1)Running TestDataGen.class

Magic Number:143689

```
[hadoop@ip-172-31-37-123 ~]$ java TestDataGen
Magic Number = 143689
[hadoop@ip-172-31-37-123 ~]$
```

Files Created after running

```
[hadoop@ip-172-31-37-123 ~]$ ls
foodplaces143689.txt  foodratings143689.txt  TestDataGen.class
[hadoop@ip-172-31-37-123 ~]$
```

Created new Directory csp554

```
[hadoop@ip-172-31-37-123 ~]$ hadoop fs -ls /user
Found 6 items
drwxr-xr-x   - hadoop   hadoop          0 2020-10-08 17:08 /user/csp554
drwxrwxrwx   - hadoop   hadoop          0 2020-10-08 17:01 /user/hadoop
drwxrwxrwx   - livy     livy            0 2020-10-08 17:01 /user/livy
drwxrwxrwx   - root     hadoop          0 2020-10-08 17:01 /user/root
drwxrwxrwx   - spark    spark           0 2020-10-08 17:01 /user/spark
drwxrwxrwx   - zeppelin hadoop          0 2020-10-08 17:01 /user/zeppelin
```

Copied generated files foodratings143894.txt and foodplaces143894.txt into /user/csp554 as foodratings and foodplaces respectively

```
[hadoop@ip-172-31-37-123 ~]$ hadoop fs -ls /user/csp554/
Found 2 items
-rw-r--r--   1 hadoop hadoop         59 2020-10-08 17:15 /user/csp554/foodplaces
.txt
-rw-r--r--   1 hadoop hadoop      17472 2020-10-08 17:15 /user/csp554/foodrating
s.txt
[hadoop@ip-172-31-37-123 ~]$
```

Creating an RDD named line

CommandUsed: line=sc.textFile('/user/csp554/foodratings.txt')

Line.take(5)

```
>>> line=sc.textFile('/user/csp554/foodratings.txt')
>>> line.take(5)
['Jill,22,45,50,11,4', 'Jill,42,21,45,45,2', 'Mel,2,44,35,6,1', 'Joe,44,15,17,28
,2', 'Joe,2,17,26,44,1']
>>>
```

2) Create a new RDD name ex2RDD and splitting it using “,”

Command Used: ex2RDD=line.map(lambda line :line.split(“,”))

ex2RDD.take(5)

```
>>> ex2RDD=sc.textFile('user/csp554/foodratings.txt')
>>> ex2RDD=line.map(lambda line:line.split(","))
>>> ex2RDD.take(5)
[['Jill', '22', '45', '50', '11', '4'], ['Jill', '42', '21', '45', '45', '2'], [
'Mel', '2', '44', '35', '6', '1'], ['Joe', '44', '15', '17', '28', '2'], ['Joe',
 '2', '17', '26', '44', '1']]
>>>
```

3) Create a new RDD name ex3RDDV2 and converting third column into int

Command Used: ex3RDDV2=ex2RDD.map(lambda line1:[line1[0], line1[1], int(line1[2]), line1[3], line1[4], line1[5]])

Ex3RDDV2.take(5)

```
>>> ex3RDDV2=ex2RDD.map(lambda line1 :[line1[0],line1[1],int(line1[2]),line1[3],
line1[4],line1[5]])
>>> ex3RDDV2.take(5)
[['Jill', '22', 45, '50', '11', '4'], ['Jill', '42', 21, '45', '45', '2'], ['Mel
', '2', 44, '35', '6', '1'], ['Joe', '44', 15, '17', '28', '2'], ['Joe', '2', 17
, '26', '44', '1']]
```

4) Create a new RDD name ex4RDDV2 and filtering third column such that it is less than 25

Command Used: ex4RDDV2=ex3RDDv2.filter(lambda line1 :line1[2]<25)

ex4RDDV2.take(5)

```
>>> ex4RDDV2=ex3RDDV2.filter(lambda line1 :line1[2]<25)
>>> ex4RDDV2.take(5)
[['Jill', '42', 21, '45', '45', '2'], ['Joe', '44', 15, '17', '28', '2'], ['Joe'
, '2', 17, '26', '44', '1'], ['Sam', '38', 9, '12', '8', '1'], ['Mel', '33', 22,
 '4', '40', '2']]
>>>
```

5) Create a new RDD name ex5RDD and making the first column the key

Command Used: ex5RDD=ex4RDDV2.map(lambda line: [line[0],line])
ex5RDD.take(5)

```
>>> ex5RDD=ex4RDDV2.map(lambda line: [line[0],line])
>>> ex5RDD.take(5)
[['Jill', ['Jill', '42', 21, '45', '45', '2']], ['Joe', ['Joe', '44', 15, '17',
'28', '2']], ['Joe', ['Joe', '2', 17, '26', '44', '1']], ['Sam', ['Sam', '38', 9
, '12', '8', '1']], ['Mel', ['Mel', '33', 22, '4', '40', '2']]]
>>>
```

6) ***Create a new RDD name ex6RDD where records are ordered by ascending value of the key***

```
>>> ex6RDD=ex5RDD.sortByKey()
>>> ex6RDD.take(6)
[('Jill', ['Jill', '42', 21, '45', '45', '2']), ('Jill', ['Jill', '8', 17, '50',
 '27', '1']), ('Jill', ['Jill', '27', 4, '23', '32', '1']), ('Jill', ['Jill', '4
7', 7, '2', '46', '5']), ('Jill', ['Jill', '26', 21, '13', '36', '5']), ('Jill',
 ['Jill', '1', 12, '25', '50', '2'])]
>>>
```