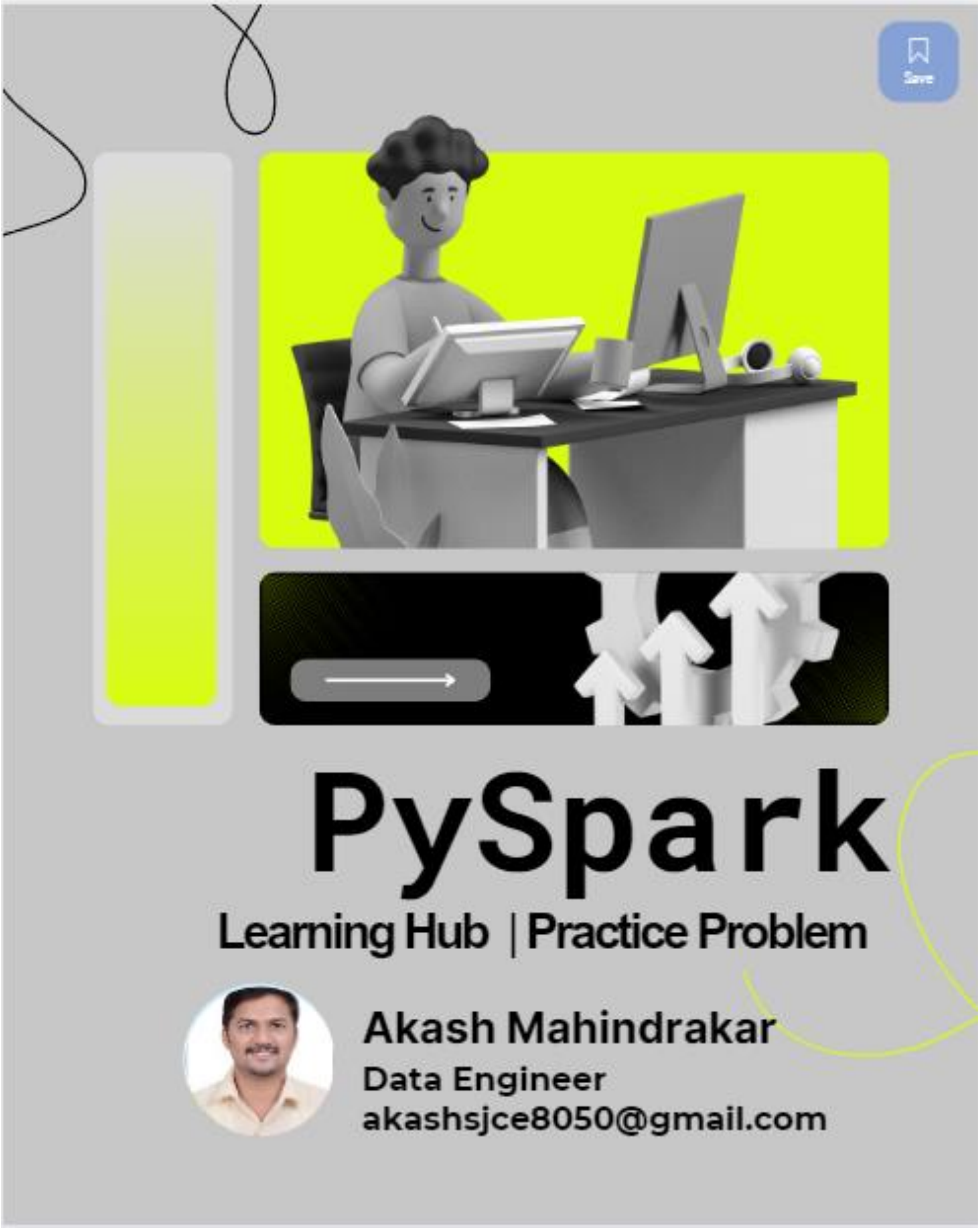
PYSPARK LEARNING HUB : DAY - 54



PySpark

Learning Hub | Practice Problem

Akash Mahindrakar
Data Engineer
akashsjce8050@gmail.com

WWW.LINKEDIN.COM/IN/AKASHMAHINDRAKAR

Step - 1 : Problem Statement

54_Students And Examinations Problem

Write an Pyspark code to find the number of times each student attended each exam.Order the result table by student_id and subject_name.

Difficult Level : EASY

DataFrame:

