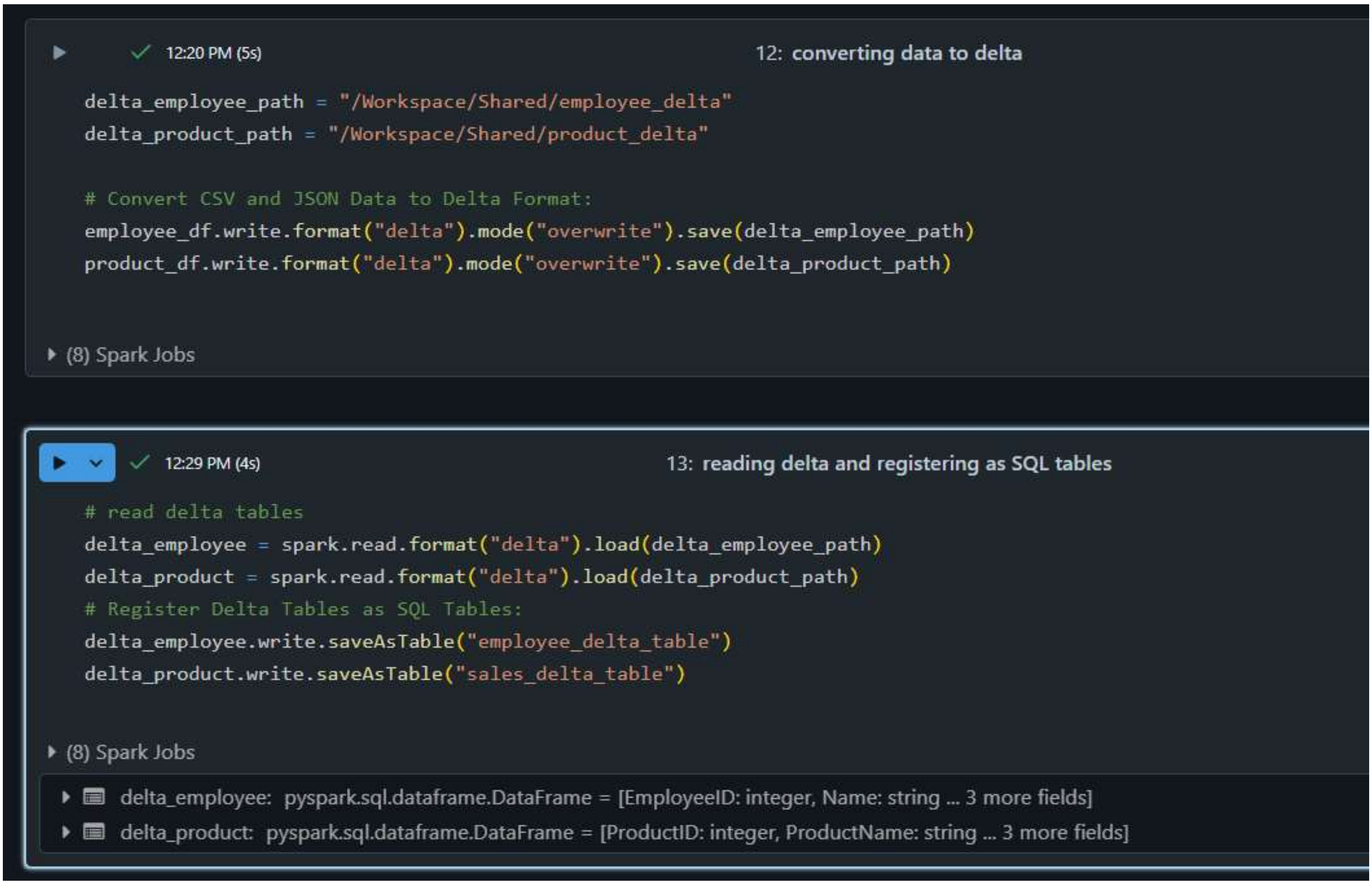
Azure DataBricks Assignment – 12th September

