**Week 3 day 3 (20-11-2024)**

**Topics covered:**

1. **Actions in pyspark RDD**
2. **Transformations in RDD**
3. **Pair RDD operations**
4. **Actions in pair RDDs**
5. **Transformations in pair RDDs.**

**Reference:**

Transforming data with PySpark RDDs & its hands on.docx -

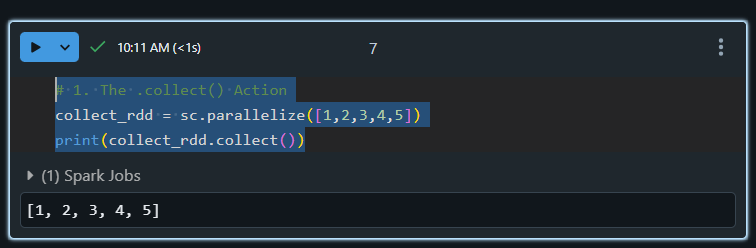
<https://drive.google.com/file/d/1ywSnK5e-iPDZ3mse2QdxB9MFNQ0UJAlP/view?usp=drive_link>

1. **Actions in pyspark RDD**

# 1. The .collect() Action

collect\_rdd = sc.parallelize([1,2,3,4,5])

print(collect\_rdd.collect())



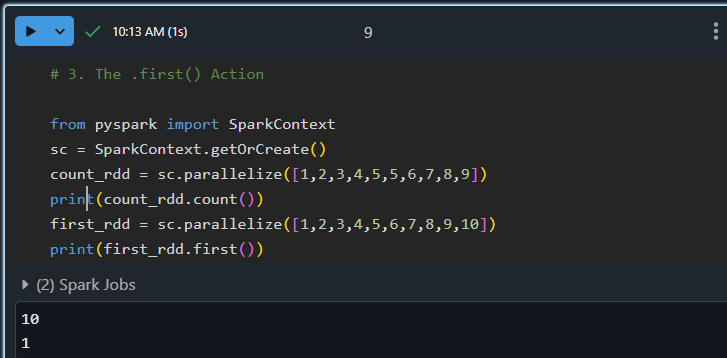
# 2. The .count() Action

from pyspark import SparkContext

sc = SparkContext.getOrCreate()

count\_rdd = sc.parallelize([1,2,3,4,5,5,6,7,8,9])

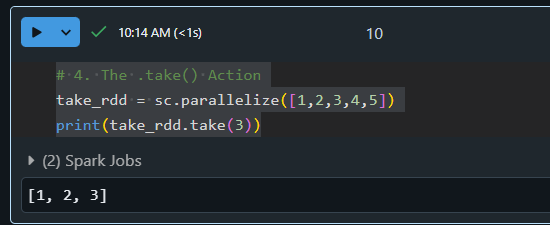
print(count\_rdd.count())



# 4. The .take() Action

take\_rdd = sc.parallelize([1,2,3,4,5])

print(take\_rdd.take(3))



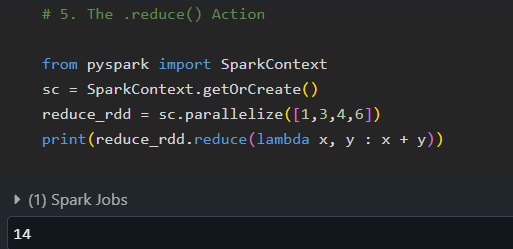
# 5. The .reduce() Action

from pyspark import SparkContext

sc = SparkContext.getOrCreate()

reduce\_rdd = sc.parallelize([1,3,4,6])

print(reduce\_rdd.reduce(lambda x, y : x + y))



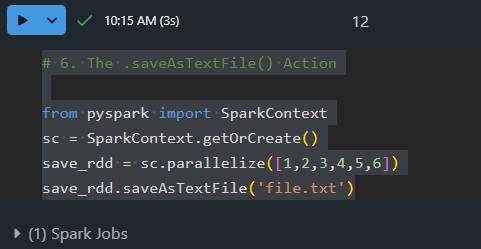
# 6. The .saveAsTextFile() Action

from pyspark import SparkContext

sc = SparkContext.getOrCreate()

save\_rdd = sc.parallelize([1,2,3,4,5,6])

save\_rdd.saveAsTextFile('file.txt')

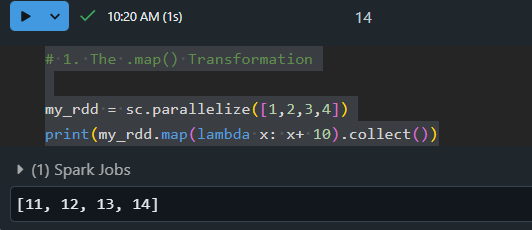


1. **Transformations in PySpark RDDs**

# 1. The .map() Transformation

my\_rdd = sc.parallelize([1,2,3,4])