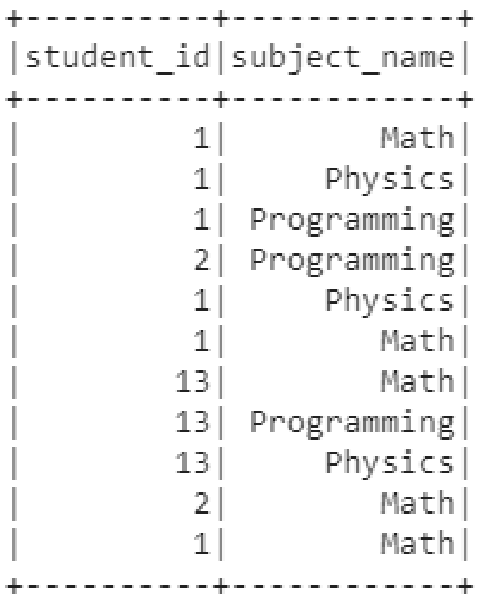
```
# Define the schema for the Examinations table
examinations_schema = StructType([
      StructField("student_id", IntegerType(), True),
      StructField("subject_name", StringType(), True)
])

# Data for the Examinations table
examinations_data = [
      (1, "Math"),
      (1, "Physics"),
      (1, "Programming"),
      (2, "Programming"),
      (1, "Physics"),
      (1, "Math"),
      (13, "Math"),
      (13, "Programming"),
      (13, "Physics"),
      (2, "Math"),
      (1, "Math")
]
 # Define the schema for the Students table
```

```
students_schema = StructType([
      StructField("student_id", IntegerType(), True),
      StructField("student_name", StringType(), True)
])


# Data for the Students table
students_data = [
      (1, "Alice"),
      (2, "Bob"),
      (13, "John"),
      (6, "Alex")
]

# Define the schema for the Subjects table
subjects_schema = StructType([
      StructField("subject_name", StringType(), True)
])

# Data for the Subjects table
subjects_data = [
      ("Math",),
      ("Physics",),
      ("Programming",)
]
```

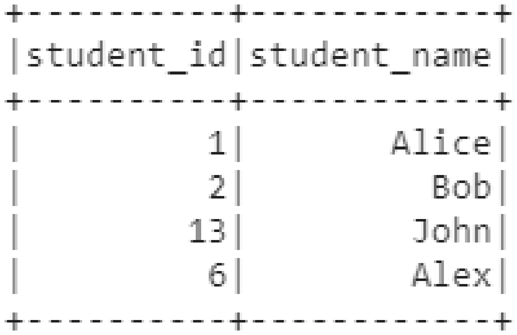
Step - 2 : Identifying The Input Data And Expected Output

PYSPARK LEARNING HUB : DAY - 54

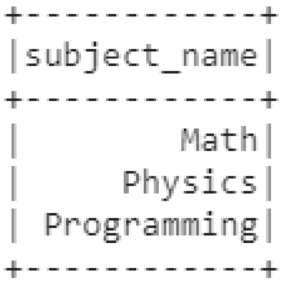
INPUT



```
+----------+------------+
|student_id|subject_name|
+----------+------------+
|         1|        Math|
|         1|     Physics|
|         1| Programming|
|         2| Programming|
|         1|     Physics|
|         1|        Math|
|        13|        Math|
|        13| Programming|
|        13|     Physics|
|         2|        Math|
|         1|        Math|
+----------+------------+
```

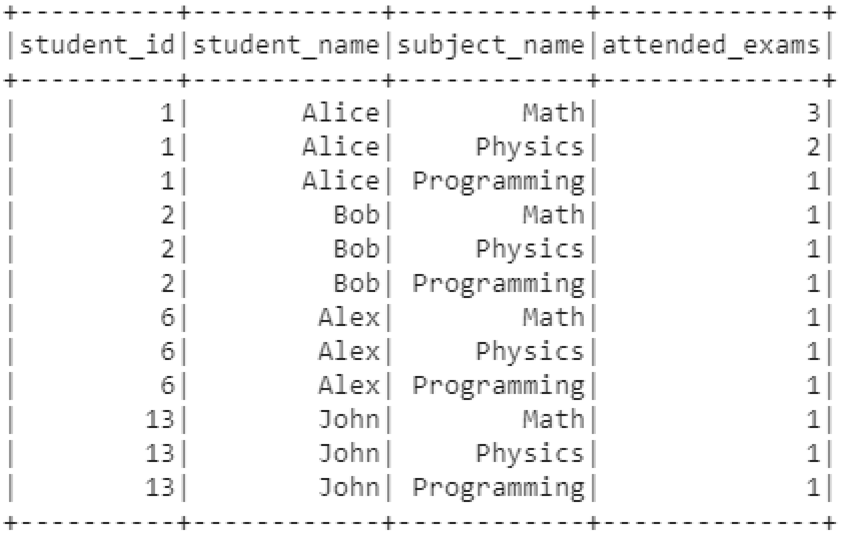


```
+----------+------------+
|student_id|student_name|
+----------+------------+
|         1|       Alice|
|         2|         Bob|
|        13|        John|
|         6|        Alex|
+----------+------------+
```



