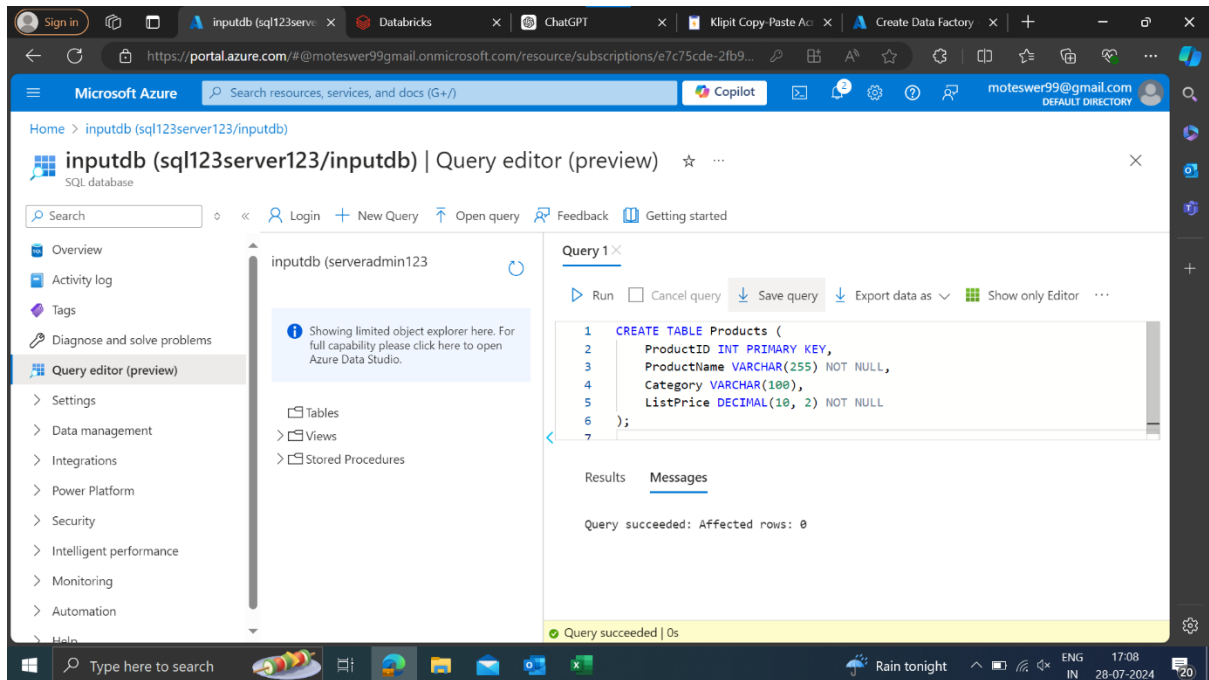


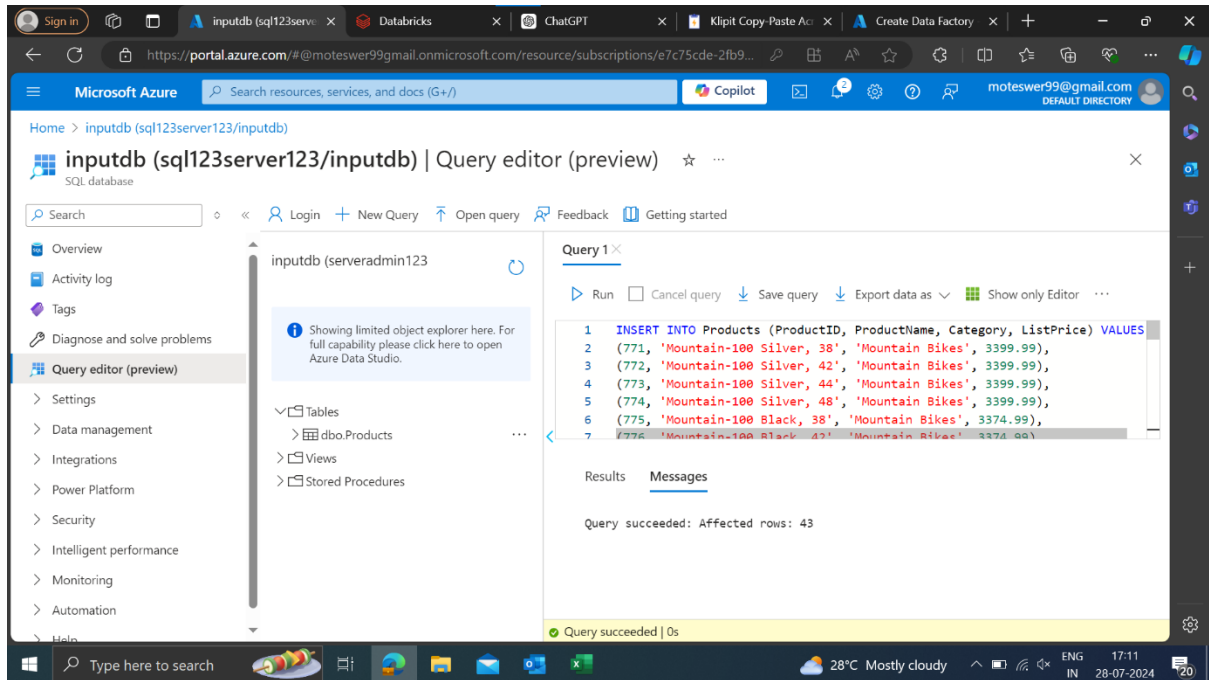
# ASSIGNMENT-7

## SQL-DATALAKE-DATABRICKS Pipeline

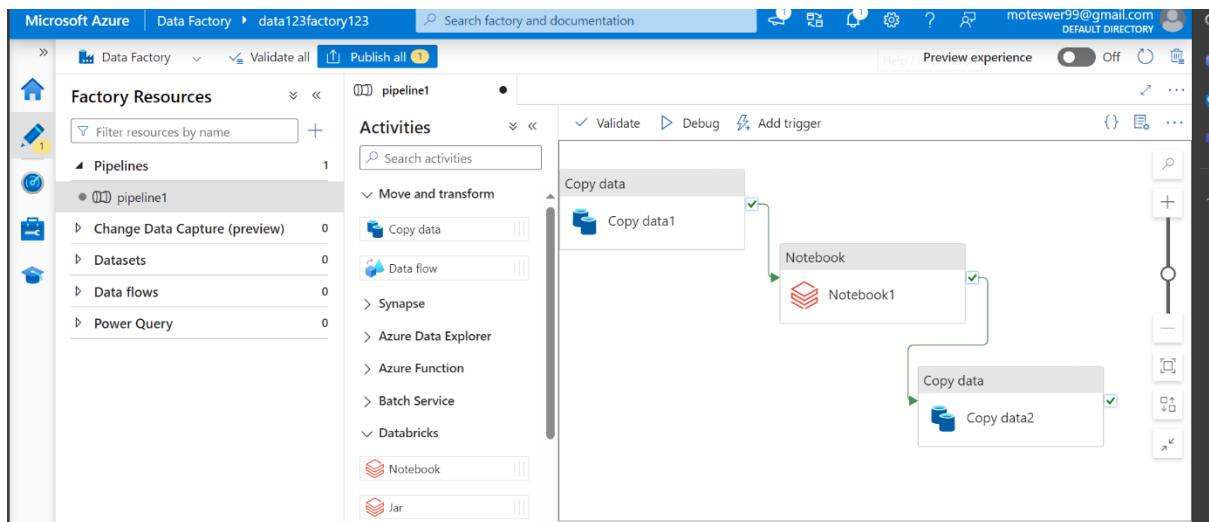
Created a source table named Products



# Inserted data from csv into products table



# Created pipeline connecting with databricks notebook



# Created notebook with spark program to read the mounted data from datalake

The image shows two screenshots of the Databricks interface. The top screenshot displays a notebook with a Spark program that reads data from a mounted Azure Data Lake. The code sets the storage account name, container name, and file path, then uses Spark's read format to load the data as a DataFrame. The bottom screenshot shows the output of the program, which is a table with 43 rows and 5 columns: ProductID, ProductName, Category, and ListPrice. The table contains data for various products, including Mountain Bike Socks, Mountain Bikes, and Forks.

