Experiment-7

Q 7(a) Broadcasting-Variables

1. create a movies RDD using movies.txt. Split the RDD based on "::" as delimeter.

→
```
movies = sc.textFile ("FileStore/tables/movies.txt")
                .map(lambda x:x.split("::"))
movies.take(2)
```

2. create a pair RDD with movie ID being the key and movie name being its value.

→
```
movies_pair = movies.map (lambda x : (x[0], x[1]))
```

3. Display 2 elements of RDD by using take().

→
```
movies_pair.take(2)
```

4. Broadcast data as dictionary

→
```
broadcast_var = sc.broadcast(movies_pair.collectAsMap())
```

5. Validate type of variable value.

→
```
type(broadcast_var)
```

6. create a ratings RDD using ratings.txt. split the RDD based on "::" as delimiter.

→
```
ratings = sc.textFile("FileStore/tables/ratings.txt")
                .map(lambda x:x.split("::"))
```

7. Display 2 elements of RDD using take()

→ ratings. take(2)

8. Map movieids in ratings RDD to movienames from broadcasted variable.

→ movie_ratings = ratings.map(lambda x:(
        broadcast_var.get(x[0]), x[2]))

9. Display 2 elements of the RDD using take()

→ movie_ratings.take(2)

10. Print toDebugString

→ print(movie_ratings.toDebugString())

Q. 7(b) Accumulators

1. Create an accumulator num.

→ num = sc.accumulator(10)

2. Define a function func(x)

```
→ def func(x):
      global num
      num += x
```

3. Initiallize a RDD using parallelize function.

```
→ mylist = [10, 20, 30, 40, 50]
  myrdd = sc.parallelize (mylist)
```

4. Pass function func to foreach to RDD. Initialize a RDD using parallelize function.

```
→ myrdd. foreach(func)
```

5. Print the Accumulator value.

```
→ final = num.value
  print(" The value of Accumulator is : %i " %(final))
```

Dated- 1/3/2021

Outputs

```
Out[1]:[['1', 'Toy story (1995)', ''Animation|children|comedy']
['2', 'Jumanji (1995)', ''Adventure|children's|Fantasy]]
```

```
Out[3]:
[('1', 'Toy story (1995)'),
 ('2', 'Jumanji (1995)')]
```

```
Out[5]: pyspark.broadcast.Broadcast
```

```
Out[7]: [['1', '1193', '5', '978300760'],
         ['1', '661', '3', '978302109']]
```

```
Out[9]: [('Toy story (1995)', '5'),
         ('Toy story (1995)', '3')]
```

```
Out[10]:
b'(2) PythonRDD[9] at RDD at PythonRDD.scala:58 [] \n |
Filestore/tables/ratings.txt MapPartitionsRDD[6] at
textFile at NativeMethodAccessorImpl.java:0 [] \n
| Filestore/tables/ratings.txt HadoopRDD[5] at
textFile at NativeMethodAccessorImpl.java:0 []
```

Outputs:

```
Out[1]:
 The value of Accumulator is : 160

 mylist = [10,20,30,40,50] ::
```