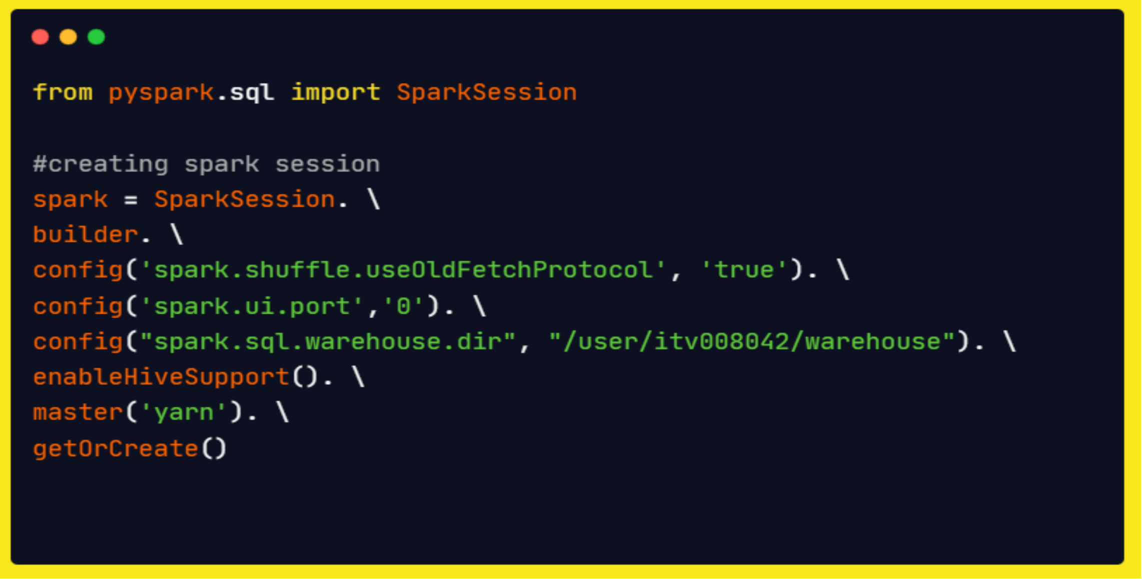
```
+------------+
|subject_name|
+------------+
|        Math|
|     Physics|
| Programming|
+------------+
```

OUTPUT

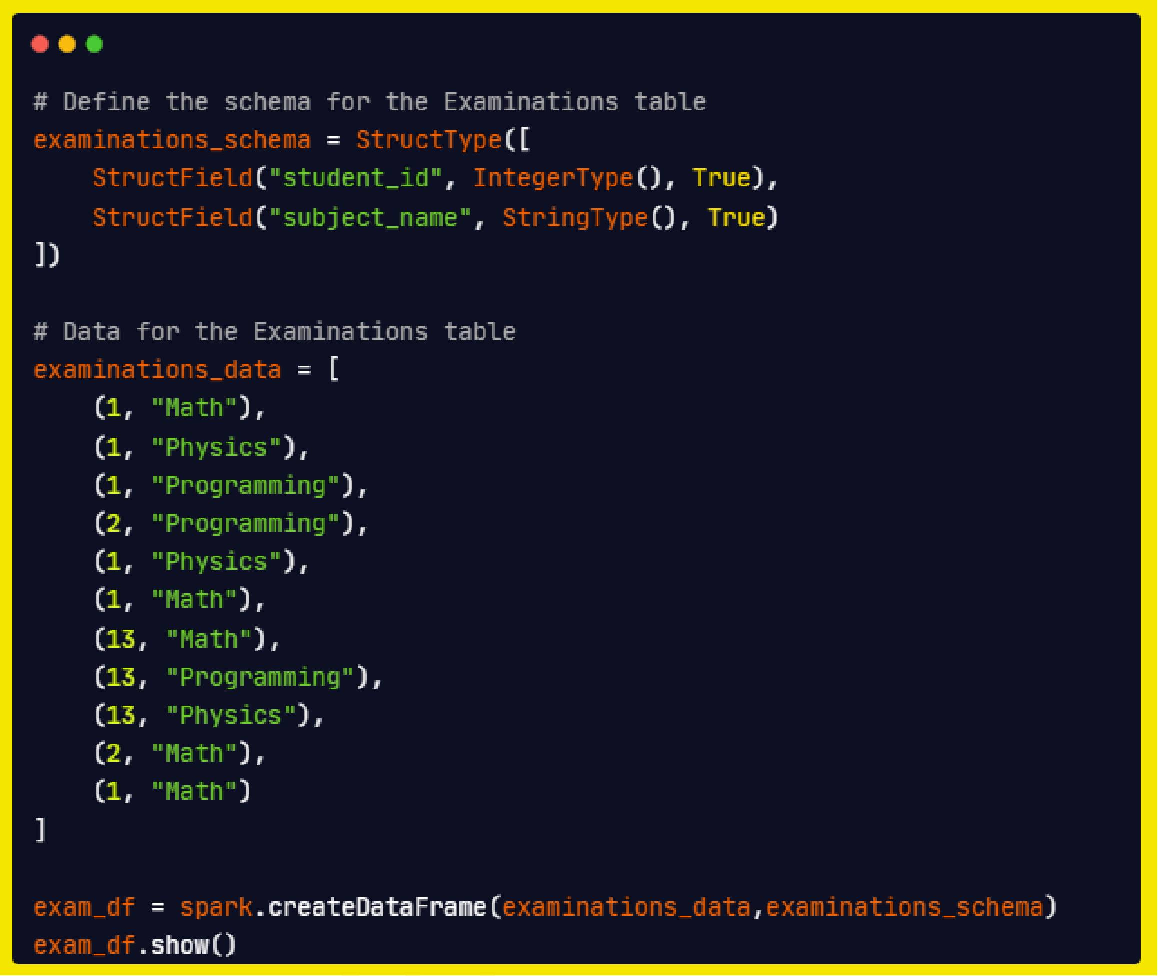


```
+----------+------------+------------+--------------+
|student_id|student_name|subject_name|attended_exams|
+----------+------------+------------+--------------+
|         1|       Alice|        Math|             3|
|         1|       Alice|     Physics|             2|
|         1|       Alice| Programming|             1|
|         2|         Bob|        Math|             1|
|         2|         Bob|     Physics|             1|
|         2|         Bob| Programming|             1|
|         6|        Alex|        Math|             1|
|         6|        Alex|     Physics|             1|
|         6|        Alex| Programming|             1|
|        13|        John|        Math|             1|
|        13|        John|     Physics|             1|
|        13|        John| Programming|             1|
+----------+------------+------------+--------------+
```