SaiPrabath Chowdary S

Assignment 3



12:20 PM (5s)
12: converting data to delta

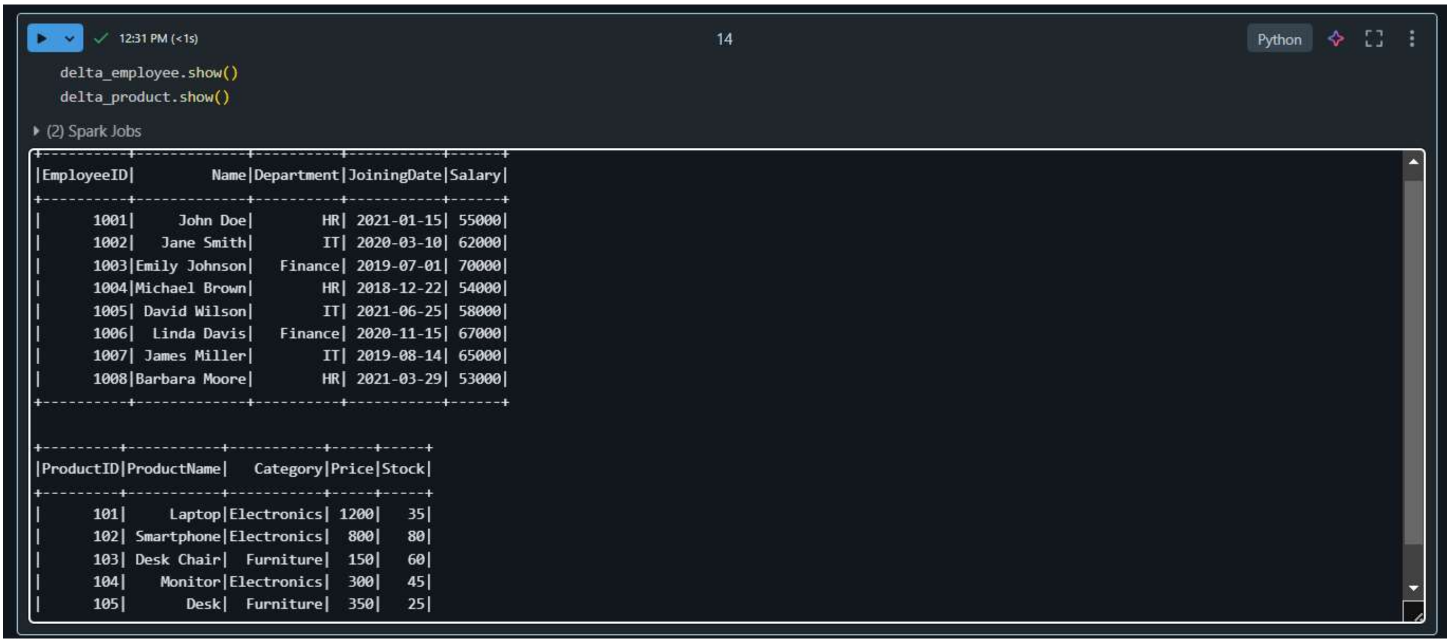
```
delta_employee_path = "/Workspace/Shared/employee_delta"
delta_product_path = "/Workspace/Shared/product_delta"

# Convert CSV and JSON Data to Delta Format:
employee_df.write.format("delta").mode("overwrite").save(delta_employee_path)
product_df.write.format("delta").mode("overwrite").save(delta_product_path)
```

(8) Spark Jobs
12:29 PM (4s)
13: reading delta and registering as SQL tables

```
# read delta tables
delta_employee = spark.read.format("delta").load(delta_employee_path)
delta_product = spark.read.format("delta").load(delta_product_path)
# Register Delta Tables as SQL Tables:
delta_employee.write.saveAsTable("employee_delta_table")
delta_product.write.saveAsTable("sales_delta_table")
```

