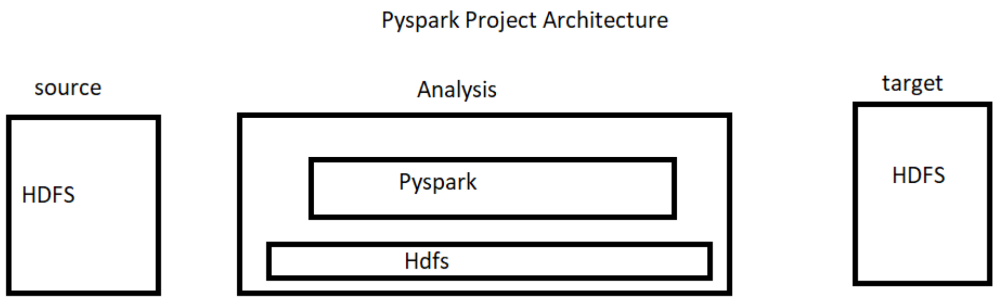
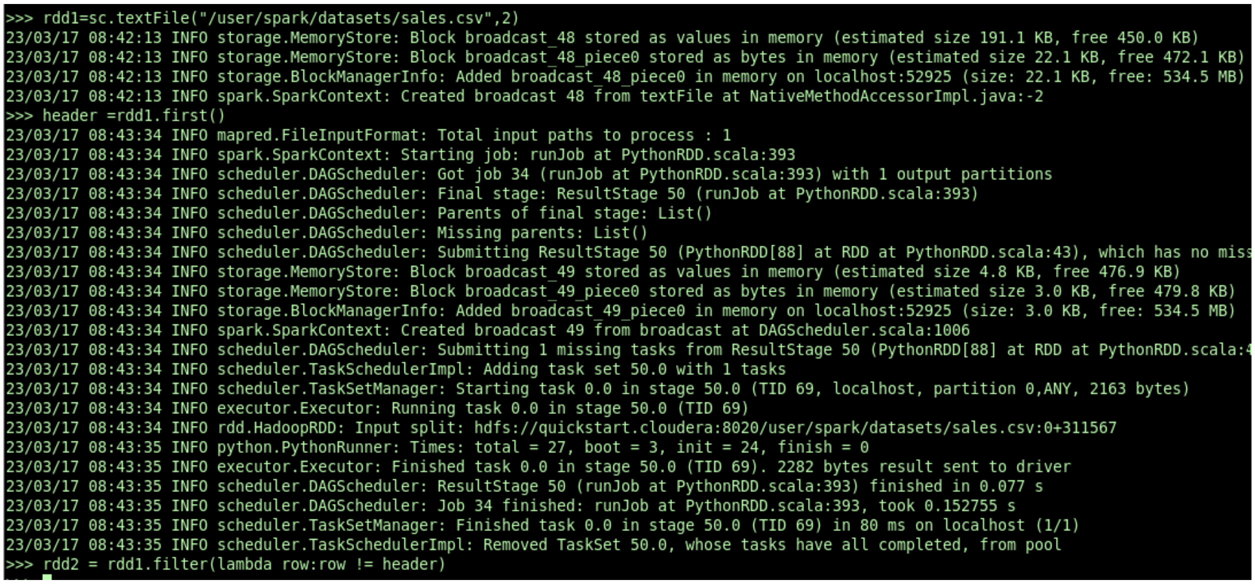
Pyspark Sales Analytics



# Requirement:
1. Display country wise number of orders
2.Display the number of units sold in each region.
3.Display the 10 most recent sales.
4.Display the products with atleast 2 occurences of 'a'
5.Display country in each region with highest units sold.
6.Display the unit price and unit cost of each item in ascending order.
7.Display the number of sales yearwise.
8.Display the number of orders for each item.