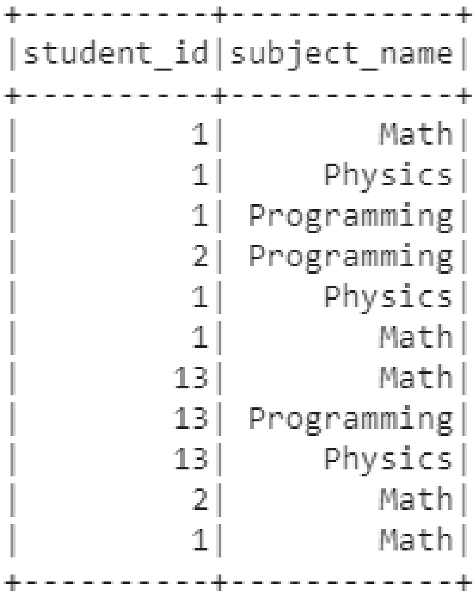
Step - 3 : Writing the pyspark code to solve the



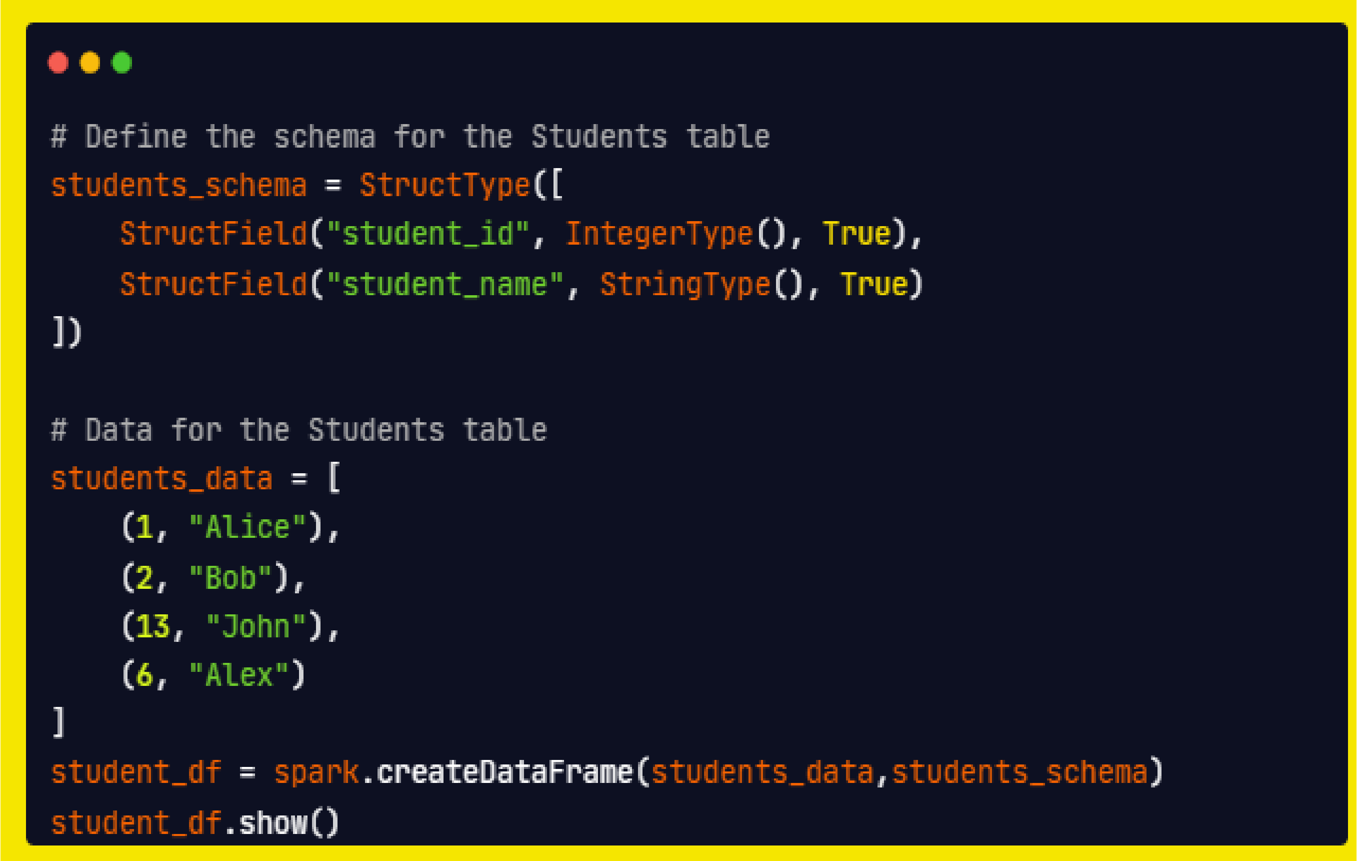
```
from pyspark.sql import SparkSession

#creating spark session
spark = SparkSession. \
builder. \
config('spark.shuffle.useOldFetchProtocol', 'true'). \
config('spark.ui.port','0'). \
config("spark.sql.warehouse.dir", "/user/itv008042/warehouse"). \
enableHiveSupport(). \
master('yarn'). \
getOrCreate()
```