**products\_notebook** Python

```
storage_account_name = "tutorialdatalake123"
container_name = "datalakecontainer"
file_path = "order_output/dbo.Products.txt"

spark.conf.set(
    f"fs.azure.account.key.{storage_account_name}.blob.core.windows.net",
    "vx1Hw1mvvbdKbnvtMG6TUSNFnuuB3OJ1UPAEi8qa0kRr1eCmFYJ1MmINwGVLI7k8Wmx96p1R1hg+ASTVLI1+A=="
)

df = spark.read.format("csv") \
    .option("header", "true") \
    .option("inferSchema", "true") \
    .load(f"wasbs://{container_name}@{storage_account_name}.blob.core.windows.net/{file_path}")

display(df)
```

(3) Spark Jobs

df: pyspark.sql.dataframe.DataFrame = [ProductID: integer, ProductName: string ... 2 more fields]

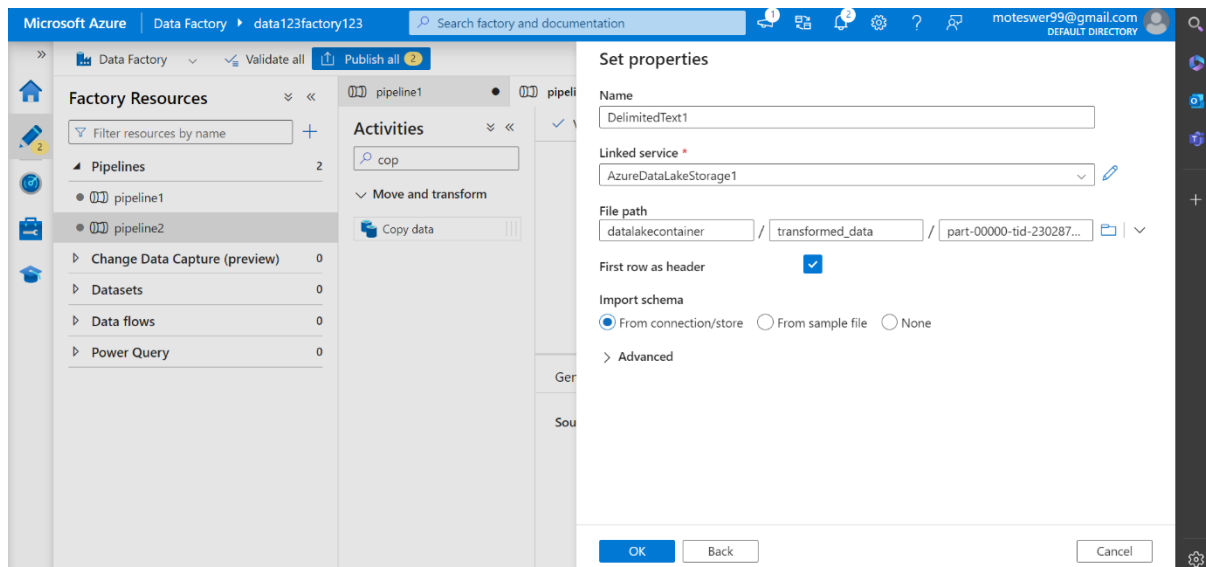
**Table**

	ProductID	ProductName	Category	ListPrice
1	709	Mountain Bike Socks, M	Socks	9.5
2	710	Mountain Bike Socks, L	Socks	9.5
3	771	Mountain-100 Silver, 38	Mountain Bikes	3399.99
4	772	Mountain-100 Silver, 42	Mountain Bikes	3399.99
5	773	Mountain-100 Silver, 44	Mountain Bikes	3399.99
6	774	Mountain-100 Silver, 48	Mountain Bikes	3399.99
7	775	Mountain-100 Black, 38	Mountain Bikes	3374.99
8	776	Mountain-100 Black, 42	Mountain Bikes	3374.99
9	777	Mountain-100 Black, 44	Mountain Bikes	3374.99
10	802	LL Fork	Forks	148.22
11	803	ML Fork	Forks	175.49
12	804	HL Fork	Forks	229.49
13	805	LL Headset	Headsets	34.2
14	806	ML Headset	Headsets	102.29
15	807	HL Headset	Headsets	124.73

