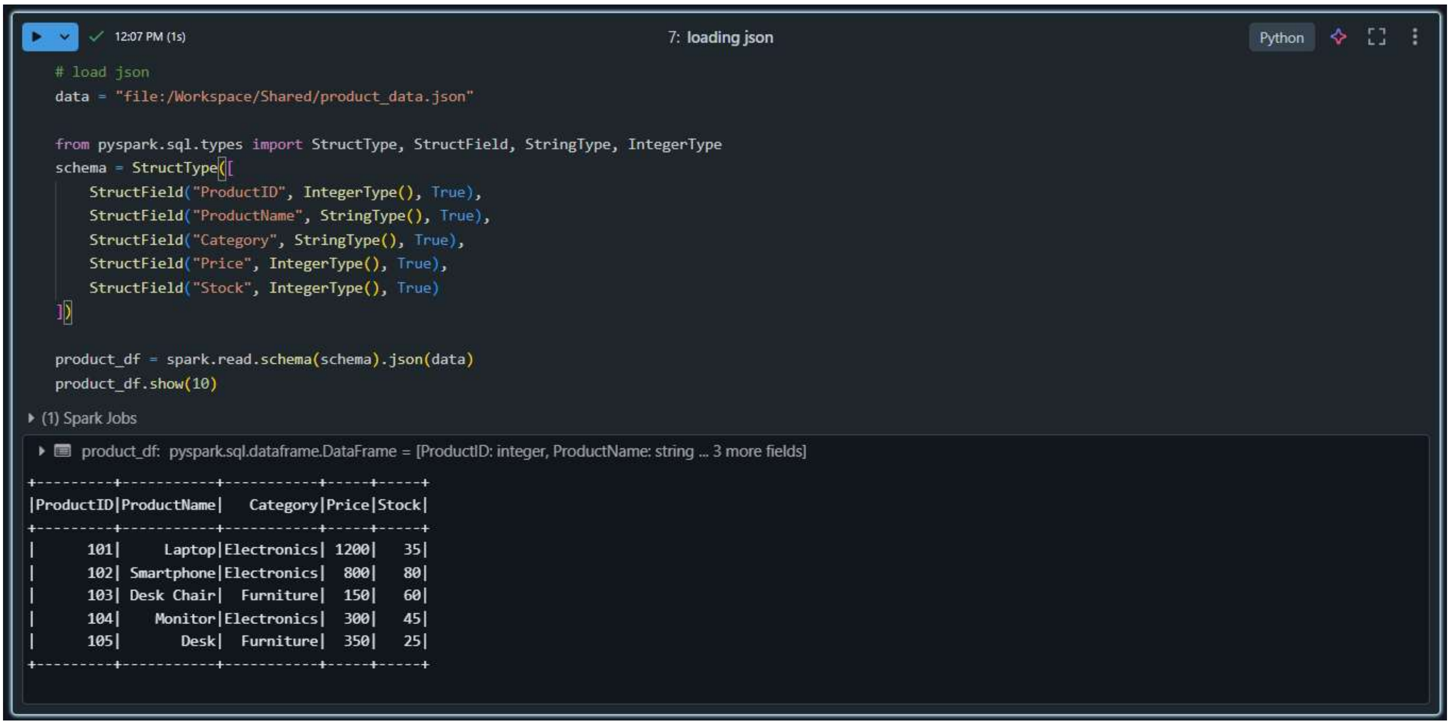
Azure DataBricks Assignment – 12th September

