CSP554—Big Data Technologies

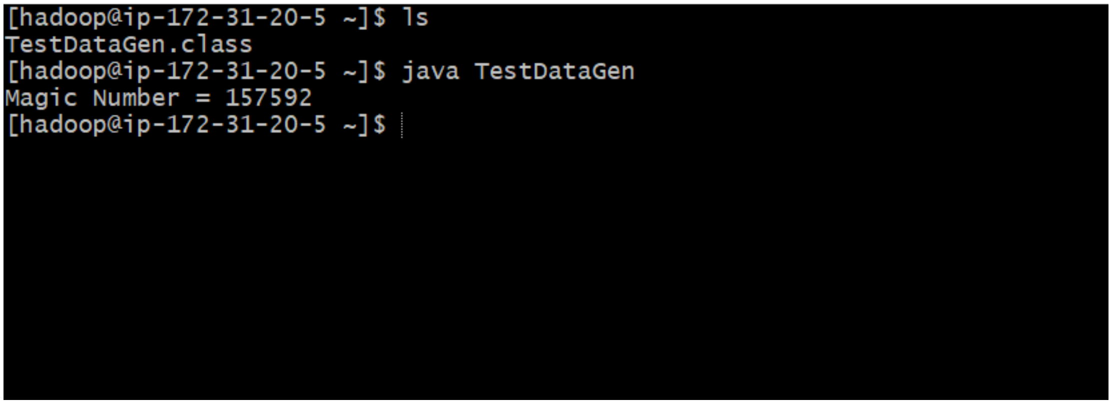
Assignment #06 (Modules 06)

Exercises

Exercise 1)

Use the TestDataGen program from previous assignments to generate a new foodratings<magic_number>.txt data file.

```
Magic Number = 157592
```



```
[hadoop@ip-172-31-20-5 ~]$ ls
TestDataGen.class
[hadoop@ip-172-31-20-5 ~]$ java TestDataGen
Magic Number = 157592
[hadoop@ip-172-31-20-5 ~]$
```

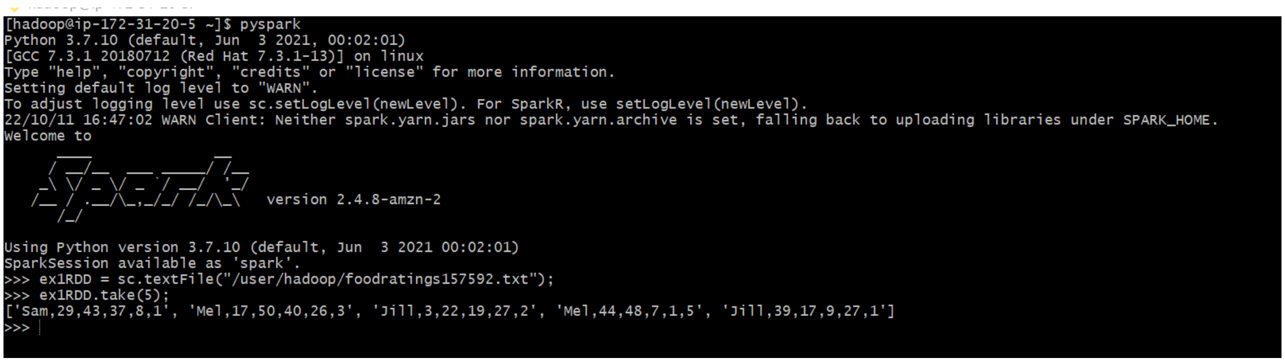
Copy the file to HDFS, say into the /user/hadoop directory.

```
hadoop fs -copyFromLocal /home/hadoop/foodplaces157592.txt
/user/hadoop/;

hadoop fs -copyFromLocal /home/hadoop/foodplaces157592.txt
/user/hadoop/;

ex1RDD = sc.textFile("/user/hadoop/foodratings157592.txt");

ex1RDD.take(5)
```



```
[hadoop@ip-172-31-20-5 ~]$ pyspark
Python 3.7.10 (default, Jun  3 2021, 00:02:01)
[GCC 7.3.1 20180712 (Red Hat 7.3.1-13)] on linux
Type "help", "copyright", "credits" or "license" for more information.
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
22/10/11 16:47:02 WARN Client: Neither spark.yarn.jars nor spark.yarn.archive is set, falling back to uploading libraries under SPARK_HOME.
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /__ / .__/\_,_/_/ /_/\_\   version 2.4.8-amzn-2
      /_/

Using Python version 3.7.10 (default, Jun  3 2021 00:02:01)
SparkSession available as 'spark'.
>>> ex1RDD = sc.textFile("/user/hadoop/foodratings157592.txt");
>>> ex1RDD.take(5);
['Sam,29,43,37,8,1', 'Mel,17,50,40,26,3', 'Jill,3,22,19,27,2', 'Mel,44,48,7,1,5', 'Jill,39,17,9,27,1']
>>>
```

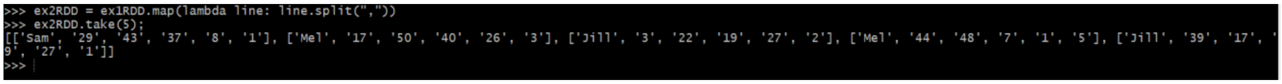
Exercise 2)

Create another RDD called ex2RDD where each record of this new RDD has 6 fields, each a string, by splitting apart each record on “,” boundaries from the ex1RDD.