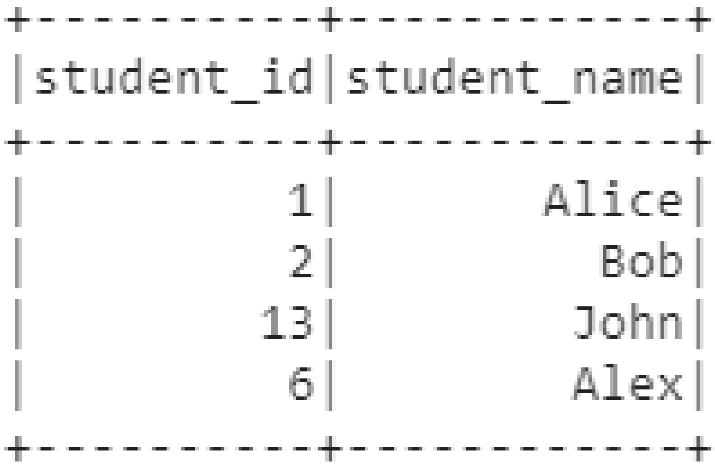
```
# Define the schema for the Examinations table
examinations_schema = StructType([
    StructField("student_id", IntegerType(), True),
    StructField("subject_name", StringType(), True)
])

# Data for the Examinations table
examinations_data = [
    (1, "Math"),
    (1, "Physics"),
    (1, "Programming"),
    (2, "Programming"),
    (1, "Physics"),
    (1, "Math"),
    (13, "Math"),
    (13, "Programming"),
    (13, "Physics"),
    (2, "Math"),
    (1, "Math")
]

exam_df = spark.createDataFrame(examinations_data,examinations_schema)
exam_df.show()
```



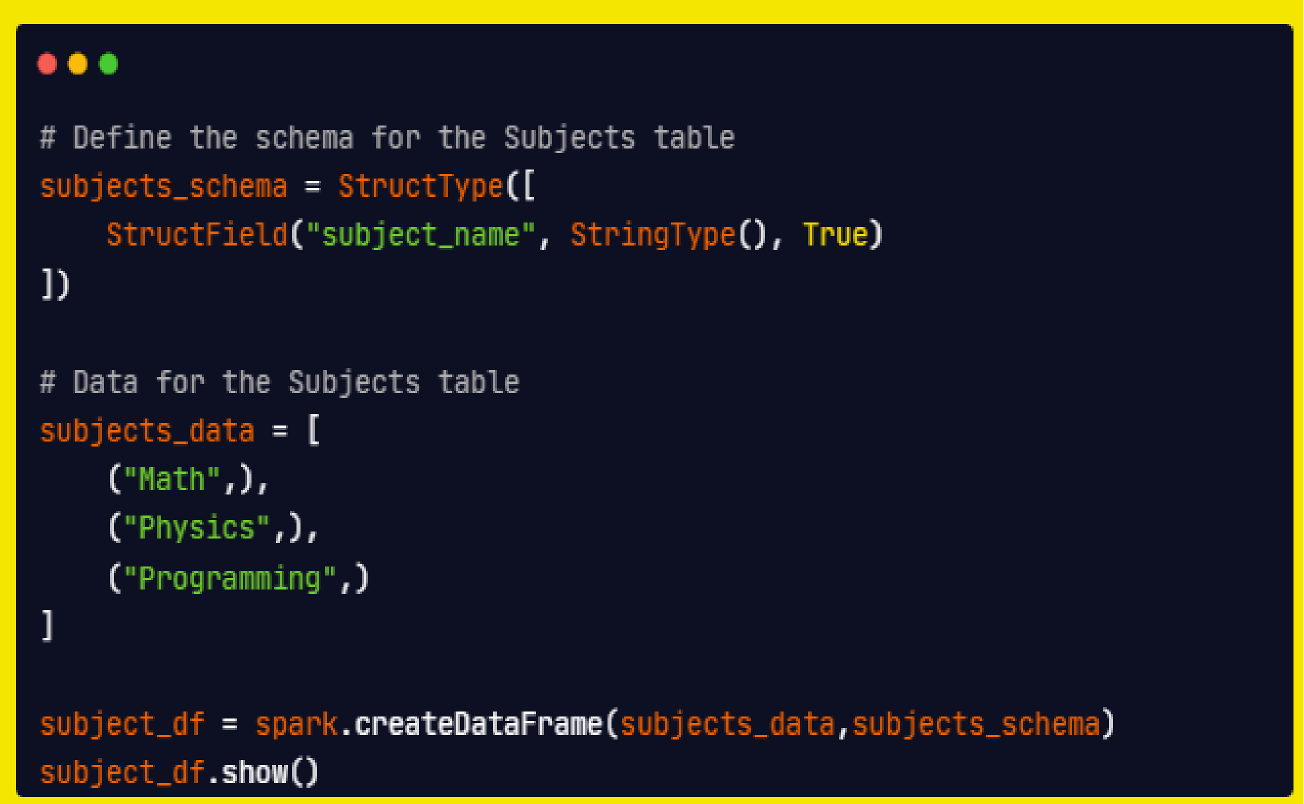
```
+----------+------------+
|student_id|subject_name|
+----------+------------+
|         1|        Math|
|         1|     Physics|
|         1| Programming|
|         2| Programming|
|         1|     Physics|
|         1|        Math|
|        13|        Math|
|        13| Programming|
|        13|     Physics|
|         2|        Math|
|         1|        Math|
+----------+------------+
```



