Experiment-8

Aim: Word count problem and caching.

Q. write a spark program to do caching.

1. Read file spark.txt

→ text_file = sc.textFile ("FileStore / tables / spark.txt")

2. Split the file

→ data_split = text_file. flatMap (lambda x: x.split(" "))

3. Make a pair RDD by adding 1 to each word.

→ data_pair = data_split. map(lambda word: (word, 1))

4 word count

→ counts = data_pair. reduceByKey (lambda a,b: a+b)

5. save the result into file.

→ counts. saveAsTextFile ("dbfs: /FileStore/ tables/ result.txt")

6. Verify the result.

→ sc. textFile ("dbfs: /FileStore/ tables/ result.txt"). take(5)

Q. **Caching**

1. create a RDD using parallelize()

→ content = sc. parallelize ([1, 2, 3, 4, 5, 2, 4, 1])

2. cache the RDD.

→ content.cache()

3. check if RDD is cached or Not.

→ content.is_cached

4. Remove persistance using unpersist()

→ content.unpersist()

5. check if RDD is cached OR NOt

→ content.is_cached.

6. cache the RDD using persist()

→ content.persist()

7. check if RDD is cached or not.

→ content.is_cached.

8. Remove persistance using unpersist()

→ content.unpersist.

9. check if RDD is cached or Not.

→ content.is_cached.

10. cache the RDD when we have multiple transformations.

```
→ a = sc.textFile ("dbfs:/FileStore/tables/movies.txt")
  a.take (5)

→ b = a.map(lambda x: x.upper())
  b.take (5)

→ c = b.filter (lambda x: x.startswith ("1")
  c.take(5)
  c.count.

→ c.persist

→ c.count.
```

8/3/2021

Output

5 (1) SparkJobs

6 (1) SparkJobs

Out[11]: ["('spark', 1)", "('run', 1)", "('programs', 2)"
, "("in", 1)", "('lazy', 1)"]

- Caching

2 Out[15]: ParallelcollectionRDD[21] at readRDDFrom InputStream at PythonRDD

3 True

4 Out[15]: ParallelcollectionRDD[21] at readRDDFrom InputStream at PythonRDD.

5 False

6 Out[17]: ParallelcollectionRDD[21] at readRDDFrom InputStream at PythonRDD.

7 True

8 <bound method RDD.unpersist of ParallelcollectionRDD [21] at readRDD from Input stream at pythonrdd.

9 True.

```
10 Out[21]:
   ["1:: Toystory (1995):: Animated|childrean's|comedy",
    "2:: Jumanji (1995):: Adventure|childreans|Fantasy"]

11 Out[22]:
  ["1:: TOYSTORY(1995):: ANIMATED|CHILDREN'S|COMEDY"
   "2:: JUMANJI (1995):: ADVENTURE|CHIDDREN'S|FANTASY"]

12 Out[23]:
  ["1:: TOYSTORY(1995):: ANINATED|CHILDREN'S|COMEDY"
   "10:: GOLDEN EYE (1995):: ACTION|ADVENTURE|THRILLER"]

13 Out[24]: 1055

14 Out[25]: Python RDD[28] at RDD at PythonRDD

15 1055
```