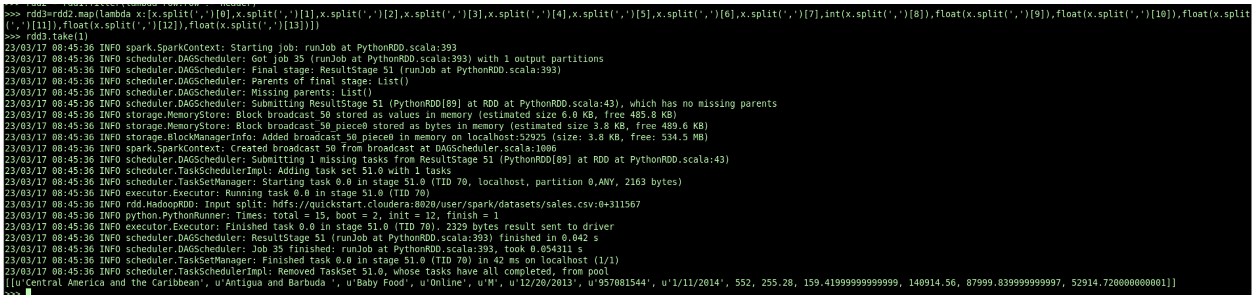
Loading sales Dataset and create RDD

```
rdd1=sc.textFile("/user/spark/datasets/sales.csv",2)

#extract header
header =rdd1.first()

#filtering records without header
rdd2 = rdd1.filter(lambda row:row != header)
```