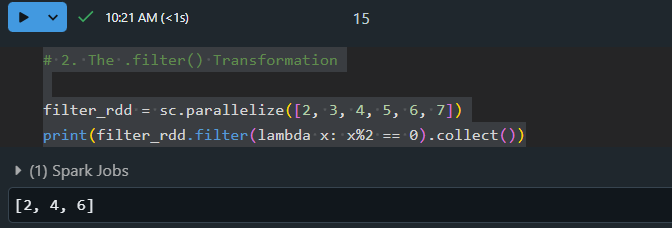
print(my\_rdd.map(lambda x: x+ 10).collect())



# 2. The .filter() Transformation

filter\_rdd = sc.parallelize([2, 3, 4, 5, 6, 7])

print(filter\_rdd.filter(lambda x: x%2 == 0).collect())



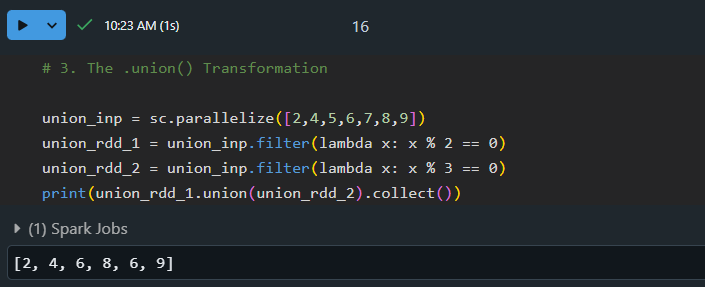
# 3. The .union() Transformation

union\_inp = sc.parallelize([2,4,5,6,7,8,9])

union\_rdd\_1 = union\_inp.filter(lambda x: x % 2 == 0)

union\_rdd\_2 = union\_inp.filter(lambda x: x % 3 == 0)

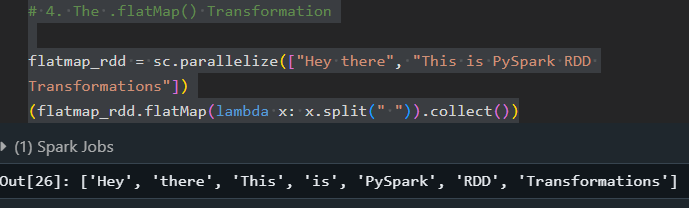
print(union\_rdd\_1.union(union\_rdd\_2).collect())



# 4. The .flatMap() Transformation

flatmap\_rdd = sc.parallelize(["Hey there", "This is PySpark RDD Transformations"])

(flatmap\_rdd.flatMap(lambda x: x.split(" ")).collect())



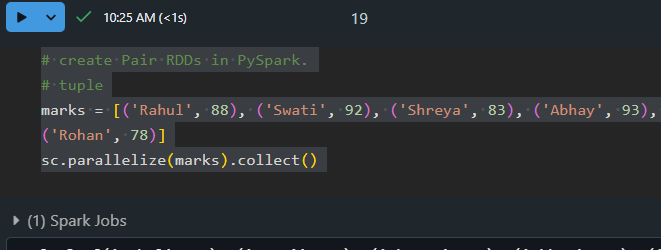
1. **Pair RDD operations**

# create Pair RDDs in PySpark.

# tuple

marks = [('Rahul', 88), ('Swati', 92), ('Shreya', 83), ('Abhay', 93), ('Rohan', 78)]

sc.parallelize(marks).collect()



1. **Actions in pair RDDs**

# 1. The countByKey() Action

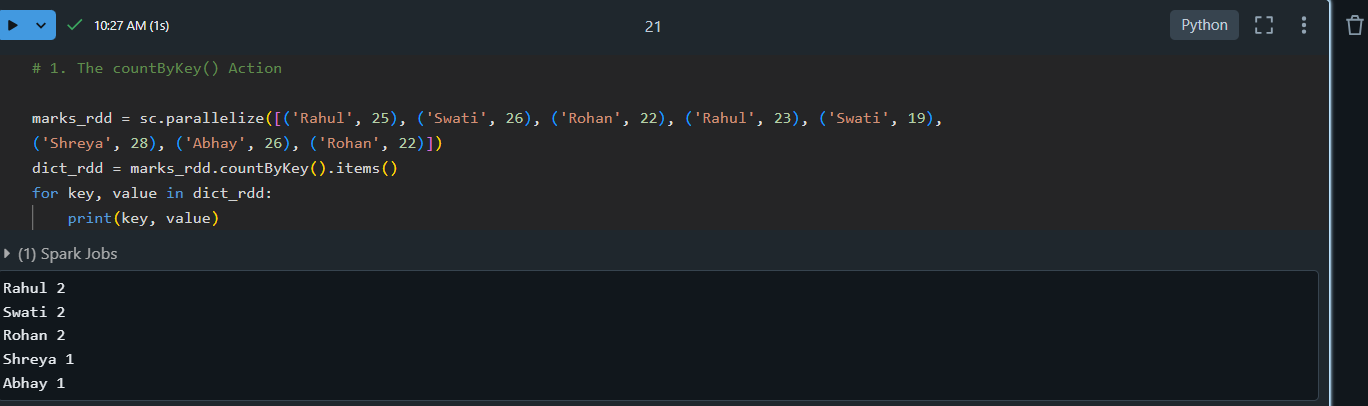
marks\_rdd = sc.parallelize([('Rahul', 25), ('Swati', 26), ('Rohan', 22), ('Rahul', 23), ('Swati', 19),

