Working on PySpark in Jupyter notebook

-Open an Anaconda Navigator and click **launch** and in top right corner **click new** in that choose **python.**

-Lets takes a students values as example.

Create CSV file:

-create a csv file named as Student in **notepad**.

ROLL\_NO,CGP,ATTEDENCE\_PERCENTAGE

,,,

1,9.012,89%

2,7.894,85%

3,8.543,90%

-save the file as student.csv in all types of file.

Create values in MySql :

-create a connection with password.

-create a **schema** and create **database** and **tables**.

-refer the video for your reference :

<https://youtu.be/wALCw0F8e9M?si=xol-1BA7rkBVHbHw>

<https://youtu.be/UzodkZUt5JY?si=9wW1T1KLnx057NkP>

* Store the vales of student with ROLL\_NO as key
* +-------+-----+----------+---+
* |ROLL\_NO| NAME|DEPARTMENT|AGR|
* +-------+-----+----------+---+
* | 1|Alice| CSE| 19|
* | 2| Bob| IT| 19|
* | 3| Chen| AIDS| 19|
* +-------+-----+----------+---+

PySpark Code:

-lets create a csv file and mysql and **joined** them and **write to another csv file and mysql.**

Cell 1 : !pip install pyspark

Cell 2 : from pyspark.sql import SparkSession

# Replace this **path** with your actual JDBC driver location

jdbc\_driver\_path = "C:\\jar\\mysql-connector-j-9.2.0 (1)\\mysql-connector-j-9.2.0"

# Create Spark session

spark = SparkSession.builder \

.appName("MySQL + CSV in Jupyter") \

.config("spark.jars", jdbc\_driver\_path) \

.getOrCreate()

Cell 3 : # install mysql-connector

!pip install mysql-connector-python

pip install sqlalchemy mysql-connector-python

Cell 4 : from pyspark.sql import SparkSession

from pyspark.sql import SparkSession

# Initialize Spark Session

spark = SparkSession.builder \

.appName("MySQL to Spark") \

.config("spark.jars", "mysql-connector-java-8.0.33.jar") \

.getOrCreate()

# Define JDBC properties

mysql\_url = "jdbc:mysql://localhost:3306/company"

mysql\_properties = {

"user": "root",

"password": "My@3066Sql", # Note: don't URL-encode special characters here

"driver": "com.mysql.cj.jdbc.Driver"

}

# Table name

table\_name = "student"

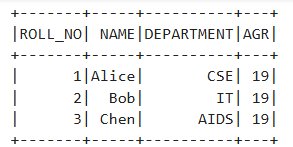
# Read MySQL table into PySpark DataFrame

df\_mysql = spark.read.jdbc(url=mysql\_url, table=table\_name, properties=mysql\_properties)

# Show top records

df\_mysql.show()

output:



Cell 4 : # Load CSV file (should be in same dir or give full path)

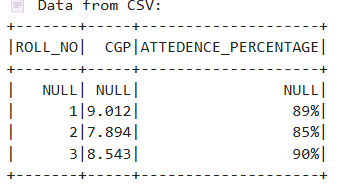
df\_csv = spark.read.csv("D:\\csv file\\Student.csv", header=True, inferSchema=True)

# Show CSV data

print("📄 Data from CSV:")

df\_csv.show()

output :



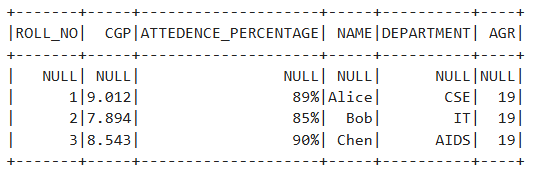
Cell 5 : #join the csv file and mysql

df\_joined = df\_csv.join(df\_mysql, on="ROLL\_NO", how="full")

# Show result

df\_joined.show()

output :