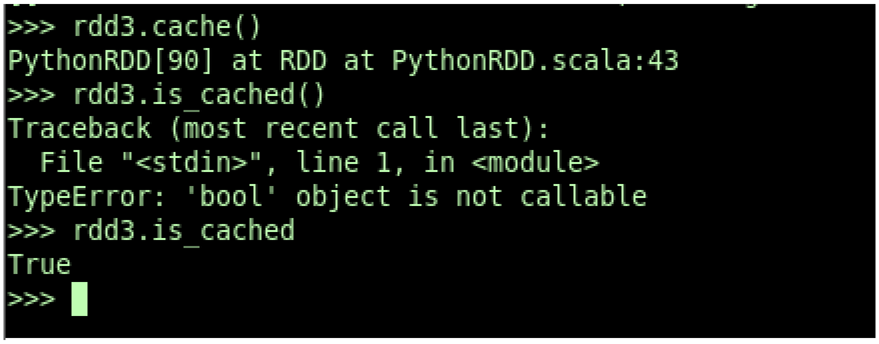
```
>>> rdd1=sc.textFile("/user/spark/datasets/sales.csv",2)
23/03/17 08:42:13 INFO storage.MemoryStore: Block broadcast_48 stored as values in memory (estimated size 191.1 KB, free 450.0 KB)
23/03/17 08:42:13 INFO storage.MemoryStore: Block broadcast_48_piece0 stored as bytes in memory (estimated size 22.1 KB, free 472.1 KB)
23/03/17 08:42:13 INFO storage.BlockManagerInfo: Added broadcast_48_piece0 in memory on localhost:52925 (size: 22.1 KB, free: 534.5 MB)
23/03/17 08:42:13 INFO spark.SparkContext: Created broadcast 48 from textFile at NativeMethodAccessorImpl.java:-2
>>> header =rdd1.first()
23/03/17 08:43:34 INFO mapred.FileInputFormat: Total input paths to process : 1
23/03/17 08:43:34 INFO spark.SparkContext: Starting job: runJob at PythonRDD.scala:393
23/03/17 08:43:34 INFO scheduler.DAGScheduler: Got job 34 (runJob at PythonRDD.scala:393) with 1 output partitions
23/03/17 08:43:34 INFO scheduler.DAGScheduler: Final stage: ResultStage 50 (runJob at PythonRDD.scala:393)
23/03/17 08:43:34 INFO scheduler.DAGScheduler: Parents of final stage: List()
23/03/17 08:43:34 INFO scheduler.DAGScheduler: Missing parents: List()
23/03/17 08:43:34 INFO scheduler.DAGScheduler: Submitting ResultStage 50 (PythonRDD[88] at RDD at PythonRDD.scala:43), which has no miss
23/03/17 08:43:34 INFO storage.MemoryStore: Block broadcast_49 stored as values in memory (estimated size 4.8 KB, free 476.9 KB)
23/03/17 08:43:34 INFO storage.MemoryStore: Block broadcast_49_piece0 stored as bytes in memory (estimated size 3.0 KB, free 479.8 KB)
23/03/17 08:43:34 INFO storage.BlockManagerInfo: Added broadcast_49_piece0 in memory on localhost:52925 (size: 3.0 KB, free: 534.5 MB)
23/03/17 08:43:34 INFO spark.SparkContext: Created broadcast 49 from broadcast at DAGScheduler.scala:1006
23/03/17 08:43:34 INFO scheduler.DAGScheduler: Submitting 1 missing tasks from ResultStage 50 (PythonRDD[88] at RDD at PythonRDD.scala:4
23/03/17 08:43:34 INFO scheduler.TaskSchedulerImpl: Adding task set 50.0 with 1 tasks
23/03/17 08:43:34 INFO scheduler.TaskSetManager: Starting task 0.0 in stage 50.0 (TID 69, localhost, partition 0,ANY, 2163 bytes)
23/03/17 08:43:34 INFO executor.Executor: Running task 0.0 in stage 50.0 (TID 69)
23/03/17 08:43:34 INFO rdd.HadoopRDD: Input split: hdfs://quickstart.cloudera:8020/user/spark/datasets/sales.csv:0+311567
23/03/17 08:43:35 INFO python.PythonRunner: Times: total = 27, boot = 3, init = 24, finish = 0
23/03/17 08:43:35 INFO executor.Executor: Finished task 0.0 in stage 50.0 (TID 69). 2282 bytes result sent to driver
23/03/17 08:43:35 INFO scheduler.DAGScheduler: ResultStage 50 (runJob at PythonRDD.scala:393) finished in 0.077 s
23/03/17 08:43:35 INFO scheduler.DAGScheduler: Job 34 finished: runJob at PythonRDD.scala:393, took 0.152755 s
23/03/17 08:43:35 INFO scheduler.TaskSetManager: Finished task 0.0 in stage 50.0 (TID 69) in 80 ms on localhost (1/1)
23/03/17 08:43:35 INFO scheduler.TaskSchedulerImpl: Removed TaskSet 50.0, whose tasks have all completed, from pool
>>> rdd2 = rdd1.filter(lambda row:row != header)
```

```
# extracting required columns and type casting
rdd3=rdd2.map(lambda x:[x.split(',')[0],x.split(',')[1],x.split(',')[2],x.split(',')[3],x.split(',')[4],x.split(',')[5],x.split(',')[6],x.split(',')[7],int(x.split(',')[8]),float(x.split(',')[9]),float(x.split(',')[10]),float(x.split(',')[11]),float(x.split(',')[12]),float(x.split(',')[13])])
```



```
>>> rdd2 = rdd1.filter(lambda row:row != header)
>>> rdd3=rdd2.map(lambda x:[x.split(',')[0],x.split(',')[1],x.split(',')[2],x.split(',')[3],x.split(',')[4],x.split(',')[5],x.split(',')[6],x.split(',')[7],int(x.split(',')[8]),float(x.split(',')[9]),float(x.split(',')[10]),float(x.split
(',')[11]),float(x.split(',')[12]),float(x.split(',')[13])])
>>> rdd3.take(1)
23/03/17 08:45:36 INFO spark.SparkContext: Starting job: runJob at PythonRDD.scala:393
23/03/17 08:45:36 INFO scheduler.DAGScheduler: Got job 35 (runJob at PythonRDD.scala:393) with 1 output partitions
23/03/17 08:45:36 INFO scheduler.DAGScheduler: Final stage: ResultStage 51 (runJob at PythonRDD.scala:393)
23/03/17 08:45:36 INFO scheduler.DAGScheduler: Parents of final stage: List()
23/03/17 08:45:36 INFO scheduler.DAGScheduler: Missing parents: List()
23/03/17 08:45:36 INFO scheduler.DAGScheduler: Submitting ResultStage 51 (PythonRDD[89] at RDD at PythonRDD.scala:43), which has no missing parents
23/03/17 08:45:36 INFO storage.MemoryStore: Block broadcast_50 stored as values in memory (estimated size 6.0 KB, free 485.8 KB)
23/03/17 08:45:36 INFO storage.MemoryStore: Block broadcast_50_piece0 stored as bytes in memory (estimated size 3.8 KB, free 489.6 KB)
23/03/17 08:45:36 INFO storage.BlockManagerInfo: Added broadcast_50_piece0 in memory on localhost:52925 (size: 3.8 KB, free: 534.5 MB)
23/03/17 08:45:36 INFO spark.SparkContext: Created broadcast 50 from broadcast at DAGScheduler.scala:1006
23/03/17 08:45:36 INFO scheduler.DAGScheduler: Submitting 1 missing tasks from ResultStage 51 (PythonRDD[89] at RDD at PythonRDD.scala:43)
23/03/17 08:45:36 INFO scheduler.TaskSchedulerImpl: Adding task set 51.0 with 1 tasks
23/03/17 08:45:36 INFO scheduler.TaskSetManager: Starting task 0.0 in stage 51.0 (TID 70, localhost, partition 0,ANY, 2163 bytes)
23/03/17 08:45:36 INFO executor.Executor: Running task 0.0 in stage 51.0 (TID 70)
23/03/17 08:45:36 INFO rdd.HadoopRDD: Input split: hdfs://quickstart.cloudera:8020/user/spark/datasets/sales.csv:0+311567
23/03/17 08:45:36 INFO python.PythonRunner: Times: total = 15, boot = 2, init = 12, finish = 1
23/03/17 08:45:36 INFO executor.Executor: Finished task 0.0 in stage 51.0 (TID 70). 2329 bytes result sent to driver
23/03/17 08:45:36 INFO scheduler.DAGScheduler: ResultStage 51 (runJob at PythonRDD.scala:393) finished in 0.042 s
23/03/17 08:45:36 INFO scheduler.DAGScheduler: Job 35 finished: runJob at PythonRDD.scala:393, took 0.054311 s
23/03/17 08:45:36 INFO scheduler.TaskSetManager: Finished task 0.0 in stage 51.0 (TID 70) in 42 ms on localhost (1/1)
23/03/17 08:45:36 INFO scheduler.TaskSchedulerImpl: Removed TaskSet 51.0, whose tasks have all completed, from pool
[[u'Central America and the Caribbean', u'Antigua and Barbuda ', u'Baby Food', u'Online', u'M', u'12/20/2013', u'957081544', u'1/11/2014', 552, 255.28, 159.41999999999999, 140914.56, 87999.839999999997, 52914.720000000001]]
>>>
```

```
# caching the rdd3

rdd3.cache()
```