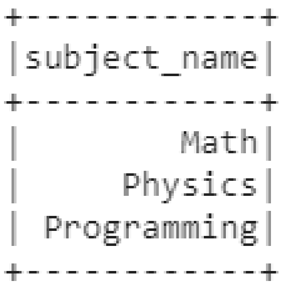
```python
# Define the schema for the Students table
students_schema = StructType([
    StructField("student_id", IntegerType(), True),
    StructField("student_name", StringType(), True)
])

# Data for the Students table
students_data = [
    (1, "Alice"),
    (2, "Bob"),
    (13, "John"),
    (6, "Alex")
]
student_df = spark.createDataFrame(students_data,students_schema)
student_df.show()
```



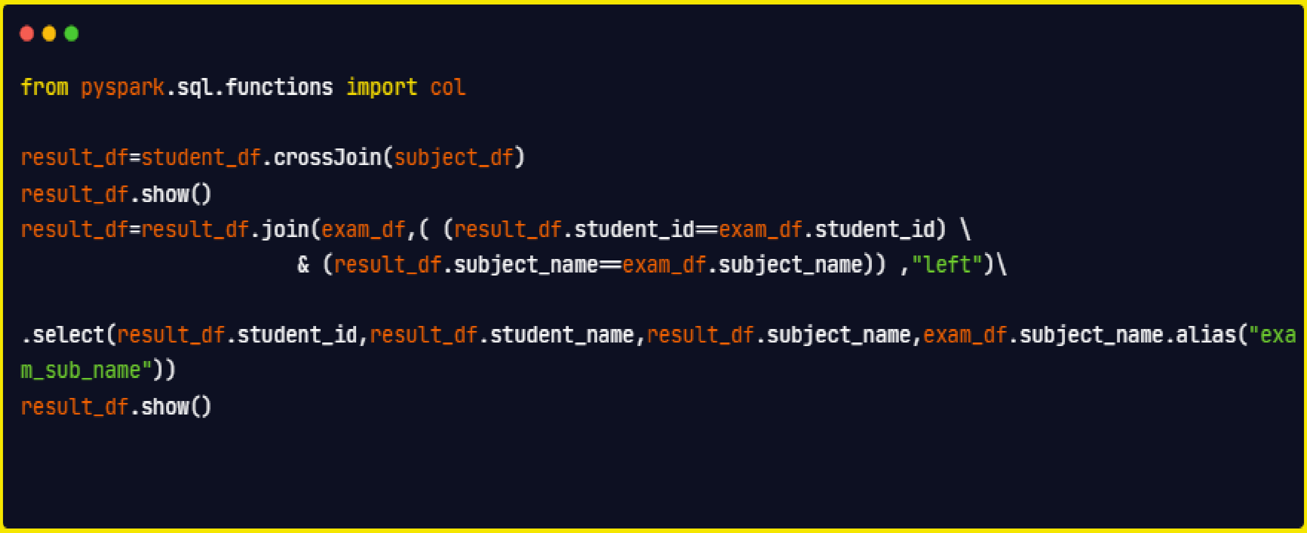
```
+----------+------------+
|student_id|student_name|
+----------+------------+
|         1|       Alice|
|         2|         Bob|
|        13|        John|
|         6|        Alex|
+----------+------------+
```



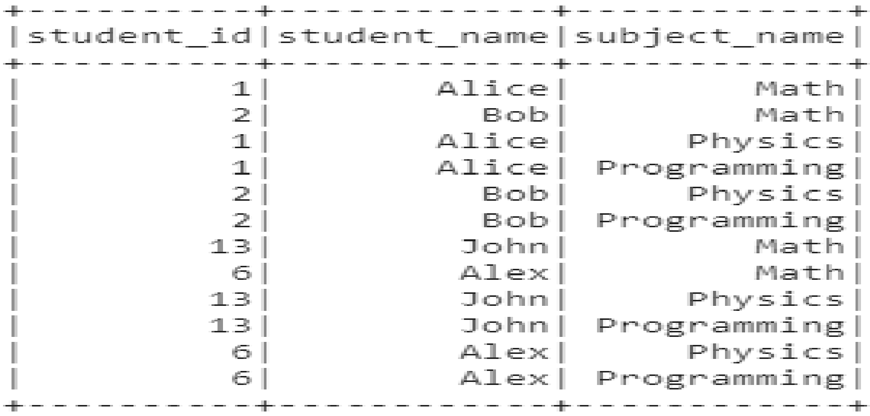
```
# Define the schema for the Subjects table
subjects_schema = StructType([
    StructField("subject_name", StringType(), True)
])

# Data for the Subjects table
subjects_data = [
    ("Math",),
    ("Physics",),
    ("Programming",)
]

subject_df = spark.createDataFrame(subjects_data,subjects_schema)
subject_df.show()
```