List the first five records of this RDD using the “take(5)” action and copy them to your assignment submission for this exercise.

```
ex2RDD = ex1RDD.map(lambda line: line.split(“,”))

ex2RDD.take(5);
```



```
>>> ex2RDD = ex1RDD.map(lambda line: line.split(","))
>>> ex2RDD.take(5);
[['Sam', '29', '43', '37', '8', '1'], ['Mel', '17', '50', '40', '26', '3'], ['Jill', '3', '22', '19', '27', '2'], ['Mel', '44', '48', '7', '1', '5'], ['Jill', '39', '17', '
9', '27', '1']]
>>>
```

Exercise 3)

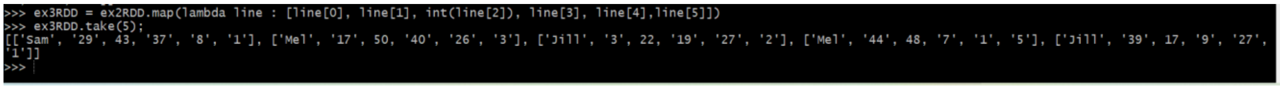
Create another RDD called ex3RDD from ex2RDD where each record of this new RDD has its thirdcolumn converted from a string to an integer.
Hint: Use a lambda function something like the following:

lambda line : [line[0], line[1], int(line[2]), line[3], line[4], line[5]]

List the first five records of this RDD using the “take(5)” action and copy them to your assignment submission for this exercise.

```
ex3RDD = ex2RDD.map(lambda line : [line[0], line[1], int(line[2]),
line[3], line[4],line[5]])

ex3RDD.take(5);
```



```
>>> ex3RDD = ex2RDD.map(lambda line : [line[0], line[1], int(line[2]), line[3], line[4],line[5]])
>>> ex3RDD.take(5);
[['Sam', '29', 43, '37', '8', '1'], ['Mel', '17', 50, '40', '26', '3'], ['Jill', '3', 22, '19', '27', '2'], ['Mel', '44', 48, '7', '1', '5'], ['Jill', '39', 17, '9', '27',
'1']]
>>>
```

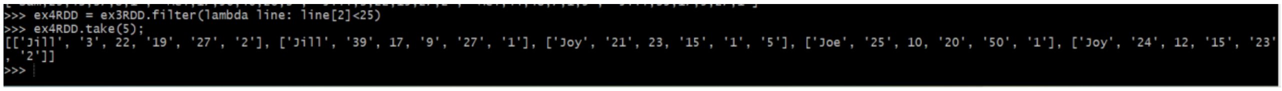
Exercise 4)

Create another RDD called ex4RDD from ex3RDD where each record of this new RDD is allowed to havea value for its third field that is less than 25 (<25).

List the first five records of this RDD using the “take(5)” action and copy them to your assignment submission for this exercise.

```
ex4RDD = ex3RDD.filter(lambda line: line[2]<25)

ex4RDD.take(5);
```



```
>>> ex4RDD = ex3RDD.filter(lambda line: line[2]<25)
>>> ex4RDD.take(5);
[['Jill', '3', 22, '19', '27', '2'], ['Jill', '39', 17, '9', '27', '1'], ['Joy', '21', 23, '15', '1', '5'], ['Joe', '25', 10, '20', '50', '1'], ['Joy', '24', 12, '15', '23'
, '2']]
>>>
```

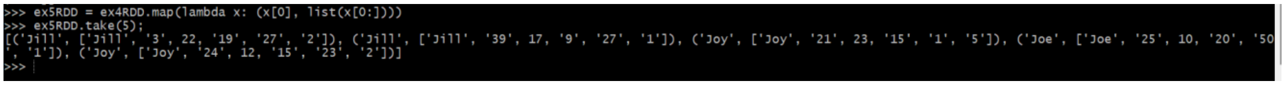
Exercise 5)

Create another RDD called ex5RDD from ex4RDD where each record is a key value pair where the key is the first field of the record and the value is the entire record

List the first five records of this RDD using the "take(5)" action and copy them to your assignment submission for this exercise.

```
ex5RDD = ex4RDD.map(lambda x: (x[0], list(x[0:])))

ex5RDD.take(5);
```



```
>>> ex5RDD = ex4RDD.map(lambda x: (x[0], list(x[0:])))
>>> ex5RDD.take(5);
[('Jill', ['Jill', '3', 22, '19', '27', '2']), ('Jill', ['Jill', '39', 17, '9', '27', '1']), ('Joy', ['Joy', '21', 23, '15', '1', '5']), ('Joe', ['Joe', '25', 10, '20', '50
', '1']), ('Joy', ['Joy', '24', 12, '15', '23', '2'])]
>>>
```

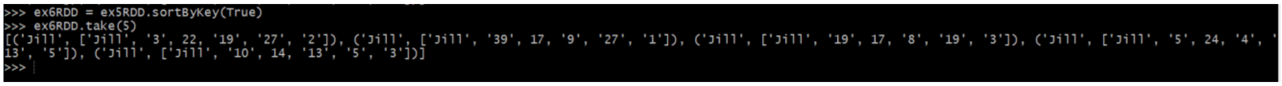
Exercise 6)

Create another RDD called ex6RDD from ex5RDD where the records are organized in ascending order by key

List the first five records of this RDD using the "take(5)" action and copy them to your assignment submission for this exercise.

```
ex6RDD = ex5RDD.sortByKey(True)

ex6RDD.take(5);
```



```
>>> ex6RDD = ex5RDD.sortByKey(True)
>>> ex6RDD.take(5)
[('Jill', ['Jill', '3', 22, '19', '27', '2']), ('Jill', ['Jill', '39', 17, '9', '27', '1']), ('Jill', ['Jill', '19', 17, '8', '19', '3']), ('Jill', ['Jill', '5', 24, '4', '
13', '5']), ('Jill', ['Jill', '10', 14, '13', '5', '3'])]
>>>
```