Cell 6 : # write the joined table to mysql

from pyspark.sql import SparkSession

spark = SparkSession.builder \

.appName("SaveToMySQL") \

.config("spark.jars", "C:/path/to/mysql-connector-j-8.0.32.jar") \

.getOrCreate()

jdbc\_url = "jdbc:mysql://localhost:3306/students"

table\_name = "student\_report"

connection\_properties = {

"user": "root",

"password": "My@3066Sql",

"driver": "com.mysql.cj.jdbc.Driver"

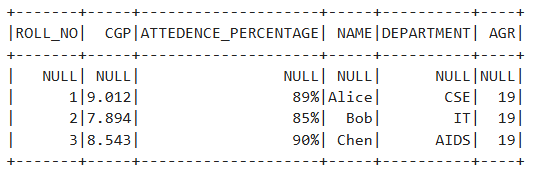
}

df\_joined.write \

.jdbc(url=jdbc\_url, table=table\_name, mode="overwrite", properties=connection\_properties)

df\_joined.show()

output :



Cell 7 : # converting to python because with pyspark we cannot store values in csv file

df\_pandas = df\_joined.toPandas()

Cell 8 :#writhing to csv file

import os

# Ensure the directory exists

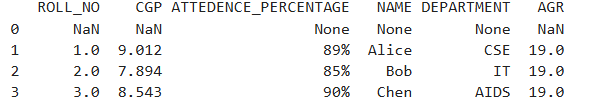
os.makedirs("D:/csv\_file", exist\_ok=True)

# Now save the DataFrame

df\_pandas.to\_csv("D:/csv\_file/student\_report\_single.csv", index=False, header=True)

print(df\_pandas) # Prints the whole DataFrame

output :



-The joined values written to student\_report csv file and mysql database.

-You can check in mysql workbench and csv file that given in d drive.

- we have done **collect** data from two structure data and **transform** them into into single source data and **load** to two different structured data.

How to run this code in cmd :

-In notebook click **file** and **choose Save and Export notebook as** and click **executable py.**

-The file is downloaded and remove emoj or install pip commands from text file.

-Open your cmd and type commands as :

Before executing the script

**-pip install pyspark**

**-pip install mysql-connector-python**

**-chcp 65001** (This switches the console to UTF-8 encoding (65001) which supports emojis.)

**-pip install pandas**

-Now execute your script like

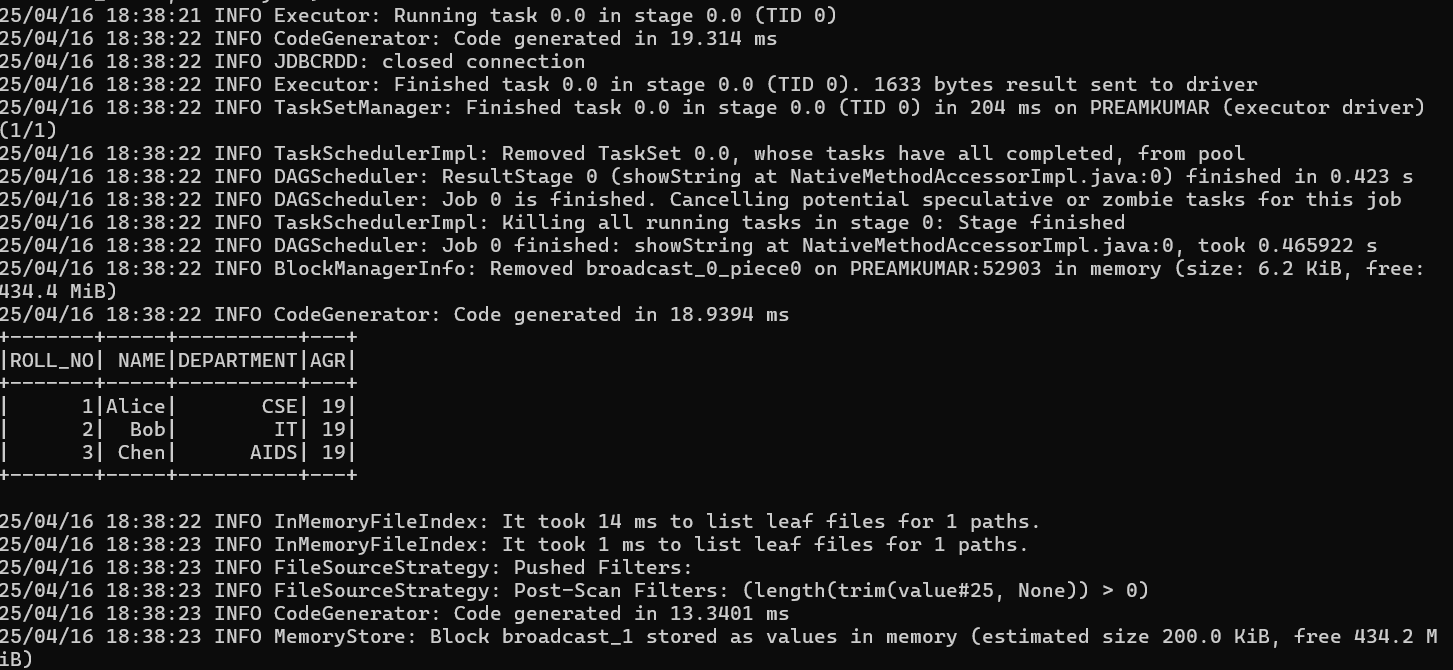
**spark-submit --jars "C:\path\to\mysql-connector-j-9.2.0.jar" C:\path\to\your\_script.py**