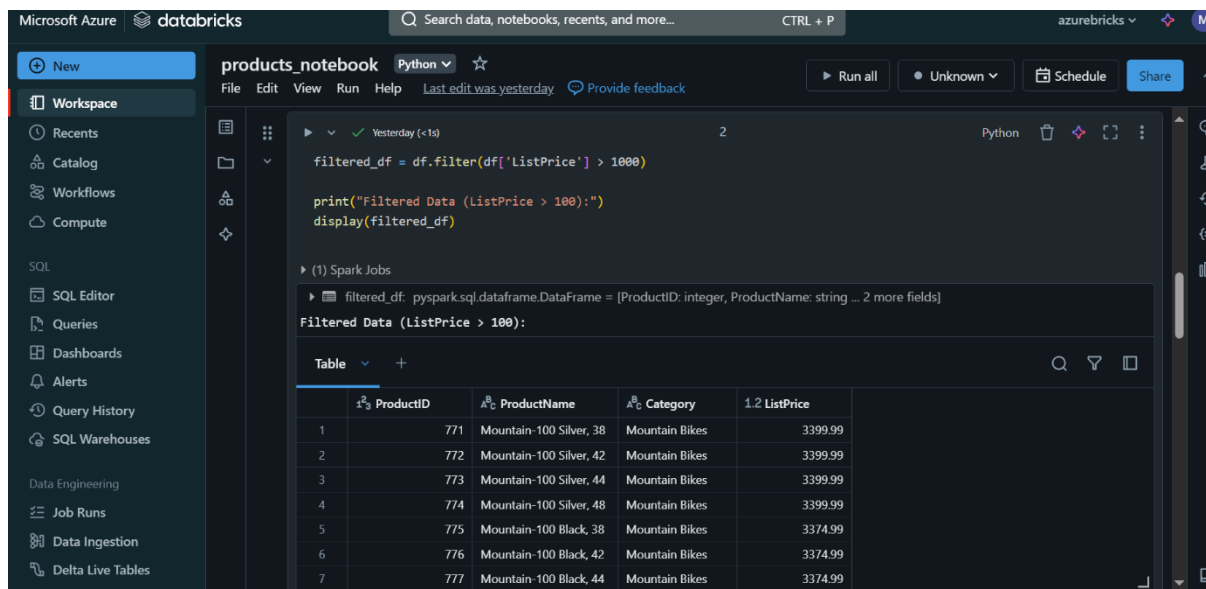
43 rows | 16.50 seconds runtime

Refreshed yesterday

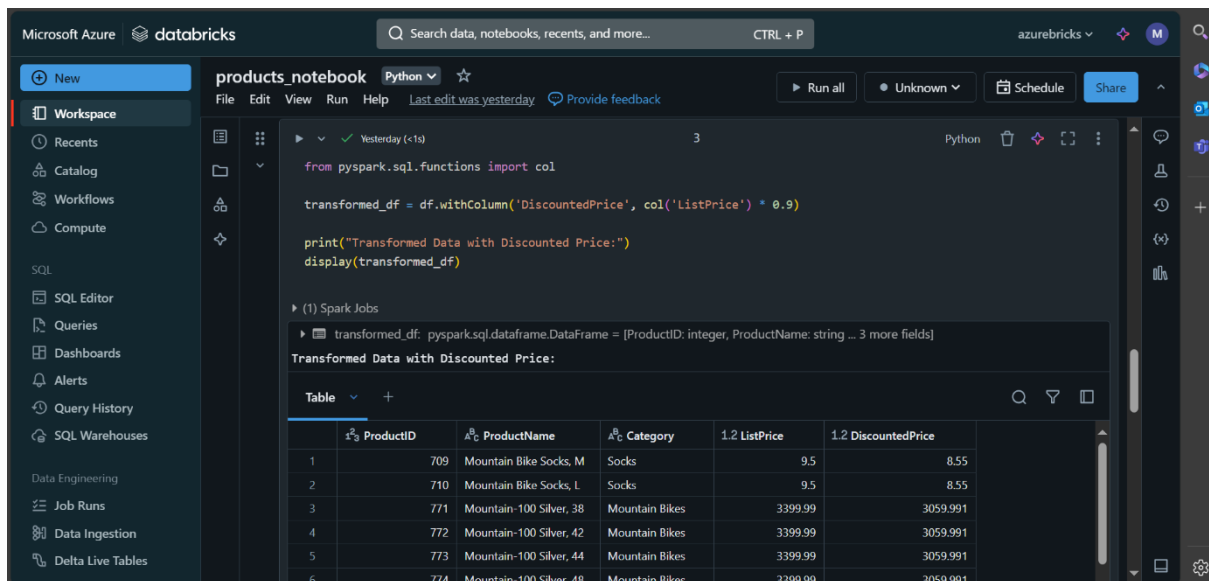
# Connected the transformed data from datalake to sql database