SaiPrabath Chowdary S

## Assignment 2

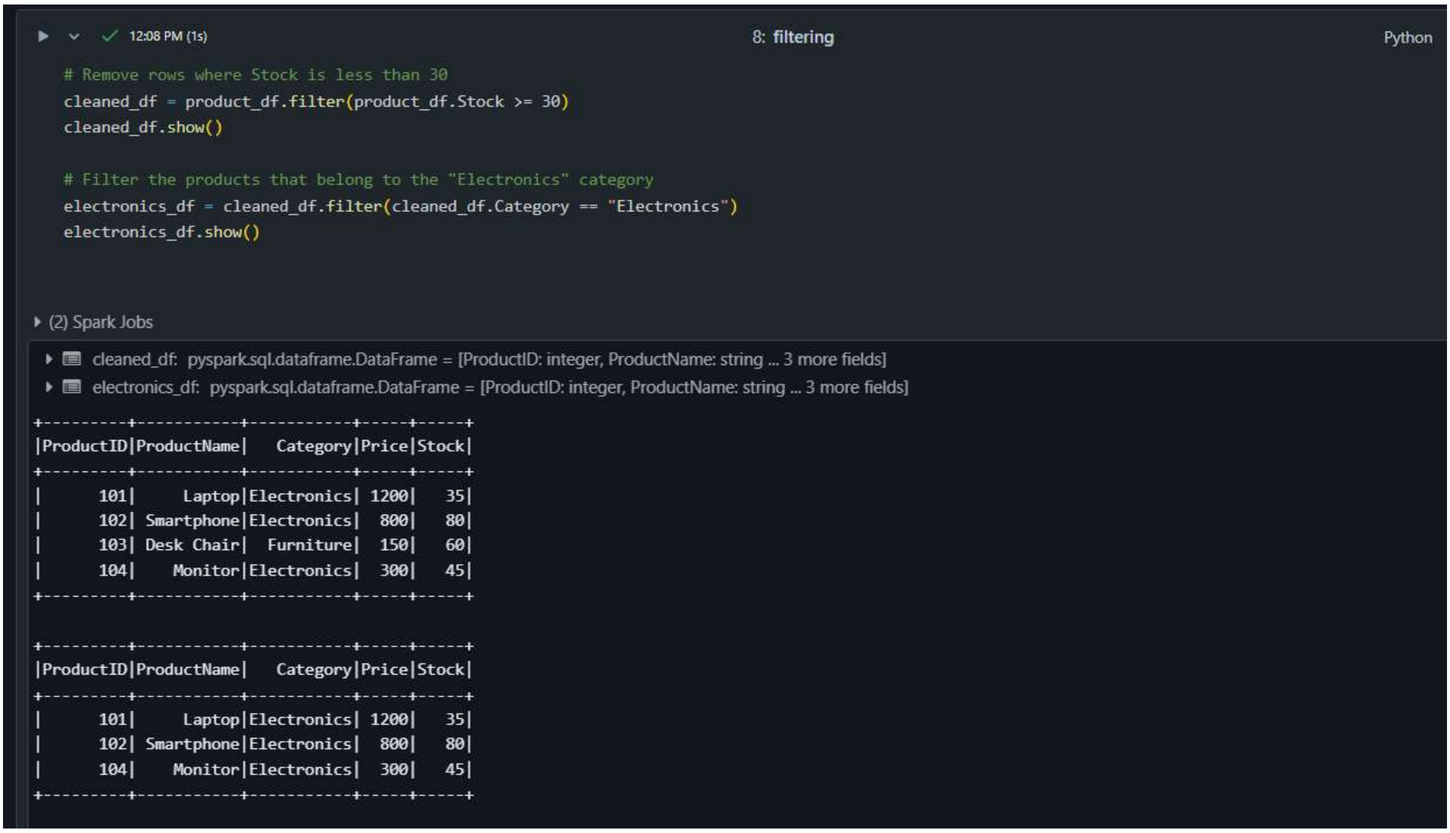


```
# load json
data = "file:/Workspace/Shared/product_data.json"

from pyspark.sql.types import StructType, StructField, StringType, IntegerType
schema = StructType([
    StructField("ProductID", IntegerType(), True),
    StructField("ProductName", StringType(), True),
    StructField("Category", StringType(), True),
    StructField("Price", IntegerType(), True),
    StructField("Stock", IntegerType(), True)
])

product_df = spark.read.schema(schema).json(data)
product_df.show(10)
```