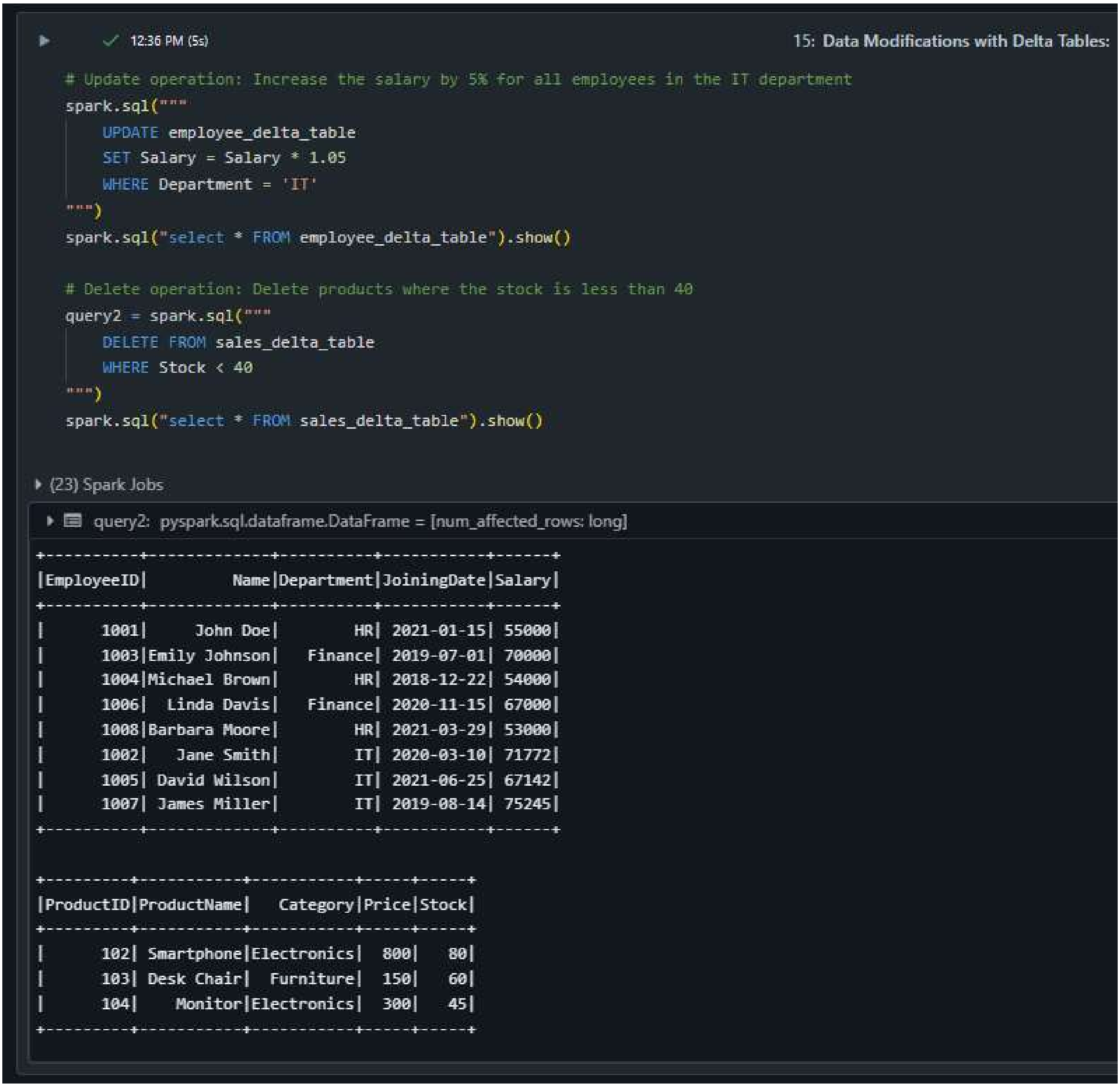
```
delta_employee:  pyspark.sql.dataframe.DataFrame = [EmployeeID: integer, Name: string ... 3 more fields]
delta_product:  pyspark.sql.dataframe.DataFrame = [ProductID: integer, ProductName: string ... 3 more fields]
```



```
delta_employee.show()
delta_product.show()
```

```
+----------+-------------+----------+-----------+------+
|EmployeeID|         Name|Department|JoiningDate|Salary|
+----------+-------------+----------+-----------+------+
|      1001|     John Doe|        HR| 2021-01-15| 55000|
|      1002|   Jane Smith|        IT| 2020-03-10| 62000|
|      1003|Emily Johnson|   Finance| 2019-07-01| 70000|
|      1004|Michael Brown|        HR| 2018-12-22| 54000|
|      1005| David Wilson|        IT| 2021-06-25| 58000|
|      1006|  Linda Davis|   Finance| 2020-11-15| 67000|
|      1007| James Miller|        IT| 2019-08-14| 65000|
|      1008|Barbara Moore|        HR| 2021-03-29| 53000|
+----------+-------------+----------+-----------+------+


+---------+-----------+-----------+-----+-----+
|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|
```