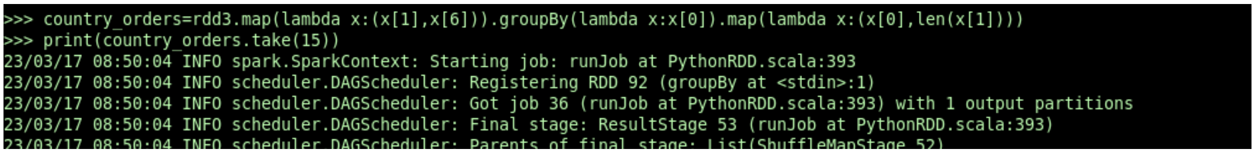


```
>>> rdd3.cache()
PythonRDD[90] at RDD at PythonRDD.scala:43
>>> rdd3.is_cached()
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: 'bool' object is not callable
>>> rdd3.is_cached
True
>>> 
```

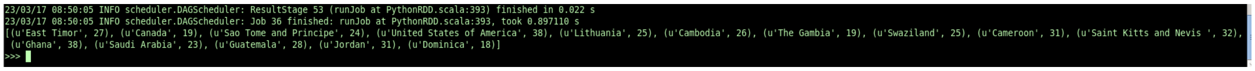
# 1. Display country wise number of orders

country_orders=rdd3.map(lambda x:(x[1],x[6])).groupBy(lambda x:x[0]).map(lambda x:(x[0],len(x[1])))

print(country_orders.take(15))



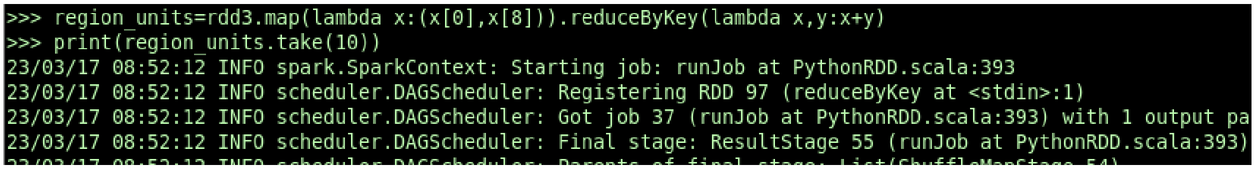
```
>>> country_orders=rdd3.map(lambda x:(x[1],x[6])).groupBy(lambda x:x[0]).map(lambda x:(x[0],len(x[1])))
>>> print(country_orders.take(15))
23/03/17 08:50:04 INFO spark.SparkContext: Starting job: runJob at PythonRDD.scala:393
23/03/17 08:50:04 INFO scheduler.DAGScheduler: Registering RDD 92 (groupBy at <stdin>:1)
23/03/17 08:50:04 INFO scheduler.DAGScheduler: Got job 36 (runJob at PythonRDD.scala:393) with 1 output partitions
23/03/17 08:50:04 INFO scheduler.DAGScheduler: Final stage: ResultStage 53 (runJob at PythonRDD.scala:393)
23/03/17 08:50:04 INFO scheduler.DAGScheduler: Parents of final stage: List(ShuffleMapStage 52)
```



