# CSP554—Big Data Technologies

## Assignment #7 (Modules 06)

## Exercise 1)

**Read in the text file into an RDD named ex1RDD.**

**This RDD should now have records each consisting of a single string having 6 comma-separated parts.**

## Magic Number – 79189

## 

## Commands:

>>> ex1RDD = sc.textFile("/user/hadoop/foodratings79189.txt")

## >>> ex1RDD.take(5)

## [

## u'Mel,43,2,15,22,4',

## u'Joy,44,46,39,18,1',

## u'Mel,26,2,34,34,3',

## u'Joe,47,50,48,18,3',

## u'Joy,45,32,29,33,2'

## ]

## 

## 

## 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.**

## Commands:

>>> ex2RDD = ex1RDD.map(lambda line: line.split(","))

>>> ex2RDD.take(5)

[

[u'Mel', u'43', u'2', u'15', u'22', u'4'],

[u'Joy', u'44', u'46', u'39', u'18', u'1'],

[u'Mel', u'26', u'2', u'34', u'34', u'3'],

[u'Joe', u'47', u'50', u'48', u'18', u'3'],

[u'Joy', u'45', u'32', u'29', u'33', u'2']

]

A screenshot of a cell phone

Description automatically generated

## Exercise 3)

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

## Commands:

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

>>> ex3RDD.take(5)

## [

## [u'Mel', u'43', 2, u'15', u'22', u'4'],

## [u'Joy', u'44', 46, u'39', u'18', u'1'],

## [u'Mel', u'26', 2, u'34', u'34', u'3'],

## [u'Joe', u'47', 50, u'48', u'18', u'3'],

## [u'Joy', u'45', 32, u'29', u'33', u'2']

## ]



## Exercise 4)

**Create 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).**

## Commands:

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

>>> ex4RDD.take(5)

## [

## [u'Mel', u'43', 2, u'15', u'22', u'4'],

## [u'Mel', u'26', 2, u'34', u'34', u'3'],

## [u'Mel', u'38', 19, u'12', u'16', u'5'],

## [u'Jill', u'15', 9, u'36', u'5', u'3'],

## [u'Jill', u'44', 17, u'21', u'48', u'5']

## ]

## 

## 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**

**Commands:**

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

>>> ex5RDD.take(5)

[

(u'Mel', [u'Mel', u'43', 2, u'15', u'22', u'4']),

(u'Mel', [u'Mel', u'26', 2, u'34', u'34', u'3']),

(u'Mel', [u'Mel', u'38', 19, u'12', u'16', u'5']),

(u'Jill', [u'Jill', u'15', 9, u'36', u'5', u'3']),

(u'Jill', [u'Jill', u'44', 17, u'21', u'48', u'5'])

]

A close up of a sign

Description automatically generated

## Exercise 6)

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

**Commands:**

>>> ex6RDD = ex5RDD.sortByKey(True)

>>> ex6RDD.take(5)

[

(u'Jill', [u'Jill', u'15', 9, u'36', u'5', u'3']),

(u'Jill', [u'Jill', u'44', 17, u'21', u'48', u'5']),

(u'Jill', [u'Jill', u'39', 8, u'15', u'49', u'2']),

(u'Jill', [u'Jill', u'3', 11, u'16', u'17', u'2']),

(u'Jill', [u'Jill', u'19', 22, u'5', u'18', u'1'])

]

A close up of a logo

Description automatically generated