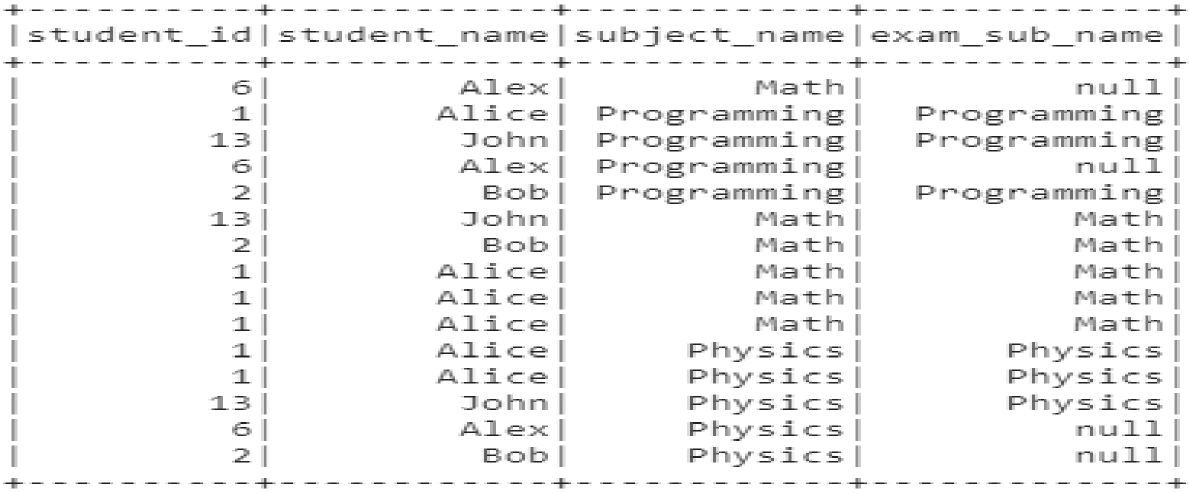
```
+------------+
|subject_name|
+------------+
|        Math|
|     Physics|
| Programming|
+------------+
```



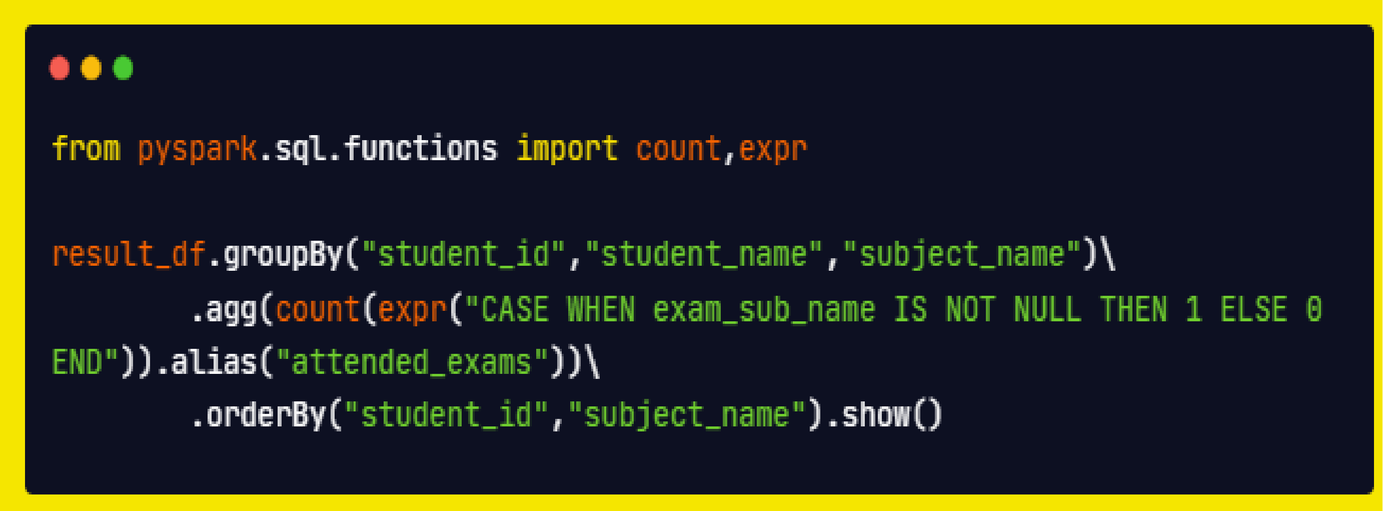
```
from pyspark.sql.functions import col

result_df=student_df.crossJoin(subject_df)
result_df.show()
result_df=result_df.join(exam_df,( (result_df.student_id==exam_df.student_id) \
                        & (result_df.subject_name==exam_df.subject_name)) ,"left")\

.select(result_df.student_id,result_df.student_name,result_df.subject_name,exam_df.subject_name.alias("exa
m_sub_name"))
result_df.show()
```