(1) Spark Jobs

product_df: pyspark.sql.dataframe.DataFrame = [ProductID: integer, ProductName: string ... 3 more fields]

```
+---------+-----------+-----------+-----+-----+
|ProductID|ProductName|   Category|Price|Stock|
+---------+-----------+-----------+-----+-----+
|      101|     Laptop|Electronics| 1200|   35|
|      102| Smartphone|Electronics|  800|   80|
|      103| Desk Chair|  Furniture|  150|   60|
|      104|    Monitor|Electronics|  300|   45|
|      105|       Desk|  Furniture|  350|   25|
+---------+-----------+-----------+-----+-----+
```



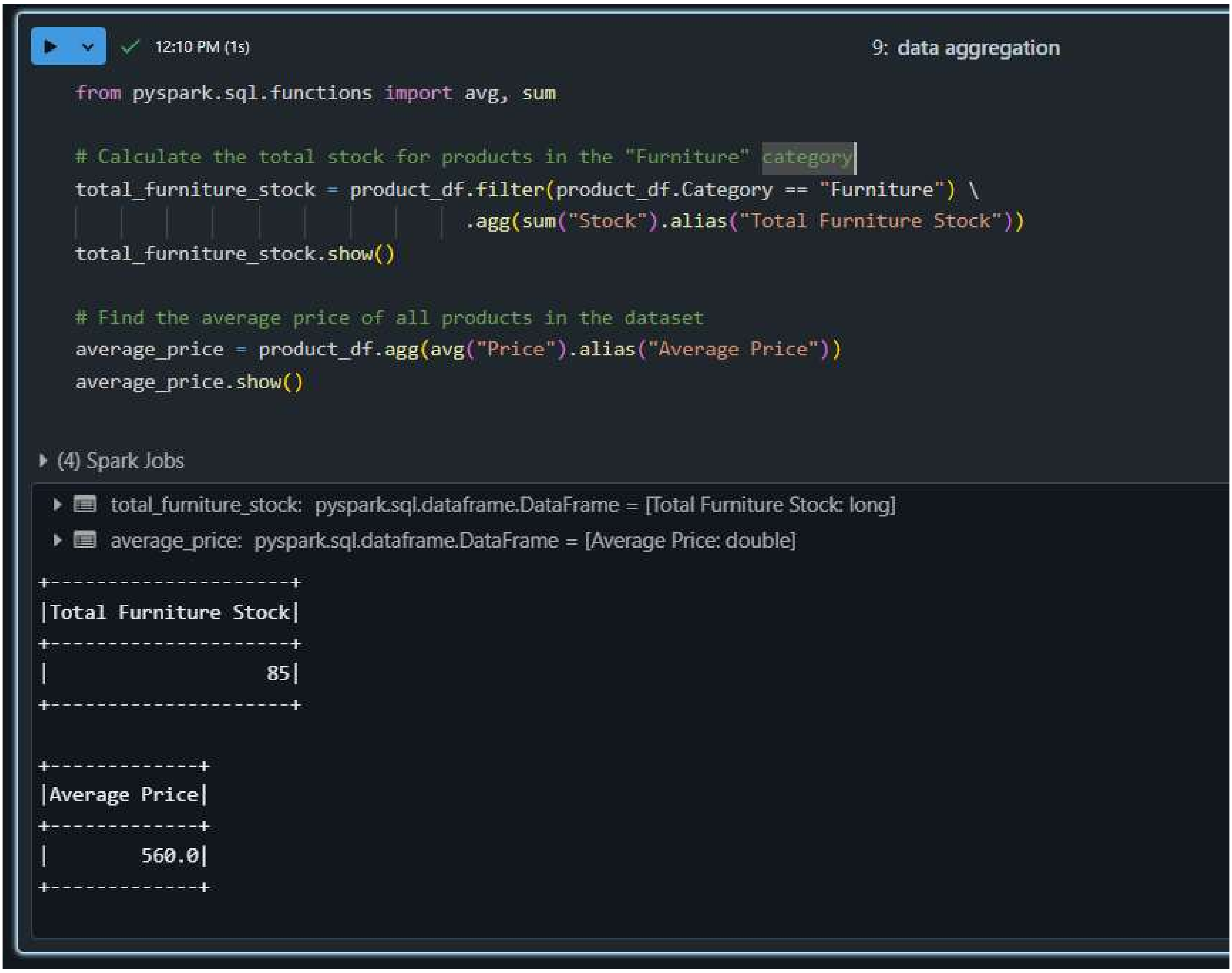
```
# Remove rows where Stock is less than 30
cleaned_df = product_df.filter(product_df.Stock >= 30)
cleaned_df.show()

# Filter the products that belong to the "Electronics" category
electronics_df = cleaned_df.filter(cleaned_df.Category == "Electronics")
electronics_df.show()
```