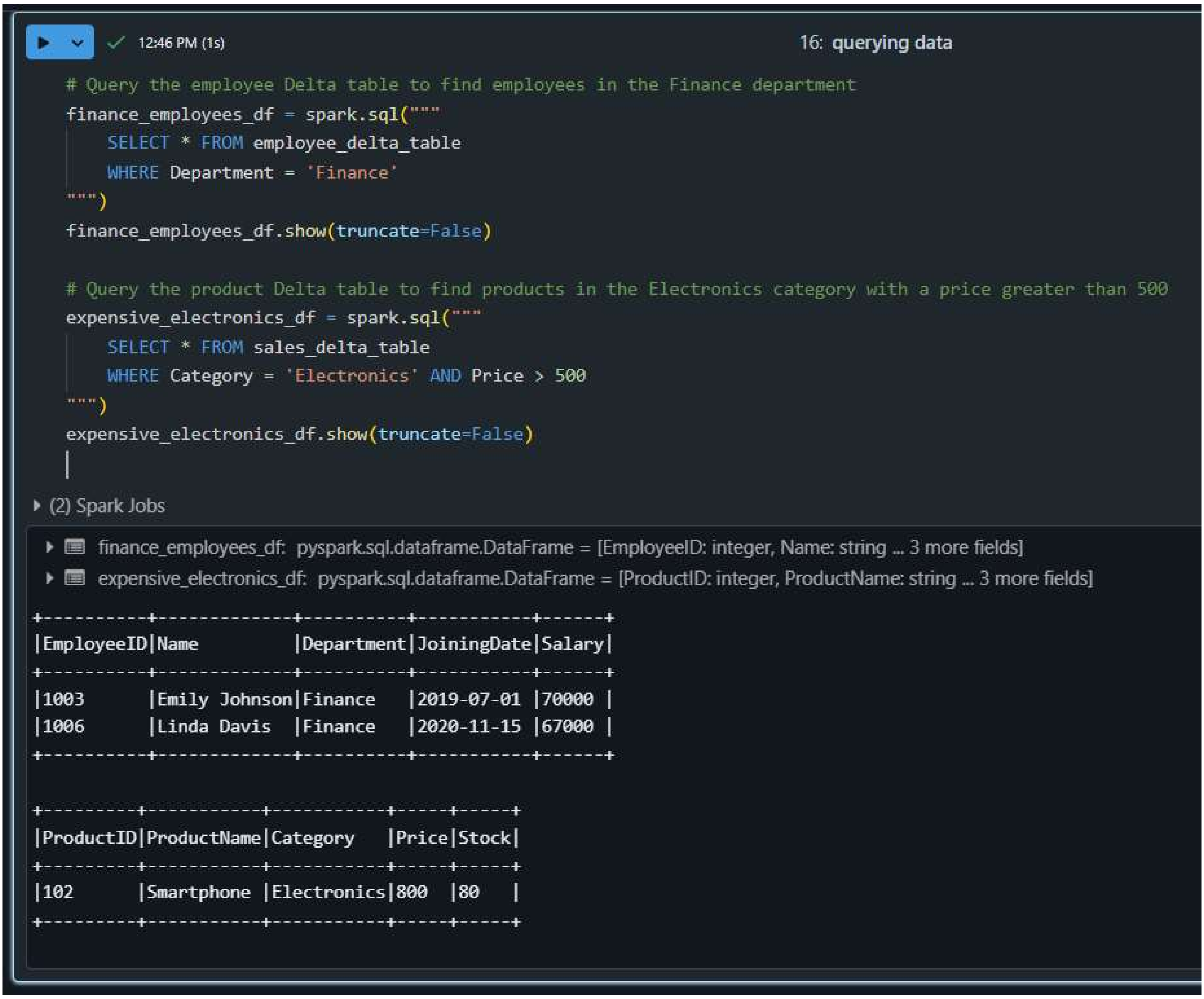
```
# Update operation: Increase the salary by 5% for all employees in the IT department
spark.sql("""
    UPDATE employee_delta_table
    SET Salary = Salary * 1.05
    WHERE Department = 'IT'
""")
spark.sql("select * FROM employee_delta_table").show()

# Delete operation: Delete products where the stock is less than 40
query2 = spark.sql("""
    DELETE FROM sales_delta_table
    WHERE Stock < 40
""")
spark.sql("select * FROM sales_delta_table").show()
```

