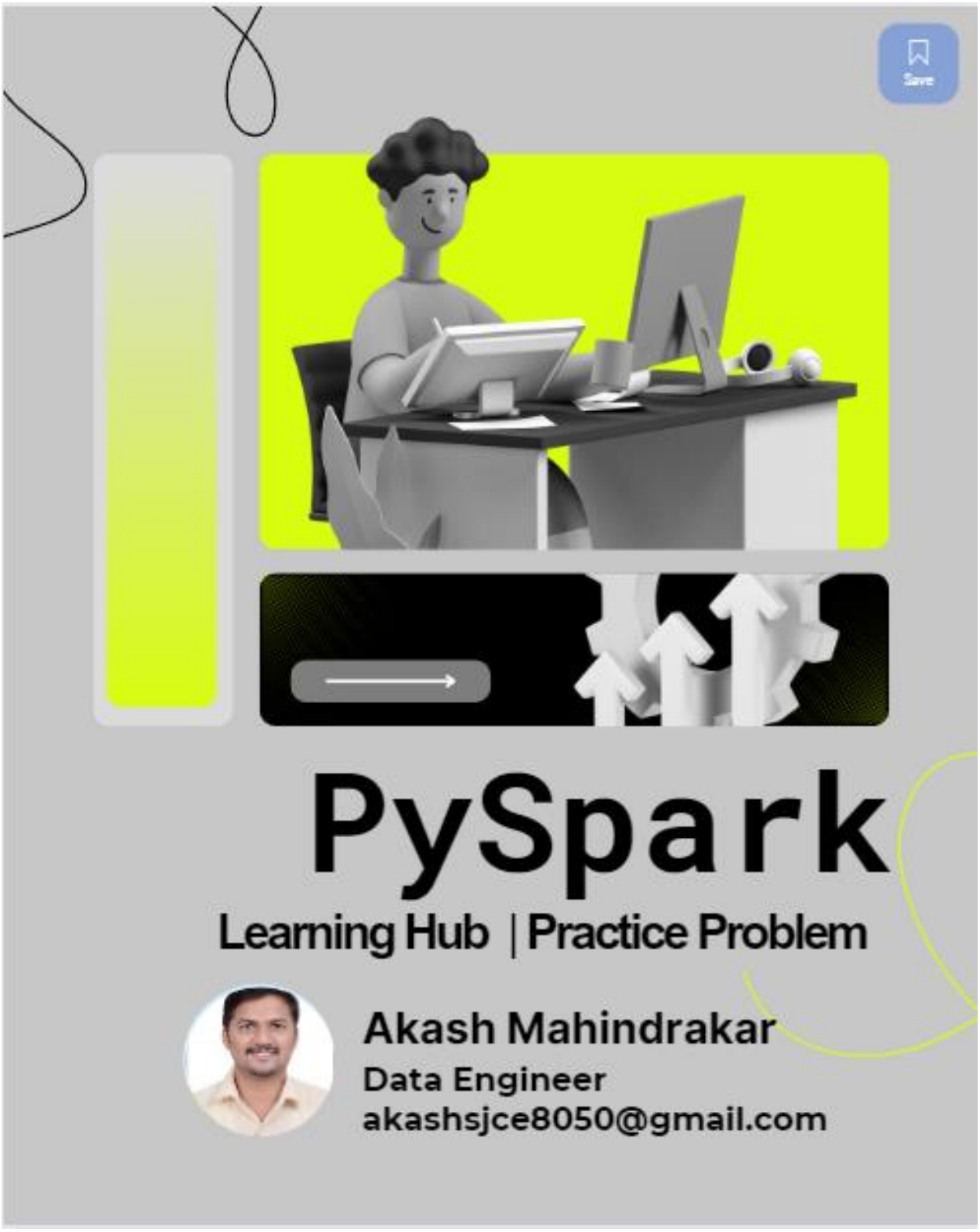
PYSPARK LEARNING HUB : DAY - 27



Step - 1 : Problem Statement

27_groupby in pyspark

Write a pyspark code perform below function

- 46. Get department wise average salary from "EmployeeDetail" table order by salary ascending
- 47. Get department wise maximum salary from "EmployeeDetail" table order by salary ascending
- 48. Get department wise minimum salary from "EmployeeDetail" table order by salary ascending

