02-groupby_and_reducebykey

February 22, 2025

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

spark = SparkSession.builder \
        .appName("Difference Group By and Reduce By Key") \
        .master("yarn") \
        .getOrCreate()
```

```
25/02/11 16:02:28 WARN SparkSession: Using an existing Spark session; only
runtime SQL configurations will take effect.
```

[2]:
```
spark
```

[2]:
```
<pyspark.sql.session.SparkSession at 0x7f2e473c6950>
```

[3]:
```
!hadoop fs -ls -h /ecommerce_data/ecommerce_data/300MB
```

```
Found 5 items
-rw-r--r--   2 lokesh hadoop    327.4 M 2025-02-10 16:01
/ecommerce_data/ecommerce_data/300MB/customers.csv
-rw-r--r--   2 lokesh hadoop    275.3 M 2025-02-10 16:01
/ecommerce_data/ecommerce_data/300MB/items.csv
-rw-r--r--   2 lokesh hadoop    271.0 M 2025-02-10 16:01
/ecommerce_data/ecommerce_data/300MB/orders.csv
-rw-r--r--   2 lokesh hadoop    268.4 M 2025-02-10 16:01
/ecommerce_data/ecommerce_data/300MB/payments.csv
-rw-r--r--   2 lokesh hadoop    256.5 M 2025-02-10 16:01
/ecommerce_data/ecommerce_data/300MB/shippings.csv
```

[4]:
```
hdfs_path = "/ecommerce_data/ecommerce_data/300MB/customers.csv"
```

[5]:
```
customer_rdd = spark.sparkContext.textFile(hdfs_path)
```

[6]:
```
header = customer_rdd.first()
```

[7]:
```
customer_rdd = customer_rdd.filter(lambda row : row != header)
```

[8]:
```
customer_rdd = customer_rdd.map(lambda row : row.split(","))
```

[9]:
```
customer_rdd.first()
```

[9]:
```
['0', 'Customer_0', 'Pune', 'Maharashtra', 'India', '2023-01-19', 'True']
```

[10]:
```
city_rdd = customer_rdd.map(lambda row : (row[2], 1))
```

0.0.1 Reduce By

[11]:
```
reduced_rdd = city_rdd.reduceByKey(lambda x, y : x + y)
```

[12]:
```
reduced_rdd.collect()
```

[12]:
```
[('Delhi', 661025),
 ('Chennai', 660249),
 ('Kolkata', 660174),
 ('Bangalore', 661013),
 ('Pune', 660737),
 ('Ahmedabad', 660218),
 ('Mumbai', 661241),
 ('Hyderabad', 662281)]
```

0.0.2 Group By Key

[13]:
```
grouped_rdd = city_rdd.groupByKey()
```

[14]:
```
grouped_result = grouped_rdd.map(lambda x : (x[0],len(x[1])))
```

[15]:
```
grouped_result.collect()
```

[15]:
```
[('Delhi', 661025),
 ('Pune', 660737),
 ('Kolkata', 660174),
 ('Chennai', 660249),
 ('Bangalore', 661013),
 ('Mumbai', 661241),
 ('Ahmedabad', 660218),
 ('Hyderabad', 662281)]
```

[16]:
```
spark.stop()
```