



PySpark
Learning Hub | Practice Problem



Akash Mahindrakar
Data Engineer
akashsjce8050@gmail.com

Step - 1 : Problem Statement

23_Date in pyspark

Write a pyspark code perform below function

- Get all employee details from EmployeeDetail table whose joining month is Jan(1).
- Get all employee details from EmployeeDetail table whose joining date between "2013-01-01" and "2013-12-01".
- Get how many employee exist in "EmployeeDetail" table.
- Select all employee detail with First name "Vikas","Ashish", and "Nikhil".

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)
])
```

PYSPARK LEARNING HUB : DAY - 23

```
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

```
# 28). Get all employee details from EmployeeDetail table whose joining month is  
# Jan(1).
```

```
from pyspark.sql.functions import month  
emp_df.filter(month(col("Joining_Date"))= 1) .show()
```

```
# 29). Get all employee details from EmployeeDetail table whose joining  
# date between "2013-01-01" and "2013-12-01".
```

```
emp_df.filter(col("Joining_Date").between("2013-01-01","2013-12-01")).show()
```

EmployeeID	First_Name	Last_Name	Salary	Joining_Date	Department	Gender
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

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

PYSPARK LEARNING HUB : DAY - 23

```
# 30). Get how many employee exist in "EmployeeDetail" table.
```

```
emp_df.count()
```

```
# 32. Select all employee detail with First name "Vikas", "Ashish", and "Nikhil".
```

```
from pyspark.sql.functions import lower
emp_df.filter(lower(col("First_Name"))\
              .isin("vikas", "ashish", "nikhil")).show()
```

Number of employees: 5

EmployeeID	First_Name	Last_Name	Salary	Joining_Date	Department	Gender
1	Vikas	Ahlawat	600000.0	2013-02-15 11:16:...	IT	Male
3	Ashish	Kumar	1000000.0	2014-01-09 10:05:...	IT	Male
4	Nikhil	Sharma	480000.0	2014-01-09 09:00:...	HR	Male



Save

**Was it
helpful?**
follow for more!



Akash Mahindrakar

Data Engineer

akashsjce8050@gmail.com



Comment

**SHARE YOUR THOUGHTS
IN COMMENT BELOW**



Share