('Shreya', 28), ('Abhay', 26), ('Rohan', 22)])

dict\_rdd = marks\_rdd.countByKey().items()

for key, value in dict\_rdd:

    print(key, value)



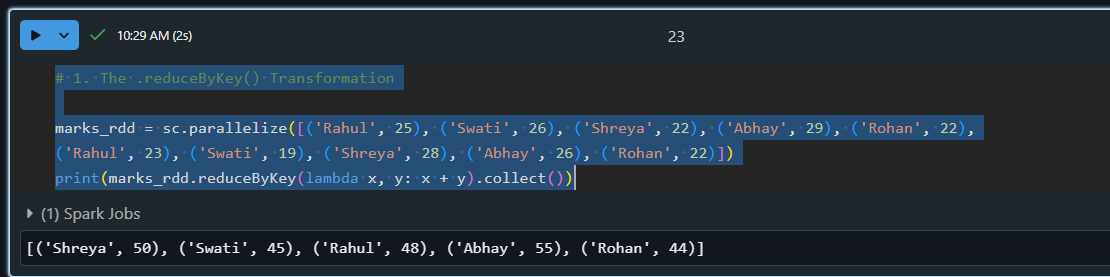
1. **Transformations in pair RDDs.**

# 1. The .reduceByKey() Transformation

marks\_rdd = sc.parallelize([('Rahul', 25), ('Swati', 26), ('Shreya', 22), ('Abhay', 29), ('Rohan', 22),

('Rahul', 23), ('Swati', 19), ('Shreya', 28), ('Abhay', 26), ('Rohan', 22)])

print(marks\_rdd.reduceByKey(lambda x, y: x + y).collect())

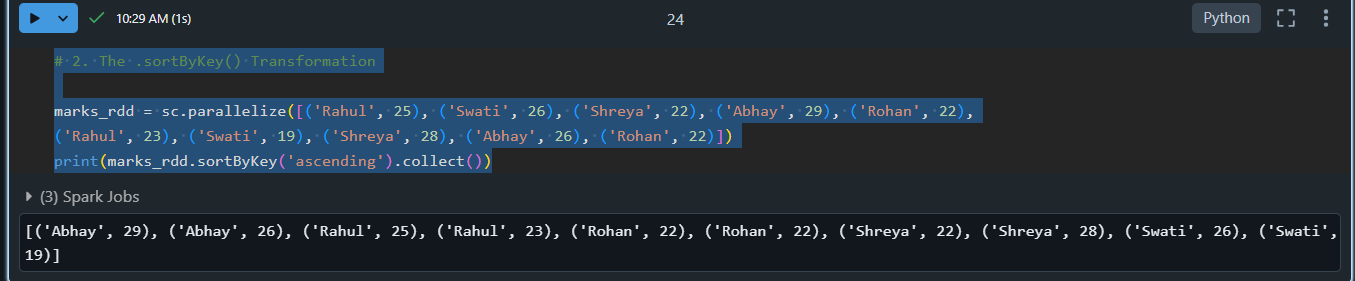


# 2. The .sortByKey() Transformation

marks\_rdd = sc.parallelize([('Rahul', 25), ('Swati', 26), ('Shreya', 22), ('Abhay', 29), ('Rohan', 22),

('Rahul', 23), ('Swati', 19), ('Shreya', 28), ('Abhay', 26), ('Rohan', 22)])

print(marks\_rdd.sortByKey('ascending').collect())



# 3. The .groupByKey() Transformation

marks\_rdd = sc.parallelize([('Rahul', 25), ('Swati', 26), ('Shreya', 22), ('Abhay', 29), ('Rohan', 22),

('Rahul', 23), ('Swati', 19), ('Shreya', 28), ('Abhay', 26), ('Rohan', 22)])

dict\_rdd = marks\_rdd.groupByKey().collect()

for key, value in dict\_rdd:

    print(key, list(value))

