Akash Tanwani A20448831

CSP554—Big Data Technologies

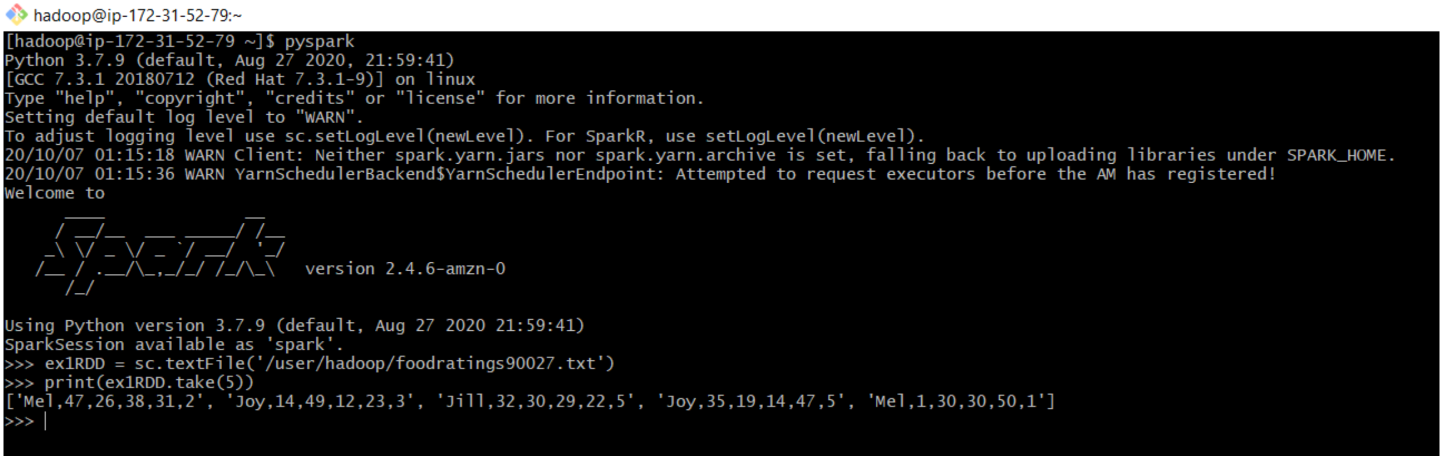
Assignment #06 (Modules 06)

Exercises

Exercise 1)

```
Magic Number = 90027
```

```
Command:
>>> ex1RDD = sc.textFile('/user/hadoop/foodratings90027.txt')
>>> print(ex1RDD.take(5))
```



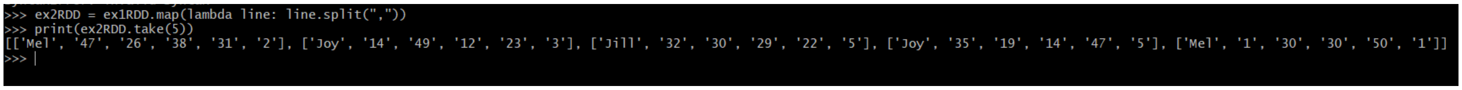
hadoop@ip-172-31-52-79:~

```
[hadoop@ip-172-31-52-79 ~]$ pyspark
Python 3.7.9 (default, Aug 27 2020, 21:59:41)
[GCC 7.3.1 20180712 (Red Hat 7.3.1-9)] 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).
20/10/07 01:15:18 WARN Client: Neither spark.yarn.jars nor spark.yarn.archive is set, falling back to uploading libraries under SPARK_HOME.
20/10/07 01:15:36 WARN YarnSchedulerBackend$YarnSchedulerEndpoint: Attempted to request executors before the AM has registered!
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /__ / .__/\_,_/_/ /_/\_\   version 2.4.6-amzn-0
      /_/

Using Python version 3.7.9 (default, Aug 27 2020 21:59:41)
SparkSession available as 'spark'.
>>> ex1RDD = sc.textFile('/user/hadoop/foodratings90027.txt')
>>> print(ex1RDD.take(5))
['Mel,47,26,38,31,2', 'Joy,14,49,12,23,3', 'Jill,32,30,29,22,5', 'Joy,35,19,14,47,5', 'Mel,1,30,30,50,1']
>>> |
```

Exercise 2)