or

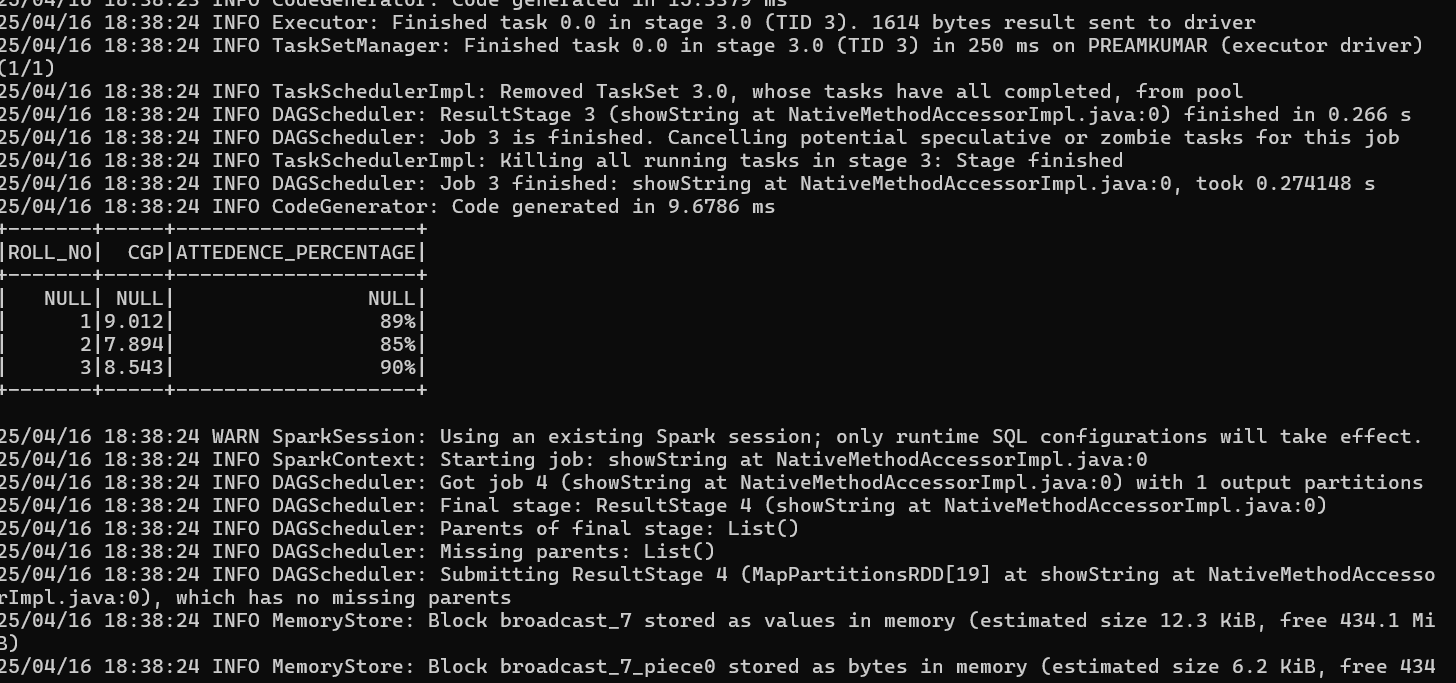
**spark-submit --jars "C:\\path\\to\\mysql-connector-j-9.2.0.jar" C:\\path\\to\\your\_script.py**

