ASSIGNMENT : 06TH

Magic num : 24046

Exercise : 1 :

Print the first 5 records of RDD by “take(5)”

The following is the Command:

```
hdfs dfs -put foodratings24046.txt /user/hadoop

ex1RDD = sc.textFile("/user/hadoop/foodratings24046.txt")
record = ex1RDD.take(5) for rc in record: print(repr(rc))
```

result :

```
u'Mel,5,17,14,46,4'
u'Mel,15,2,34,46,4'
u'Sam,17,1,16,45,2'
u'Sam,4,40,6,31,4'
u'Joy,6,28,4,38,3'
>>>
```

Exercise : 2

each record of this new RDD has 6 fields, each a string, by splitting apart each record on “,” boundaries from the ex1RDD

The following is the command:

```
ex2RDD = ex1RDD.map(lambda record: record.split(','))
record = ex2RDD.take(5)
for rc in record:
 print(repr(rc))
```

result:

```
...
[u'Mel', u'5', u'17', u'14', u'46', u'4']
[u'Mel', u'15', u'2', u'34', u'46', u'4']
[u'Sam', u'17', u'1', u'16', u'45', u'2']
[u'Sam', u'4', u'40', u'6', u'31', u'4']
[u'Joy', u'6', u'28', u'4', u'38', u'3']
>>>
```

Exercise : 3:

another RDD called ex3RDD from ex2RDD where each record of this new RDD has its third column converted from a string to an integer

The following is the command:

```
ex3RDD = ex2RDD.map(lambda line: [line[0], line[1], int(line[2]), line[3],
line[4],line[5]])
ex3 = ex3RDD.take(5)
for rc in ex3:
 print(repr(rc))
```

Result:

```
[u'Mel', u'5', 17, u'14', u'46', u'4']
[u'Mel', u'15', 2, u'34', u'46', u'4']
[u'Sam', u'17', 1, u'16', u'45', u'2']
[u'Sam', u'4', 40, u'6', u'31', u'4']
[u'Joy', u'6', 28, u'4', u'38', u'3']
>>>
```

Exercise : 4:

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

The following is the command :

```
ex4RDD = ex3RDD.filter(lambda line: line[2] < 25)
ex4 = ex4RDD.take(5)
for ln in ex4:
 print(repr(ln))
```

Result:

```
...
[u'Mel', u'5', 17, u'14', u'46', u'4']
[u'Mel', u'15', 2, u'34', u'46', u'4']
[u'Sam', u'17', 1, u'16', u'45', u'2']
[u'Mel', u'21', 9, u'48', u'3', u'4']
[u'Joy', u'27', 5, u'3', u'17', u'1']
>>>
```

Exercise : 5

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

The following is the command :

```
ex5RDD = ex4RDD.map(lambda record: (record[0], tuple(record)))
ex5 = ex5RDD.take(5)
for rc in ex5:
 print(repr(rc))
```

Result :

```
...
(u'Mel', (u'Mel', u'5', 17, u'14', u'46', u'4'))
(u'Mel', (u'Mel', u'15', 2, u'34', u'46', u'4'))
(u'Sam', (u'Sam', u'17', 1, u'16', u'45', u'2'))
(u'Mel', (u'Mel', u'21', 9, u'48', u'3', u'4'))
(u'Joy', (u'Joy', u'27', 5, u'3', u'17', u'1'))
>>>
```

Exercise : 6:

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

Result:

```
...
(u'Jill', (u'Jill', u'49', 7, u'35', u'36', u'5'))
(u'Jill', (u'Jill', u'49', 10, u'15', u'3', u'2'))
(u'Jill', (u'Jill', u'34', 20, u'10', u'26', u'2'))
(u'Jill', (u'Jill', u'13', 4, u'30', u'28', u'4'))
(u'Jill', (u'Jill', u'10', 24, u'8', u'15', u'2'))
>>>
```