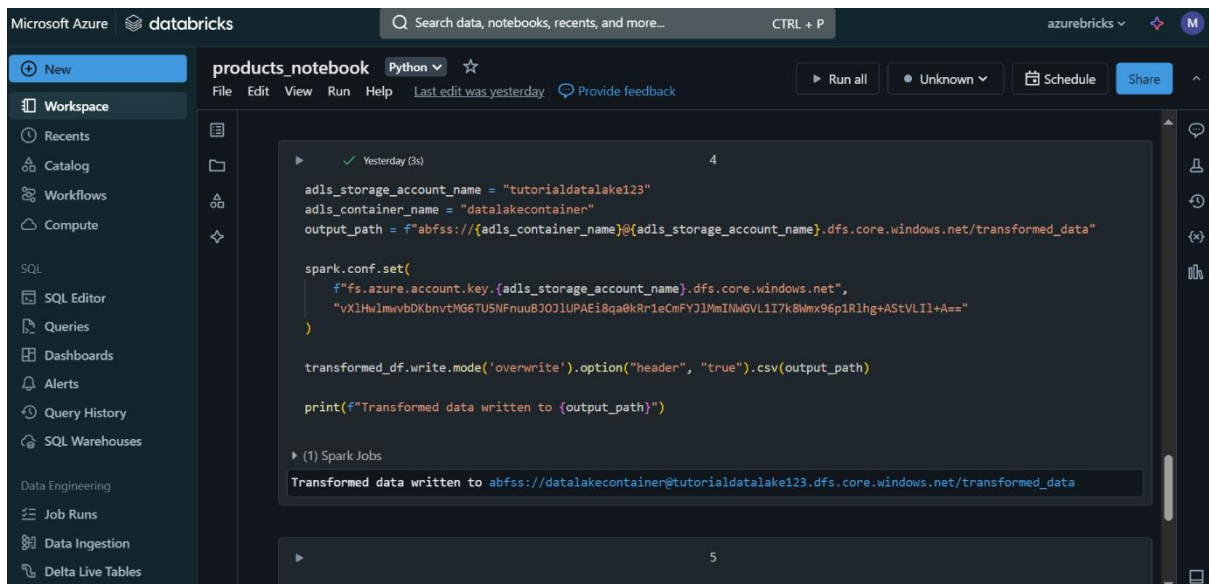
# Performed data transformation on the data in the notebook using spark



Stored the transformed data in a transformed\_df variable



Wrote the transformed data into the datalake container



# Original data in Products table

The screenshot shows the Microsoft Azure portal interface for the 'inputdb (sql123server123/inputdb)' SQL database. The 'Query editor (preview)' is open, displaying a query that selects the top 1000 rows from the 'Products' table. The results are shown in a table with columns: ProductID, ProductName, Category, and ListPrice. The table contains two rows of data.

ProductID	ProductName	Category	ListPrice
707	Sport-100 Helmet, Red	Helmets	34.99
708	Sport-100 Helmet, Black	Helmets	34.99

# Products table after transformation

The screenshot shows the Microsoft Azure portal interface for the 'inputdb (sql123server123/inputdb)' SQL database. The 'Query editor (preview)' is open, displaying a query that selects the top 1000 rows from the 'Products\_New' table. The results are shown in a table with columns: ProductID, ProductName, Category, ListPrice, and DiscountedPrice. The table contains two rows of data. A status bar at the bottom indicates 'Query succeeded | 0s'.

ProductID	ProductName	Category	ListPrice	DiscountedPrice
709	Mountain Bike Socks, M	Socks	9.50	8.55
710	Mountain Bike Socks, L	Socks	9.50	8.55