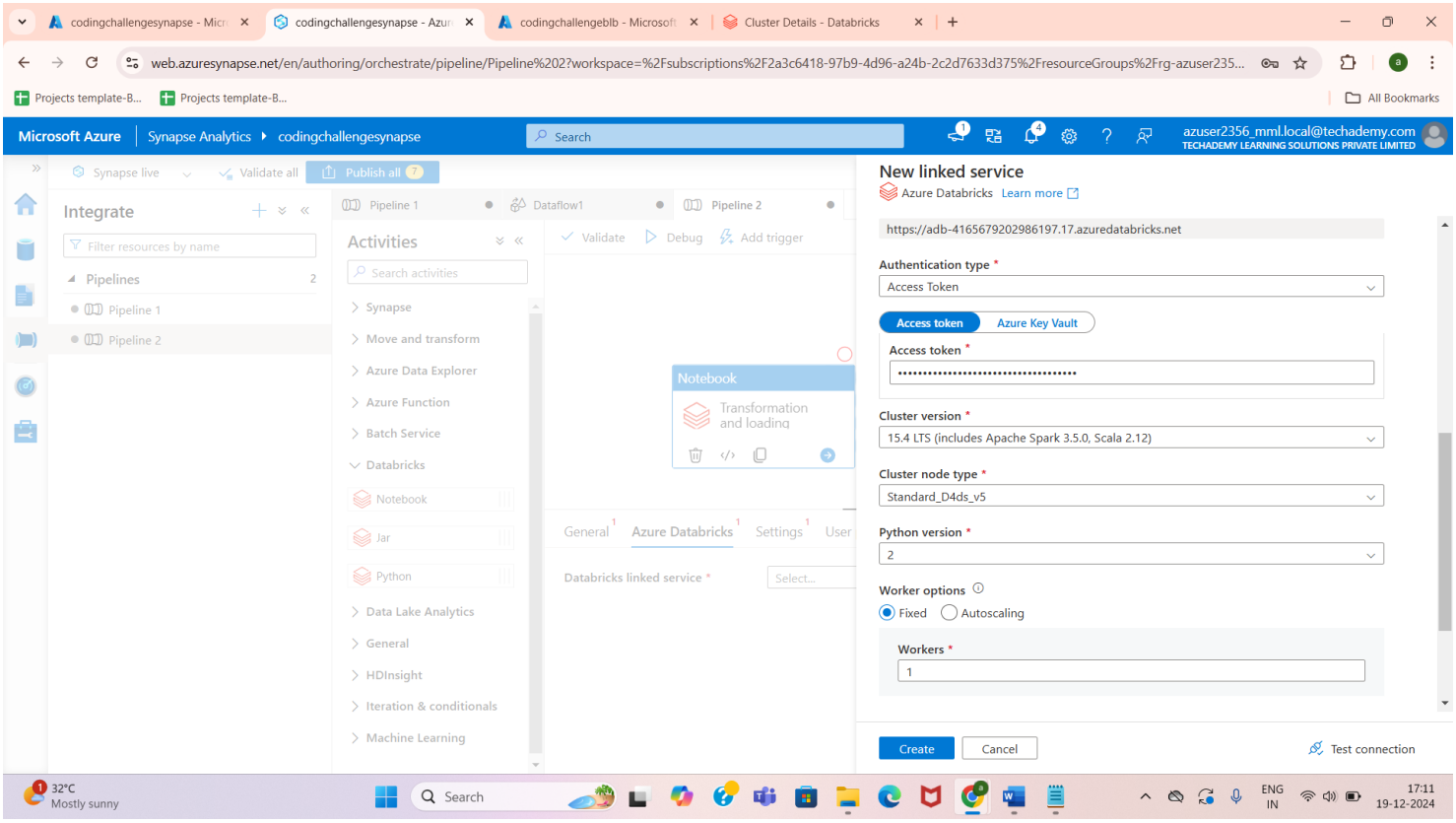
(9) Spark Jobs

## *Configure Databricks and the Databricks notebook to Azure Synapse Pipeline:*

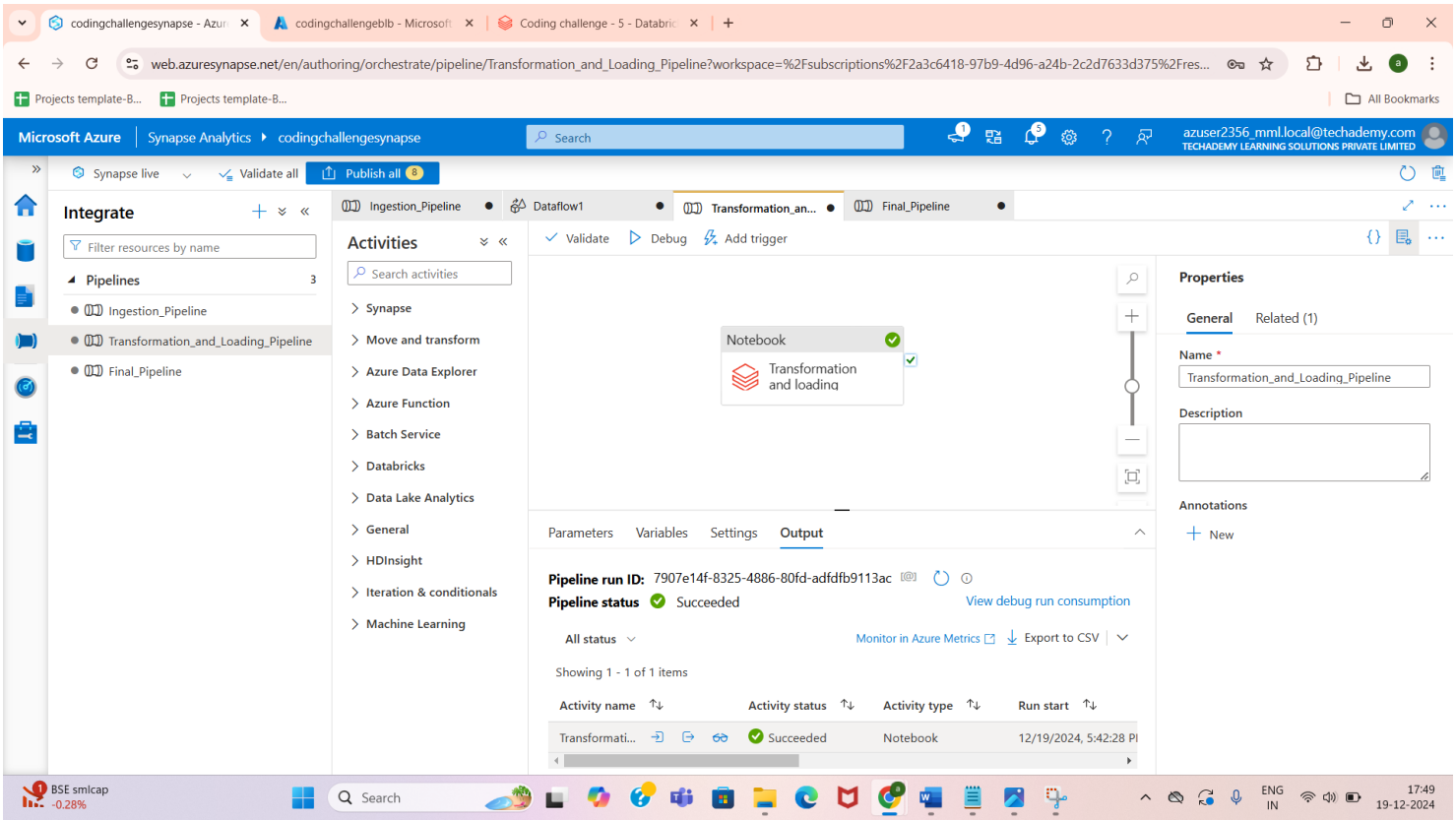