Output of cmd :

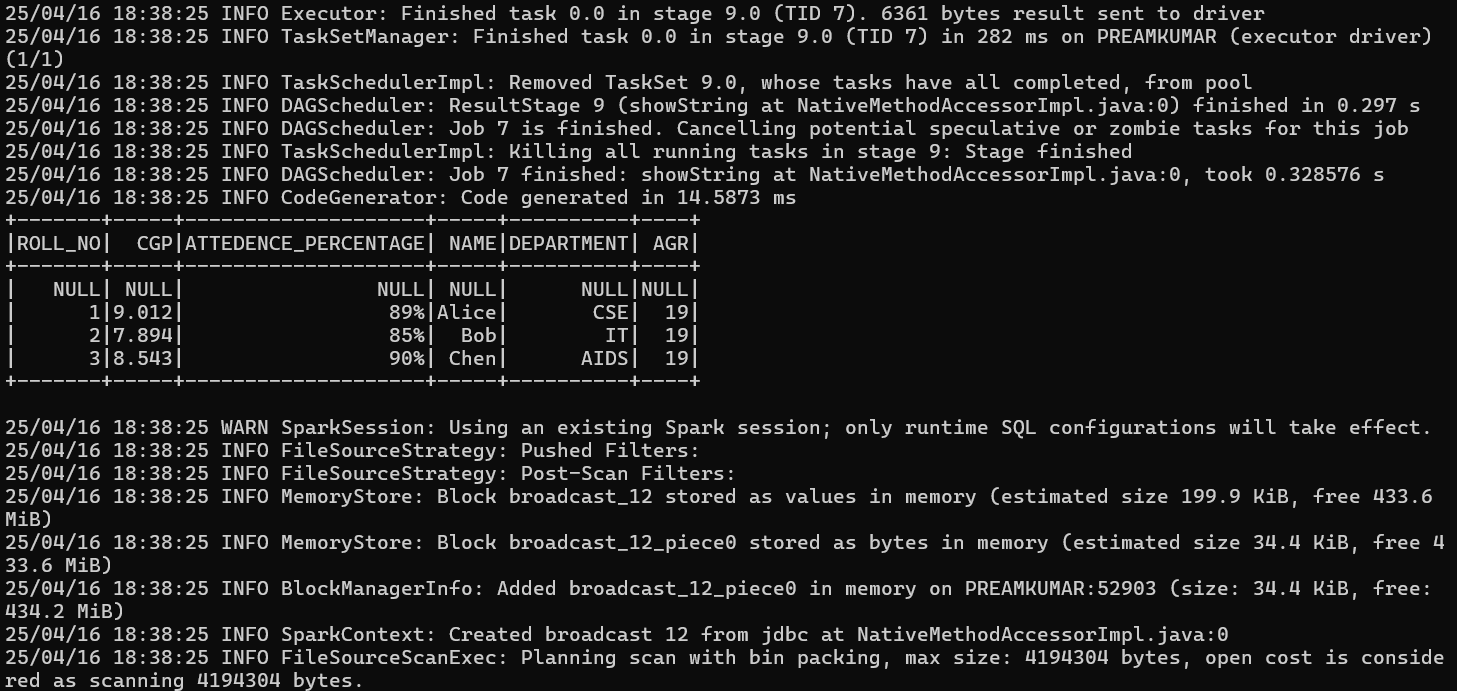
MySql :