(2) Spark Jobs

cleaned_df: pyspark.sql.dataframe.DataFrame = [ProductID: integer, ProductName: string ... 3 more fields]

electronics_df: pyspark.sql.dataframe.DataFrame = [ProductID: integer, ProductName: string ... 3 more fields]

```
+---------+-----------+-----------+-----+-----+
|ProductID|ProductName|   Category|Price|Stock|
+---------+-----------+-----------+-----+-----+
|      101|     Laptop|Electronics| 1200|   35|
|      102| Smartphone|Electronics|  800|   80|
|      103| Desk Chair|  Furniture|  150|   60|
|      104|    Monitor|Electronics|  300|   45|
+---------+-----------+-----------+-----+-----+

+---------+-----------+-----------+-----+-----+
|ProductID|ProductName|   Category|Price|Stock|
+---------+-----------+-----------+-----+-----+
|      101|     Laptop|Electronics| 1200|   35|
|      102| Smartphone|Electronics|  800|   80|
|      104|    Monitor|Electronics|  300|   45|
+---------+-----------+-----------+-----+-----+
```



12:10 PM (1s)

9: data aggregation

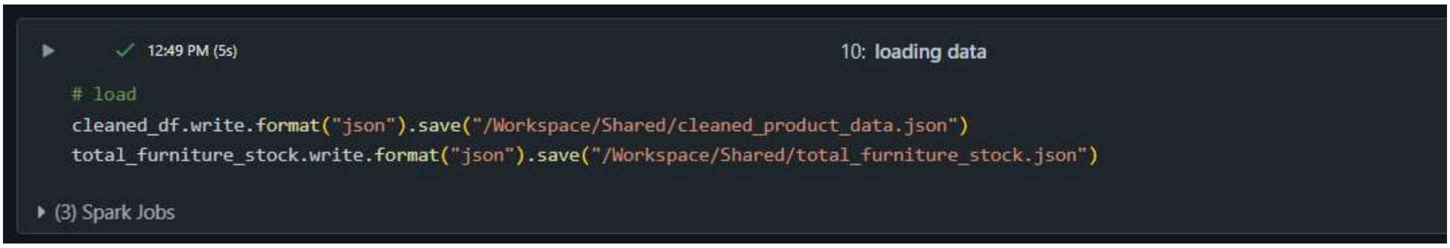
```
from pyspark.sql.functions import avg, sum

# Calculate the total stock for products in the "Furniture" category
total_furniture_stock = product_df.filter(product_df.Category == "Furniture") \
                                   .agg(sum("Stock").alias("Total Furniture Stock"))
total_furniture_stock.show()

# Find the average price of all products in the dataset
average_price = product_df.agg(avg("Price").alias("Average Price"))
average_price.show()
```

(4) Spark Jobs

total_furniture_stock: pyspark.sql.dataframe.DataFrame = [Total Furniture Stock: long]

average_price: pyspark.sql.dataframe.DataFrame = [Average Price: double]

```
+---------------------+
|Total Furniture Stock|
+---------------------+
|                   85|
+---------------------+

+-------------+
|Average Price|
+-------------+
|        560.0|
+-------------+
```



12:49 PM (5s)

10: loading data

```
# load
cleaned_df.write.format("json").save("/Workspace/Shared/cleaned_product_data.json")
total_furniture_stock.write.format("json").save("/Workspace/Shared/total_furniture_stock.json")
```

(3) Spark Jobs