query2: pyspark.sql.dataframe.DataFrame = [num_affected_rows: long]

```
+----------+-------------+----------+-----------+------+
|EmployeeID|         Name|Department|JoiningDate|Salary|
+----------+-------------+----------+-----------+------+
|      1001|     John Doe|        HR| 2021-01-15| 55000|
|      1003|Emily Johnson|   Finance| 2019-07-01| 70000|
|      1004|Michael Brown|        HR| 2018-12-22| 54000|
|      1006|   Linda Davis|   Finance| 2020-11-15| 67000|
|      1008|Barbara Moore|        HR| 2021-03-29| 53000|
|      1002|   Jane Smith|        IT| 2020-03-10| 71772|
|      1005| David Wilson|        IT| 2021-06-25| 67142|
|      1007| James Miller|        IT| 2019-08-14| 75245|
+----------+-------------+----------+-----------+------+

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



```
# Query the employee Delta table to find employees in the Finance department
finance_employees_df = spark.sql("""
    SELECT * FROM employee_delta_table
    WHERE Department = 'Finance'
""")
finance_employees_df.show(truncate=False)

# Query the product Delta table to find products in the Electronics category with a price greater than 500
expensive_electronics_df = spark.sql("""
    SELECT * FROM sales_delta_table
    WHERE Category = 'Electronics' AND Price > 500
""")
expensive_electronics_df.show(truncate=False)
```

finance_employees_df: pyspark.sql.dataframe.DataFrame = [EmployeeID: integer, Name: string ... 3 more fields]

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

```
+----------+-------------+----------+-----------+------+
|EmployeeID|Name         |Department|JoiningDate|Salary|
+----------+-------------+----------+-----------+------+
|1003      |Emily Johnson|Finance   |2019-07-01 |70000 |
|1006      |Linda Davis  |Finance   |2020-11-15 |67000 |
+----------+-------------+----------+-----------+------+


+---------+-----------+-----------+-----+-----+
|ProductID|ProductName|Category   |Price|Stock|
+---------+-----------+-----------+-----+-----+
|102      |Smartphone |Electronics|800  |80   |
+---------+-----------+-----------+-----+-----+
```