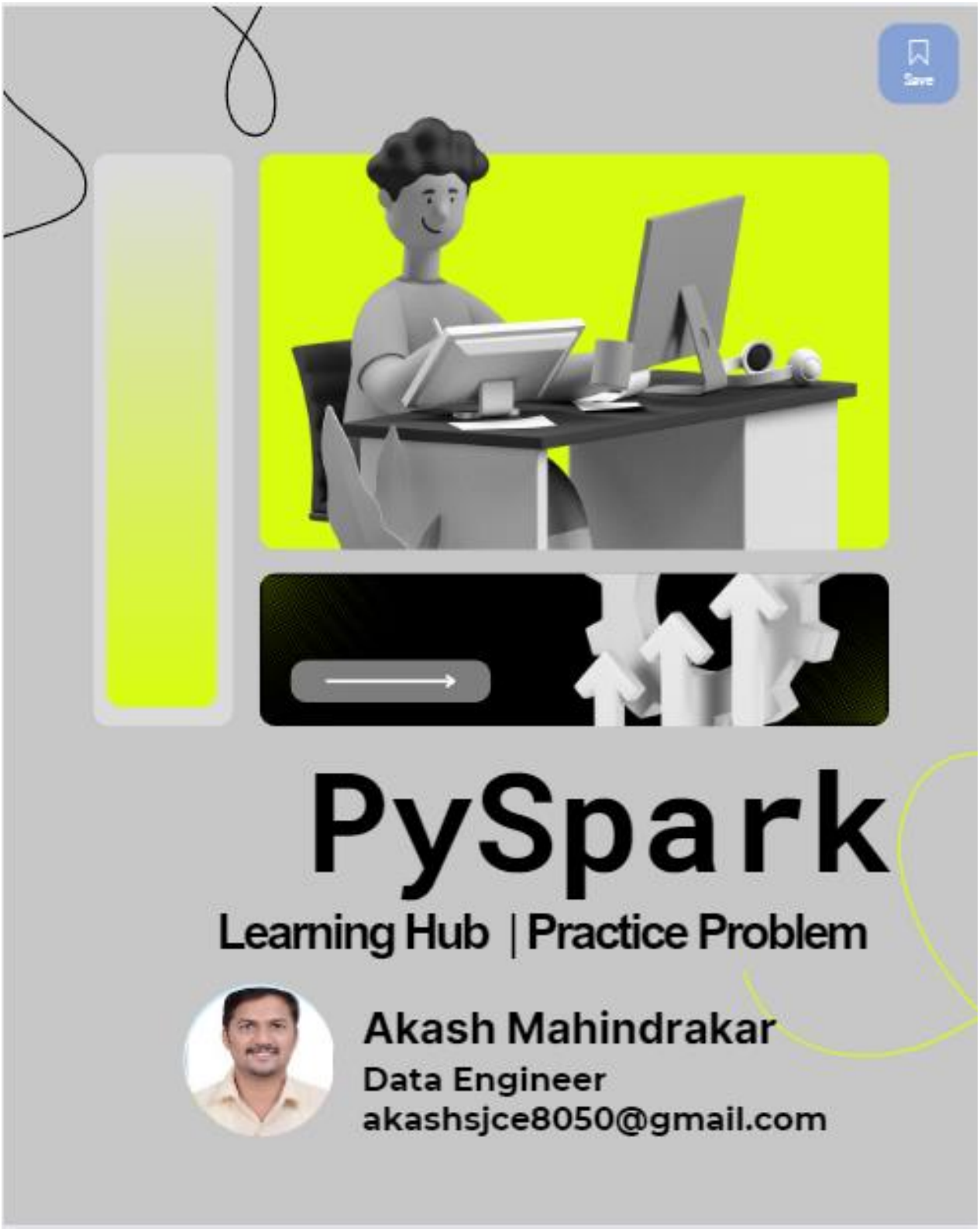
PYSPARK LEARNING HUB : DAY - 17



Step - 1 : Problem Statement

17_Select in pyspark

Write a pyspark code perform below function

- Write a pyspark code for combine FirstName and LastName and display it as "Name" (also include white space between first name & last name)
- Select employee detail whose name is "Vikas"
- Get all employee detail from EmployeeDetail table whose "FirstName" start with letter 'a'.

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

```
# 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)
```

#1. Write a pyspark code for combine FirstName and LastName and display it as "Name" (also include white space between first name & last name)

```
from pyspark.sql.functions import concat_ws
emp_df.select(concat_ws(" ","First_Name","Last_Name")\
        .alias("Name")).show()
```

```
+--------------+
|          Name|
+--------------+
|Vikas Ahlawat|
|  nikita Jain|
| Ashish Kumar|
|Nikhil Sharma|
| anish kadian|
+--------------+
```

2. Select employee detail whose name is "Vikas"

```
# Methos 1
from pyspark.sql.functions import col
emp_df.filter(col("First_Name") == 'Vikas' ).show(truncate=False)

# Methos 2
emp_df.filter(emp_df.First_Name == 'Vikas'  ).show(truncate=False)

# Methos 3
emp_df.filter(emp_df['First_Name'] == 'Vikas'  ).show(truncate=False)


# Methos 4
emp_df.where(emp_df['First_Name'] == 'Vikas'  ).show(truncate=False)
```

EmployeeID	First_Name	Last_Name	Salary	Joining_Date	Department	Gender
1	Vikas	Ahlawat	600000.0	2013-02-15 11:16:28.290	IT	Male

EmployeeID	First_Name	Last_Name	Salary	Joining_Date	Department	Gender
1	Vikas	Ahlawat	600000.0	2013-02-15 11:16:28.290	IT	Male

EmployeeID	First_Name	Last_Name	Salary	Joining_Date	Department	Gender
1	Vikas	Ahlawat	600000.0	2013-02-15 11:16:28.290	IT	Male

EmployeeID	First_Name	Last_Name	Salary	Joining_Date	Department	Gender
1	Vikas	Ahlawat	600000.0	2013-02-15 11:16:28.290	IT	Male

- Get all employee detail from EmployeeDetail table whose "FirstName" start with letter 'a'.

```
# Method 1
from pyspark.sql.functions import lower
emp_df.filter(lower(emp_df['First_Name']).like("a%")).show()

# Method 2

emp_df.filter((emp_df['First_Name'].like("a%")) |
(emp_df['First_Name'].like("A%")) ).show()
```

```
+----------+----------+---------+---------+-------------------+----------+------+
|EmployeeID|First_Name|Last_Name|   Salary|       Joining_Date|Department|Gender|
+----------+----------+---------+---------+-------------------+----------+------+
|         3|    Ashish|    Kumar|1000000.0|2014-01-09 10:05:...|        IT|  Male|
|         5|     anish|   kadian| 500000.0|2014-01-09 09:31:...|   Payroll|  Male|
+----------+----------+---------+---------+-------------------+----------+------+

+----------+----------+---------+---------+-------------------+----------+------+
|EmployeeID|First_Name|Last_Name|   Salary|       Joining_Date|Department|Gender|
+----------+----------+---------+---------+-------------------+----------+------+
|         3|    Ashish|    Kumar|1000000.0|2014-01-09 10:05:...|        IT|  Male|
|         5|     anish|   kadian| 500000.0|2014-01-09 09:31:...|   Payroll|  Male|
+----------+----------+---------+---------+-------------------+----------+------+
```

PYSPARK LEARNING HUB : DAY - 17

Save

Was it helpful?

follow for more!

Akash Mahindrakar
Data Engineer
akashsjce8050@gmail.com

Comment

SHARE YOUR THOUGHTS IN COMMENT BELOW

Share