Module 4 Spark

```
val rdd=spark.sparkContext.parallelize(Seq(("Java", 20000),
  ("Python", 100000), ("Scala", 3000)))
```

FINISHED

```
rdd: org.apache.spark.rdd.RDD[(String, Int)] = ParallelCollectionRDD[0] at parallelize at <console
>:23
```

Took 1 min 7 sec. Last updated by anonymous at October 06 2022, 9:28:41 AM.

```
rdd.collect()
```

FINISHED

```
res3: Array[(String, Int)] = Array((Java,20000), (Python,100000), (Scala,3000))
```

Took 0 sec. Last updated by anonymous at October 06 2022, 9:29:27 AM.

```
import spark.implicits._
val columns = Seq("language","users_count")
val data = Seq(("Java", "20000"), ("Python", "100000"), ("Scala", "3000"))
```

FINISHED

```
import spark.implicits._
columns: Seq[String] = List(language, users_count)
data: Seq[(String, String)] = List((Java,20000), (Python,100000), (Scala,3000))
```

Took 1 sec. Last updated by anonymous at October 06 2022, 9:50:32 AM.

```
val rdd = spark.sparkContext.parallelize(data)
```

FINISHED

```
rdd: org.apache.spark.rdd.RDD[(String, String)] = ParallelCollectionRDD[1] at parallelize at <conso
le>:28
```

Took 1 sec. Last updated by anonymous at October 06 2022, 9:50:41 AM.

```
val dfFromRDD1 = rdd.toDF(columns:_*)
dfFromRDD1.printSchema()
```

FINISHED

```
dfFromRDD1: org.apache.spark.sql.DataFrame = [_1: string, _2: string]
root
 |-- _1: string (nullable = true)
 |-- _2: string (nullable = true)
```

Took 0 sec. Last updated by anonymous at October 06 2022, 10:04:41 AM. (outdated)

```
dfFromRDD1.show()
```

FINISHED

```
+------+------+
|    _1|    _2|
+------+------+
|  Java| 20000|
|Python|100000|
| Scala|  3000|
+------+------+
```

Took 0 sec. Last updated by anonymous at October 06 2022, 10:04:44 AM.

```
val dfFromRDD2 = spark.createDataFrame(rdd).toDF(columns:_*)
```

FINISHED

```
dfFromRDD2: org.apache.spark.sql.DataFrame = [language: string, users_count: string]
```

Took 1 sec. Last updated by anonymous at October 06 2022, 9:51:35 AM.

```
dfFromRDD2.show()
```

FINISHED

```
+--------+-----------+
|language|users_count|
+--------+-----------+
|    Java|      20000|
|  Python|     100000|
|   Scala|       3000|
+--------+-----------+
```

Took 1 sec. Last updated by anonymous at October 06 2022, 9:57:06 AM.

```
import spark.implicits._
val simpleData = Seq(("James","Sales","NY",90000,34,10000),
    ("Michael","Sales","NY",86000,56,20000),
    ("Robert","Sales","CA",81000,30,23000),
    ("Maria","Finance","CA",90000,24,23000),
    ("Raman","Finance","CA",99000,40,24000),
    ("Scott","Finance","NY",83000,36,19000),
    ("Jen","Finance","NY",79000,53,15000),
    ("Jeff","Marketing","CA",80000,25,18000),
    ("Kumar","Marketing","NY",91000,50,21000)
  )
val df = simpleData.toDF("employee_name","department","state","salary","age","bonus")
df.show()
```

FINISHED

```
import spark.implicits._
simpleData: Seq[(String, String, String, Int, Int, Int)] = List((James,Sales,NY,90000,34,1000
0), (Michael,Sales,NY,86000,56,20000), (Robert,Sales,CA,81000,30,23000), (Maria,Finance,CA,9000
0,24,23000), (Raman,Finance,CA,99000,40,24000), (Scott,Finance,NY,83000,36,19000), (Jen,Financ
e,NY,79000,53,15000), (Jeff,Marketing,CA,80000,25,18000), (Kumar,Marketing,NY,91000,50,21000))
df: org.apache.spark.sql.DataFrame = [employee_name: string, department: string ... 4 more fiel
ds]
+-------------+----------+-----+------+---+-----+
|employee_name|department|state|salary|age|bonus|
+-------------+----------+-----+------+---+-----+
|        James|     Sales|   NY| 90000| 34|10000|
|      Michael|     Sales|   NY| 86000| 56|20000|
|       Robert|     Sales|   CA| 81000| 30|23000|
|        Maria|   Finance|   CA| 90000| 24|23000|
|        Raman|   Finance|   CA| 99000| 40|24000|
|        Scott|   Finance|   NY| 83000| 36|19000|
|          Jen|   Finance|   NY| 79000| 53|15000|
```