## *Generate a new token in Databricks to connect to Azure Synapse:*

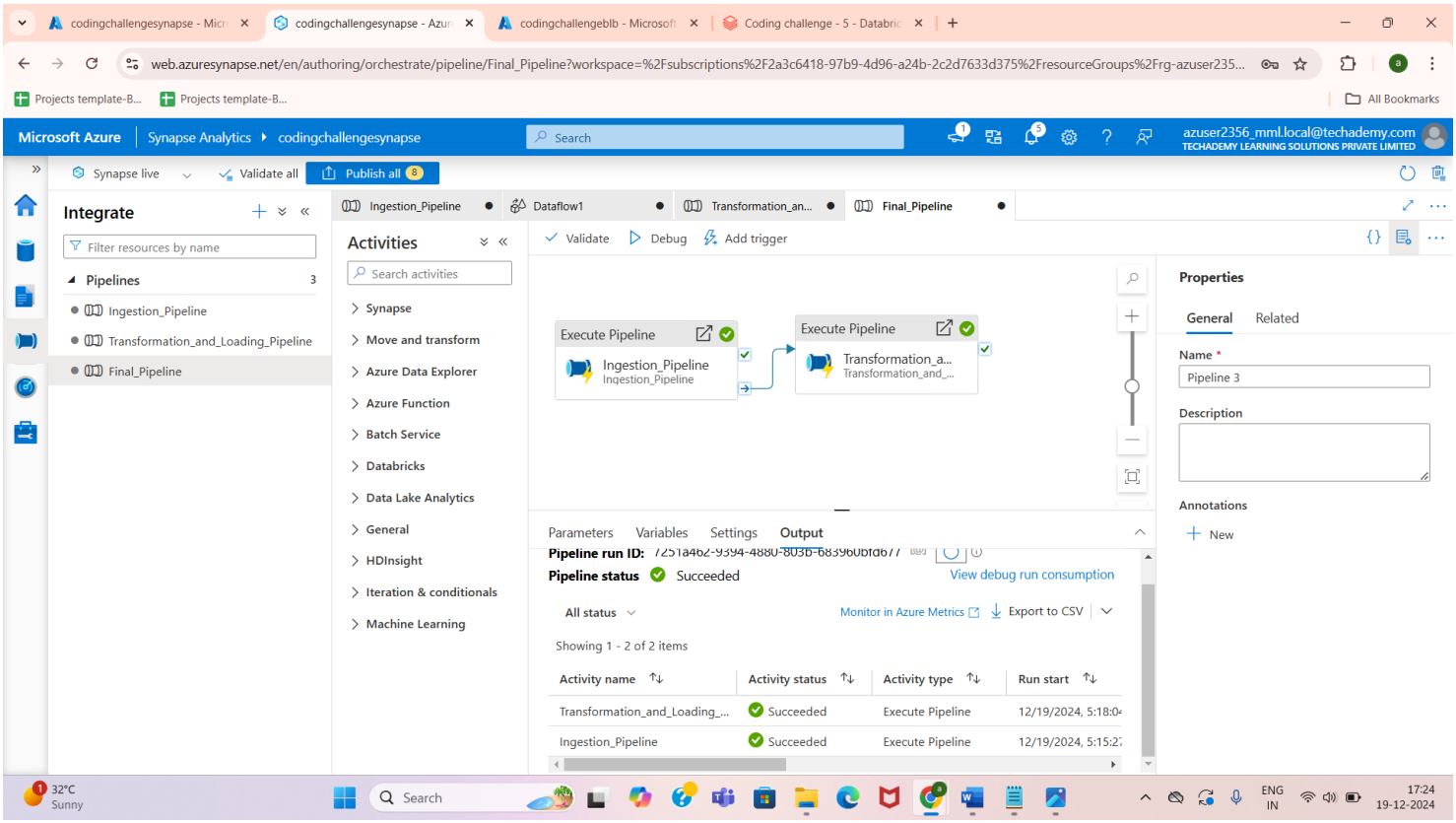





## *Validate and Debug the pipeline:*



***Create the final pipeline to connect the ingestion pipeline and Transformation pipeline and then Validate and Debug it:***




**Submitted By:**
Aathirainathan P