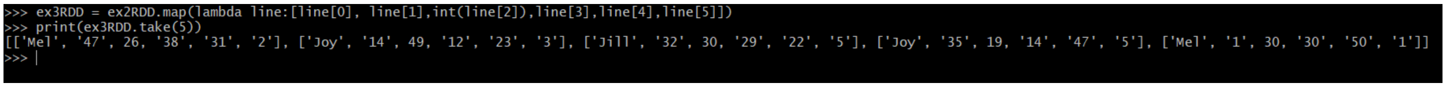
```
Command:
>>> ex2RDD = ex1RDD.map(lambda line: line.split(","))
>>> print(ex2RDD.take(5))
```



```
>>> ex2RDD = ex1RDD.map(lambda line: line.split(","))
>>> print(ex2RDD.take(5))
[['Mel', '47', '26', '38', '31', '2'], ['Joy', '14', '49', '12', '23', '3'], ['Jill', '32', '30', '29', '22', '5'], ['Joy', '35', '19', '14', '47', '5'], ['Mel', '1', '30', '30', '50', '1']]
>>> |
```

Exercise 3)

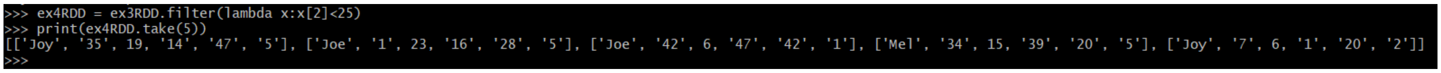
```
Command:
>>> ex3RDD = ex2RDD.map(lambda line:[line[0], line[1],int(line[2]),line[3],line[4],line[5]])
>>> print(ex3RDD.take(5))
```



```
>>> ex3RDD = ex2RDD.map(lambda line:[line[0], line[1],int(line[2]),line[3],line[4],line[5]])
>>> print(ex3RDD.take(5))
[['Mel', '47', 26, '38', '31', '2'], ['Joy', '14', 49, '12', '23', '3'], ['Jill', '32', 30, '29', '22', '5'], ['Joy', '35', 19, '14', '47', '5'], ['Mel', '1', 30, '30', '50', '1']]
>>> |
```

Exercise 4)

```
Command:
>>> ex4RDD = ex3RDD.filter(lambda x:x[2]<25)
>>> print(ex4RDD.take(5))
```

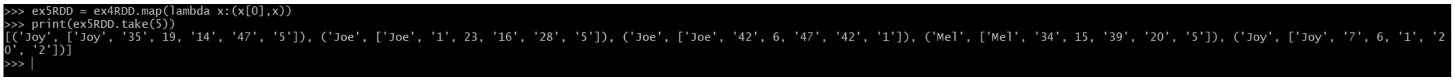


```
>>> ex4RDD = ex3RDD.filter(lambda x:x[2]<25)
>>> print(ex4RDD.take(5))
[['Joy', '35', 19, '14', '47', '5'], ['Joe', '1', 23, '16', '28', '5'], ['Joe', '42', 6, '47', '42', '1'], ['Mel', '34', 15, '39', '20', '5'], ['Joy', '7', 6, '1', '20', '2']]
>>>
```

Exercise 5)

Command:

```
>>> ex5RDD = ex4RDD.map(lambda x:(x[0],x))
>>> print(ex5RDD.take(5))
```

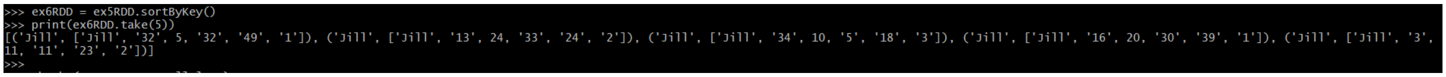


```
>>> ex5RDD = ex4RDD.map(lambda x:(x[0],x))
>>> print(ex5RDD.take(5))
[('Joy', ['Joy', '35', 19, '14', '47', '5']), ('Joe', ['Joe', '1', 23, '16', '28', '5']), ('Joe', ['Joe', '42', 6, '47', '42', '1']), ('Mel', ['Mel', '34', 15, '39', '20', '5']), ('Joy', ['Joy', '7', 6, '1', '2
0', '2'])]
>>> |
```

Exercise 6)

Command:

```
>>> ex6RDD = ex5RDD.sortByKey()
>>> print(ex6RDD.take(5))
```



```
>>> ex6RDD = ex5RDD.sortByKey()
>>> print(ex6RDD.take(5))
[('Jill', ['Jill', '32', 5, '32', '49', '1']), ('Jill', ['Jill', '13', 24, '33', '24', '2']), ('Jill', ['Jill', '34', 10, '5', '18', '3']), ('Jill', ['Jill', '16', 20, '30', '39', '1']), ('Jill', ['Jill', '3',
11, '11', '23', '2'])]
>>>
```