Took 1 sec. Last updated by anonymous at October 06 2022, 10:34:10 AM.

```
df.groupBy("department").sum("salary").show()
```

FINISHED

```
+----------+-----------+
|department|sum(salary)|
+----------+-----------+
|     Sales|     257000|
|   Finance|     351000|
```

```
| Marketing|     171000|
+----------+-----------+
```

Took 6 sec. Last updated by anonymous at October 06 2022, 10:34:33 AM.

```
df.groupBy("department").min("salary").show()
```

FINISHED

```
+----------+-----------+
|department|min(salary)|
+----------+-----------+
|     Sales|      81000|
|   Finance|      79000|
| Marketing|      80000|
+----------+-----------+
```

Took 3 sec. Last updated by anonymous at October 06 2022, 10:41:27 AM.

```
val g=df.groupBy("department")
```

FINISHED

```
g: org.apache.spark.sql.RelationalGroupedDataset = RelationalGroupedDataset: [grouping expressions: [department: string], value: [employee_name: string, department: string ... 4 more fields], type: GroupBy]
```

Took 0 sec. Last updated by anonymous at October 06 2022, 10:41:54 AM.

```
g.min("salary").show()
```

FINISHED

```
+----------+-----------+
|department|min(salary)|
+----------+-----------+
|     Sales|      81000|
|   Finance|      79000|
| Marketing|      80000|
+----------+-----------+
```

Took 2 sec. Last updated by anonymous at October 06 2022, 10:42:23 AM.

```
g
```

FINISHED

```
res17: org.apache.spark.sql.RelationalGroupedDataset = RelationalGroupedDataset: [grouping expressions: [department: string], value: [employee_name: string, department: string ... 4 more fields], type: GroupBy]
```

Took 0 sec. Last updated by anonymous at October 06 2022, 10:42:32 AM.

```
val d=df.select("department").distinct()
```

FINISHED

```
d: org.apache.spark.sql.Dataset[org.apache.spark.sql.Row] = [department: string]
```

Took 0 sec. Last updated by anonymous at October 06 2022, 10:53:18 AM.

```
d.show()
```

FINISHED

```
+----------+
|department|
+----------+
|     Sales|
|   Finance|
| Marketing|
```

```
+----------+
```

Took 3 sec. Last updated by anonymous at October 06 2022, 10:53:26 AM.

```
df.filter("department='Sales'").show()
```

FINISHED

```
+-------------+----------+-----+------+---+-----+
|employee_name|department|state|salary|age|bonus|
+-------------+----------+-----+------+---+-----+
|        James|     Sales|   NY| 90000| 34|10000|
|      Michael|     Sales|   NY| 86000| 56|20000|
|       Robert|     Sales|   CA| 81000| 30|23000|
+-------------+----------+-----+------+---+-----+
```

Took 1 sec. Last updated by anonymous at October 06 2022, 10:54:41 AM.

```
df.filter("department='Marketing'").show()
```

FINISHED

```
+-------------+----------+-----+------+---+-----+
|employee_name|department|state|salary|age|bonus|
+-------------+----------+-----+------+---+-----+
|         Jeff| Marketing|   CA| 80000| 25|18000|
|        Kumar| Marketing|   NY| 91000| 50|21000|
+-------------+----------+-----+------+---+-----+
```

Took 1 sec. Last updated by anonymous at October 06 2022, 11:16:58 AM.

```
val x1=sc.parallelize (Array("Jhon","Fred"))
```

FINISHED

```
x1: org.apache.spark.rdd.RDD[String] = ParallelCollectionRDD[68] at parallelize at <console>:31
```

Took 0 sec. Last updated by anonymous at October 06 2022, 11:24:47 AM.

```
val x2=sc.parallelize(Array("jebil","Fery"))
```

FINISHED

```
x2: org.apache.spark.rdd.RDD[String] = ParallelCollectionRDD[69] at parallelize at <console>:31
```

Took 0 sec. Last updated by anonymous at October 06 2022, 11:44:19 AM.

```
x1.union(x2).collect()
```

FINISHED

```
res23: Array[String] = Array(Jhon, Fred, jebil, Fery)
```

Took 1 sec. Last updated by anonymous at October 06 2022, 11:44:49 AM.

```
val y1=x1.flatMap(z=>z.split(","))
val y2=x2.flatMap(z=>z.split(","))
```

FINISHED

```
y1: org.apache.spark.rdd.RDD[String] = MapPartitionsRDD[71] at flatMap at <console>:33
y2: org.apache.spark.rdd.RDD[String] = MapPartitionsRDD[72] at flatMap at <console>:33
```

Took 1 sec. Last updated by anonymous at October 06 2022, 11:45:56 AM.

```
y1.collect()
```

FINISHED

```
res24: Array[String] = Array(Jhon, Fred)
```

Took 0 sec. Last updated by anonymous at October 06 2022, 11:46:02 AM.

READY