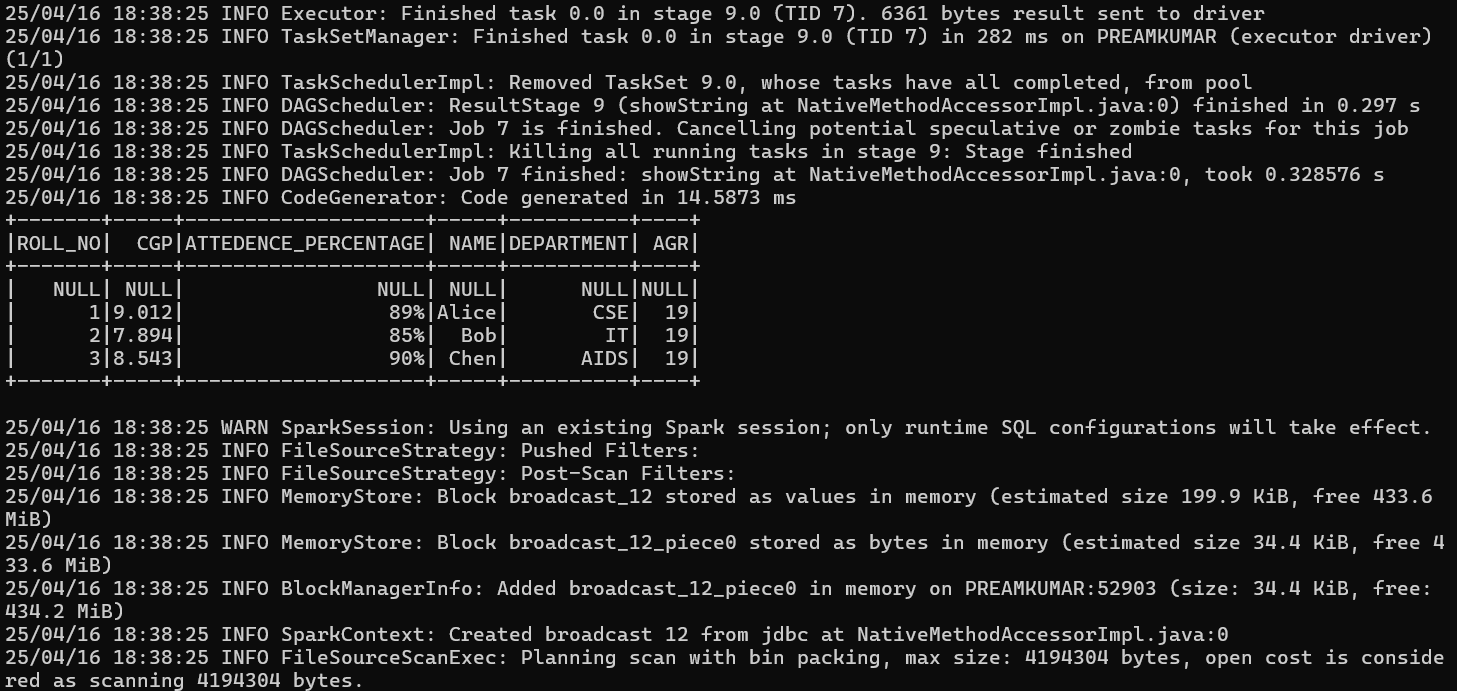
CSV file:



Joined data :



Mysql joined values :



CSV Joined values :

