Pyspark GroupBy And Aggregate Functions

In [1]:
```
from pyspark.sql import SparkSession
```

In [2]:
```
spark = SparkSession.builder.appName('Agg').getOrCreate()
```

In [3]:
```
spark
```

Out[3]:

SparkSession - in-memory

SparkContext

Spark UI

Version	`v3.3.0`
Master	`local[*]`
AppName	`Agg`

In [4]:
```
df_pyspark = spark.read.csv('test4.csv', header=True, inferSchema=True)
```

In [5]:
```
df_pyspark.show()
```

```
+-------+------------+------+
|   name| departments|salary|
+-------+------------+------+
|Ajinkya|Data Science| 10000|
|Ajinkya|         IOT|  5000|
| Mahesh|    Big Data|  4000|
|Ajinkya|    Big Data|  4000|
| Mahesh|Data Science|  3000|
| Nikhil|Data Science| 20000|
| Nikhil|         IOT| 10000|
| Nikhil|    Big Data|  5000|
|  Sunny|Data Science| 10000|
|  Sunny|    Big Data|  2000|
+-------+------------+------+
```

In [7]:
```
df_pyspark.printSchema()
```

```
root
 |--  name: string (nullable = true)
 |-- departments: string (nullable = true)
 |-- salary: integer (nullable = true)
```

In [38]:
```
## Groupby
from pyspark.sql.functions import col
from pyspark.sql.types import FloatType
df_pyspark.groupBy('name').agg({'salary':'mean'}).show()
```

```
+-------+------------------+
|   name|       avg(salary)|
+-------+------------------+
|Ajinkya| 6333.333333333333|
|  Sunny|            6000.0|
| Nikhil|11666.666666666666|
| Mahesh|            3500.0|
+-------+------------------+
```

In [99]:
```
df = df_pyspark.groupBy('name').agg({'salary':'mean'}).select("name",col('avg(salary)').cast('flo
df.show()
```

```
+-------+---------+
|   name|      sal|
+-------+---------+
|Ajinkya|6333.3335|
|  Sunny|   6000.0|
| Nikhil|11666.667|
| Mahesh|   3500.0|
+-------+---------+
```

In [108…
```
df.printSchema()
```

```
root
 |-- name: string (nullable = true)
 |-- sal: float (nullable = true)
```

In [113…
```
from pyspark.sql.functions import floor, col, round
df.withColumn('sal', round('sal',2)).show()
```

```
+-------+--------+
|   name|     sal|
+-------+--------+
|Ajinkya| 6333.33|
|  Sunny|  6000.0|
| Nikhil|11666.67|
| Mahesh|  3500.0|
+-------+--------+
```

In [8]:
```
### grouped to find the maximum salary
df_pyspark.groupBy('name').sum('salary').show()
```

```
+-------+-----------+
|   name|sum(salary)|
+-------+-----------+
|Ajinkya|      19000|
|  Sunny|      12000|
| Nikhil|      35000|
| Mahesh|       7000|
+-------+-----------+
```

In [10]:
```
### Gorup by Departments which give maximum salary to employee
df_pyspark.groupBy('departments').max().show()
```

```
+------------+-----------+
| departments|max(salary)|
+------------+-----------+
|         IOT|      10000|
|    Big Data|       5000|
|Data Science|      20000|
+------------+-----------+
```

In [25]:
```
### Goup by departments and find total salary department wise
from pyspark.sql.functions import col
df_pyspark.groupBy('departments').sum('salary').withColumnRenamed('sum(salary)', 'Total').sort(c
```

```
+------------+-----+
| departments|Total|
+------------+-----+
|Data Science|43000|
|         IOT|15000|
|    Big Data|15000|
+------------+-----+
```

In [5]:
```
### Mean Salary
df_pyspark.groupBy('departments').mean('salary').show()
```

```
+------------+-----------+
| departments|avg(salary)|
+------------+-----------+
|         IOT|     7500.0|
|    Big Data|     3750.0|
|Data Science|    10750.0|
+------------+-----------+
```

In [9]:
```
### Number of employees in each department
df_pyspark.groupBy('departments').count().show()
```

```
+------------+-----+
| departments|count|
+------------+-----+
|         IOT|    2|
|    Big Data|    4|
|Data Science|    4|
+------------+-----+
```

In [10]:
```
### Total expenditure
df_pyspark.agg({'salary':'sum'}).show()
```

```
+-----------+
|sum(salary)|
+-----------+
|      73000|
+-----------+
```

In [12]:
```
### Employee getting max salary
df_pyspark.groupBy('name').max().show()
```

```
+-------+-----------+
|   name|max(salary)|
+-------+-----------+
|Ajinkya|      10000|
|  Sunny|      10000|
| Nikhil|      20000|
| Mahesh|       4000|
+-------+-----------+
```

In [13]:
```
df_pyspark.groupBy('name').min().show()
```

```
+-------+-----------+
|   name|min(salary)|
+-------+-----------+
|Ajinkya|       4000|
|  Sunny|       2000|
| Nikhil|       5000|
| Mahesh|       3000|
+-------+-----------+
```

In [6]:
```
df_pyspark.groupBy('name').avg().show()
```

```
+-------+------------------+
|   name|       avg(salary)|
+-------+------------------+
|Ajinkya| 6333.333333333333|
|  Sunny|            6000.0|
| Nikhil|11666.666666666666|
| Mahesh|            3500.0|
+-------+------------------+
```

In [ ]:
```

```