Difficult Level : EASY

DataFrame:

```
data = [
        [1, "Vikas", "Ahlawat", 600000.0, "2013-02-15 11:16:28.290", "IT", "Male"],
        [2, "nikita", "Jain", 530000.0, "2014-01-09 17:31:07.793", "HR", "Female"],
        [3, "Ashish", "Kumar", 1000000.0, "2014-01-09 10:05:07.793", "IT", "Male"],
        [4, "Nikhil", "Sharma", 480000.0, "2014-01-09 09:00:07.793", "HR", "Male"],
        [5, "anish", "kadian", 500000.0, "2014-01-09 09:31:07.793", "Payroll", "Male"],
]


# Create a schema for the DataFrame
schema = StructType([
        StructField("EmployeeID", IntegerType(), True),
        StructField("First_Name", StringType(), True),
        StructField("Last_Name", StringType(), True),
        StructField("Salary", DoubleType(), True),
        StructField("Joining_Date", StringType(), True),
        StructField("Department", StringType(), True),
        StructField("Gender", StringType(), True)
])
```

Step - 2 : Writing the pyspark code to solve the

```
# Creating Spark Session
from pyspark.sql import SparkSession
from pyspark.sql.types import
StructType,StructField,IntegerType,StringType

#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()

# Create a list of rows from the image
data = [
        [1, "Vikas", "Ahlawat", 600000.0, "2013-02-15 11:16:28.290", "IT", "Male"],
        [2, "nikita", "Jain", 530000.0, "2014-01-09 17:31:07.793", "HR", "Female"],
        [3, "Ashish", "Kumar", 1000000.0, "2014-01-09 10:05:07.793", "IT", "Male"],
        [4, "Nikhil", "Sharma", 480000.0, "2014-01-09 09:00:07.793", "HR", "Male"],
        [5, "anish", "kadian", 500000.0, "2014-01-09 09:31:07.793", "Payroll", "Male"],
]

# Create a schema for the DataFrame
schema = StructType([
        StructField("EmployeeID", IntegerType(), True),
        StructField("First_Name", StringType(), True),
        StructField("Last_Name", StringType(), True),
        StructField("Salary", DoubleType(), True),
        StructField("Joining_Date", StringType(), True),
        StructField("Department", StringType(), True),
```

```
        StructField("Gender", StringType(), True)
])

emp_df=spark.createDataFrame(data,schema)
```

```
+----------+----------+---------+---------+--------------------+----------+------+
|EmployeeID|First_Name|Last_Name|   Salary|        Joining_Date|Department|Gender|
+----------+----------+---------+---------+--------------------+----------+------+
|         1|     Vikas|  Ahlawat| 600000.0|2013-02-15 11:16:...|        IT|  Male|
|         2|    nikita|     Jain| 530000.0|2014-01-09 17:31:...|        HR|Female|
|         3|    Ashish|    Kumar|1000000.0|2014-01-09 10:05:...|        IT|  Male|
|         4|    Nikhil|   Sharma| 480000.0|2014-01-09 09:00:...|        HR|  Male|
|         5|     anish|   kadian| 500000.0|2014-01-09 09:31:...|   Payroll|  Male|
+----------+----------+---------+---------+--------------------+----------+------+
```

```python
# 46. Get department wise average salary from
"EmployeeDetail" table order by salary ascending
from pyspark.sql.functions import avg

emp_df.groupby("Department")\
      .agg(avg(col('Salary'))).show()
```

```
+----------+-----------+
|Department|avg(Salary)|
+----------+-----------+
|        HR|   505000.0|
|   Payroll|   500000.0|
|        IT|   800000.0|
+----------+-----------+
```

PYSPARK LEARNING HUB : DAY - 27

```python
# 47. Get department wise maximum salary from "EmployeeDetail" table order by salary
# ascending
from pyspark.sql.functions import max
emp_df.groupby("Department")\
      .agg(max(col('Salary')).alias("max_salary"))\
      .orderBy(col('max_salary').asc()).show()

# 48. Get department wise minimum salary from "EmployeeDetail" table order by
# salary ascending
from pyspark.sql.functions import min
emp_df.groupby("Department")\
    .agg(min(col('Salary')).alias("max_salary"))\
    .orderBy(col('max_salary').asc()).show()
```

```
+----------+----------+
|Department|max_salary|
+----------+----------+
|   Payroll|  500000.0|
|        HR|  530000.0|
|        IT| 1000000.0|
+----------+----------+

+----------+----------+
|Department|max_salary|
+----------+----------+
|        HR|  480000.0|
|   Payroll|  500000.0|
|        IT|  600000.0|
+----------+----------+
```

PYSPARK LEARNING HUB : DAY - 27

Save

Was it helpful?

follow for more!

Akash Mahindrakar
Data Engineer
akashsjce8050@gmail.com

Comment

SHARE YOUR THOUGHTS IN COMMENT BELOW

Share