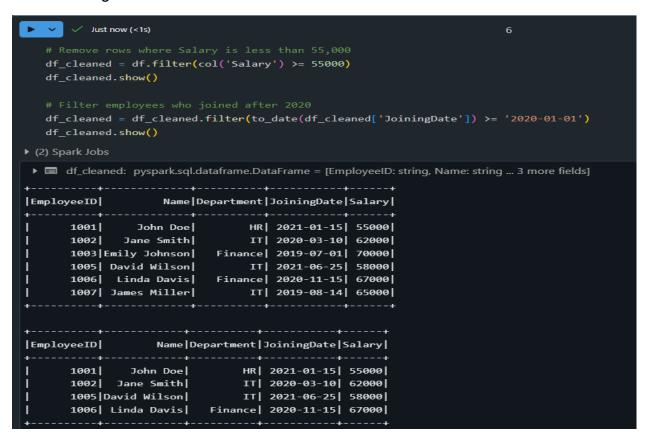
Assignment 1: Working with CSV Data (employee_data.csv)

Tasks:

1. Load the CSV data:

```
2 minutes ago (1s)
      1 from pyspark.sql import SparkSession
         from pyspark.sql.functions import col, to_date, avg, count
      5 spark = SparkSession.builder.appName("EmployeeDataAnalysis").getOrCreate()
      7 dbutils.fs.cp("file:/Workspace/Users/azuser2130_mml.local@techademy.com/Employees_data.csv", "dbfs:/FileStore/Employees_data.csv")
      9 df = spark.read.format("csv").option("header","true").load("dbfs:/FileStore/Employees_data.csv")
     11 df.show(10)
Output Terminal Debug Console
    ▶ ■ df: pyspark.sql.dataframe.DataFrame = [EmployeeID: string, Name: string ... 3 more fields]
   |EmployeeID|
                         Name|Department|JoiningDate|Salary|
                   John Doe | HR | 2021-01-15 | 55000 |
Jane Smith | IT | 2020-03-10 | 62000 |
           1002 Jane Smith
           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 |
```

2. Data Cleaning:



3. Data Aggregation:

```
Just now (1s)
        average_salary_by_department = df_cleaned.groupby('Department').agg(avg('Salary').alias('AverageSalary'))
        print("Average Salary by Department:\n")
        average_salary_by_department.show()
       employee_count_by_department = df_cleaned.groupby('Department').count()
        print("Employee Count by Department:\n")
        employee_count_by_department.show()
Output Terminal Debug Console
   Average Salary by Department:
   +----+
   |Department|AverageSalary|
           HR
                   55000.0
                   67000.0
      Finance
           IT
                   60000.0
   Employee Count by Department:
   +----+
   |Department|count|
           HR
                  1
                  1
      Finance
           IT
                  2
```

4. Write the Data to CSV:

Save the cleaned data (from the previous steps) to a new CSV file.

```
# Save the cleaned data to a new CSV file

df_cleaned.write.mode("overwrite").csv("dbfs:/path/to/cleaned_employee_data.csv", header=True)

(1) Spark Jobs
```