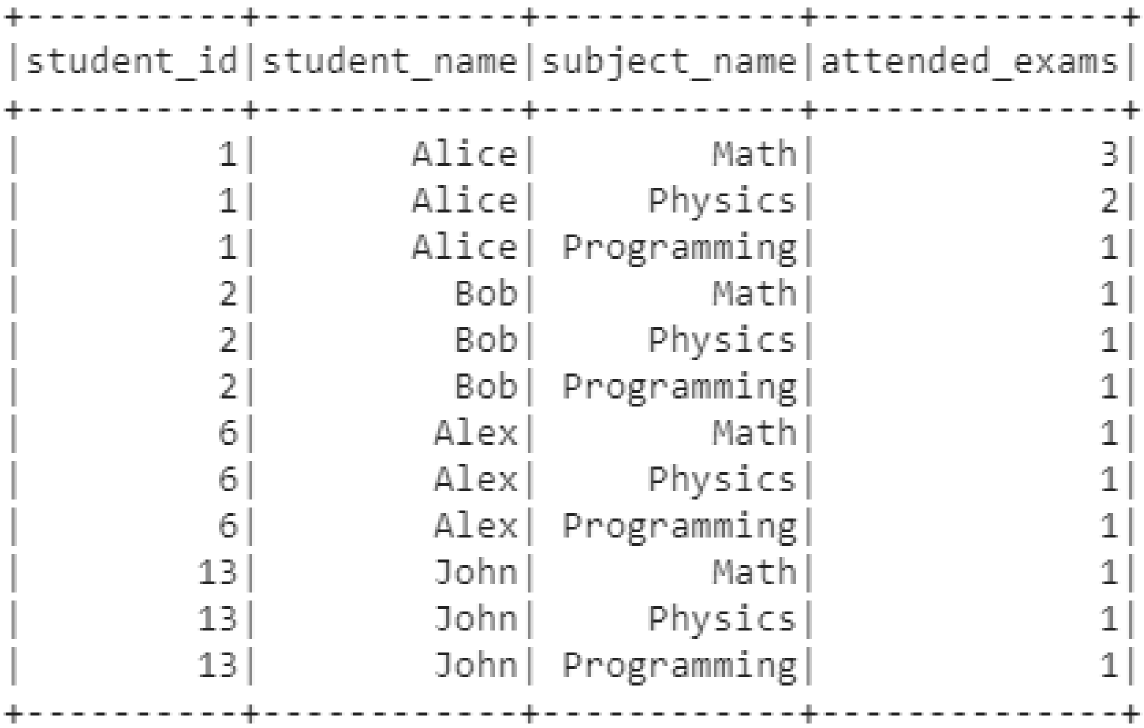
```
+----------+------------+------------+
|student_id|student_name|subject_name|
+----------+------------+------------+
|         1|       Alice|        Math|
|         2|         Bob|        Math|
|         1|       Alice|     Physics|
|         1|       Alice| Programming|
|         2|         Bob|     Physics|
|         2|         Bob| Programming|
|        13|        John|        Math|
|         6|        Alex|        Math|
|        13|        John|     Physics|
|        13|        John| Programming|
|         6|        Alex|     Physics|
|         6|        Alex| Programming|
+----------+------------+------------+
```



```
+----------+------------+------------+-------------+
|student_id|student_name|subject_name|exam_sub_name|
+----------+------------+------------+-------------+
|         6|        Alex|        Math|         null|
|         1|       Alice| Programming|  Programming|
|        13|        John| Programming|  Programming|
|         6|        Alex| Programming|         null|
|         2|         Bob| Programming|  Programming|
|        13|        John|        Math|         Math|
|         2|         Bob|        Math|         Math|
|         1|       Alice|        Math|         Math|
|         1|       Alice|        Math|         Math|
|         1|       Alice|        Math|         Math|
|         1|       Alice|     Physics|      Physics|
|         1|       Alice|     Physics|      Physics|
|        13|        John|     Physics|      Physics|
|         6|        Alex|     Physics|         null|
|         2|         Bob|     Physics|         null|
+----------+------------+------------+-------------+
```



```
from pyspark.sql.functions import count,expr

result_df.groupBy("student_id","student_name","subject_name")\
        .agg(count(expr("CASE WHEN exam_sub_name IS NOT NULL THEN 1 ELSE 0
END")).alias("attended_exams"))\
        .orderBy("student_id","subject_name").show()
```



```
+----------+------------+------------+--------------+
|student_id|student_name|subject_name|attended_exams|
+----------+------------+------------+--------------+
|         1|       Alice|        Math|             3|
|         1|       Alice|     Physics|             2|
|         1|       Alice| Programming|             1|
|         2|         Bob|        Math|             1|
|         2|         Bob|     Physics|             1|
|         2|         Bob| Programming|             1|
|         6|        Alex|        Math|             1|
|         6|        Alex|     Physics|             1|
|         6|        Alex| Programming|             1|
|        13|        John|        Math|             1|
|        13|        John|     Physics|             1|
|        13|        John| Programming|             1|
+----------+------------+------------+--------------+
```

Was it helpful?

follow for more!

Akash Mahindrakar
Data Engineer
akashsjce8050@gmail.com

SHARE YOUR THOUGHTS IN COMMENT BELOW