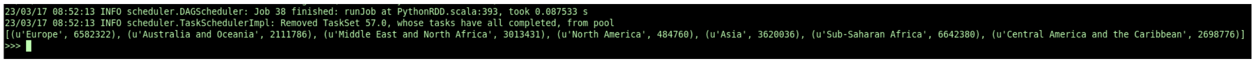
```
23/03/17 08:50:05 INFO scheduler.DAGScheduler: ResultStage 53 (runJob at PythonRDD.scala:393) finished in 0.022 s
23/03/17 08:50:05 INFO scheduler.DAGScheduler: Job 36 finished: runJob at PythonRDD.scala:393, took 0.897110 s
[(u'East Timor', 27), (u'Canada', 19), (u'Sao Tome and Principe', 24), (u'United States of America', 38), (u'Lithuania', 25), (u'Cambodia', 26), (u'The Gambia', 19), (u'Swaziland', 25), (u'Cameroon', 31), (u'Saint Kitts and Nevis ', 32),
 (u'Ghana', 38), (u'Saudi Arabia', 23), (u'Guatemala', 28), (u'Jordan', 31), (u'Dominica', 18)]
>>> 
```

# 2.Display the number of units sold in each region

region_units=rdd3.map(lambda x:(x[0],x[8])).reduceByKey(lambda x,y:x+y)
print(region_units.take(10))
region_units.coalesce(1).saveAsTextFile('/user/spark/region_units')



```
>>> region_units=rdd3.map(lambda x:(x[0],x[8])).reduceByKey(lambda x,y:x+y)
>>> print(region_units.take(10))
23/03/17 08:52:12 INFO spark.SparkContext: Starting job: runJob at PythonRDD.scala:393
23/03/17 08:52:12 INFO scheduler.DAGScheduler: Registering RDD 97 (reduceByKey at <stdin>:1)
23/03/17 08:52:12 INFO scheduler.DAGScheduler: Got job 37 (runJob at PythonRDD.scala:393) with 1 output pa
23/03/17 08:52:12 INFO scheduler.DAGScheduler: Final stage: ResultStage 55 (runJob at PythonRDD.scala:393)
23/03/17 08:52:12 INFO scheduler.DAGScheduler: Parents of final stage: List(ShuffleMapStage 54)
```

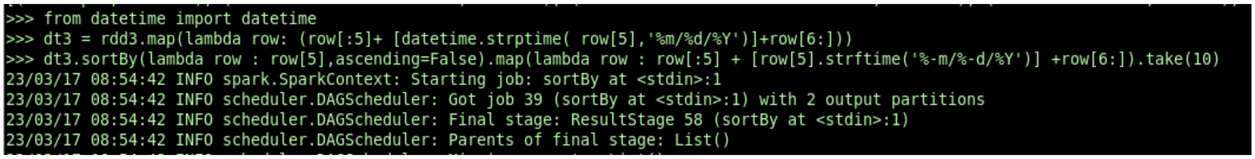


```
23/03/17 08:52:13 INFO scheduler.DAGScheduler: Job 38 finished: runJob at PythonRDD.scala:393, took 0.087533 s
23/03/17 08:52:13 INFO scheduler.TaskSchedulerImpl: Removed TaskSet 57.0, whose tasks have all completed, from pool
[(u'Europe', 6582322), (u'Australia and Oceania', 2111786), (u'Middle East and North Africa', 3013431), (u'North America', 484760), (u'Asia', 3620036), (u'Sub-Saharan Africa', 6642380), (u'Central America and the Caribbean', 2698776)]
>>> 
```

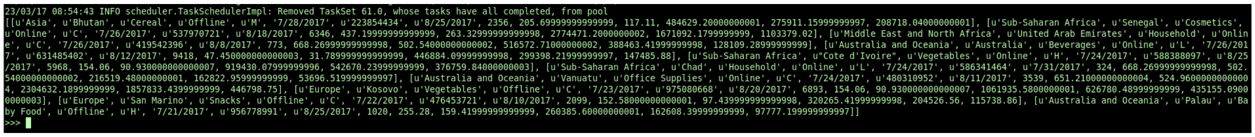
# 3.Display the 10 most recent sales.

from datetime import datetime

dt3 = rdd3.map(lambda row: (row[:5]+ [datetime.strptime(row[5],'%m/%d/%Y')]+row[6:]))
dt3.sortBy(lambda row : row[5],ascending=False).map(lambda row : row[:5] + [row[5].strftime('%-m/%-d/%Y')] +row[6:]).take(10)



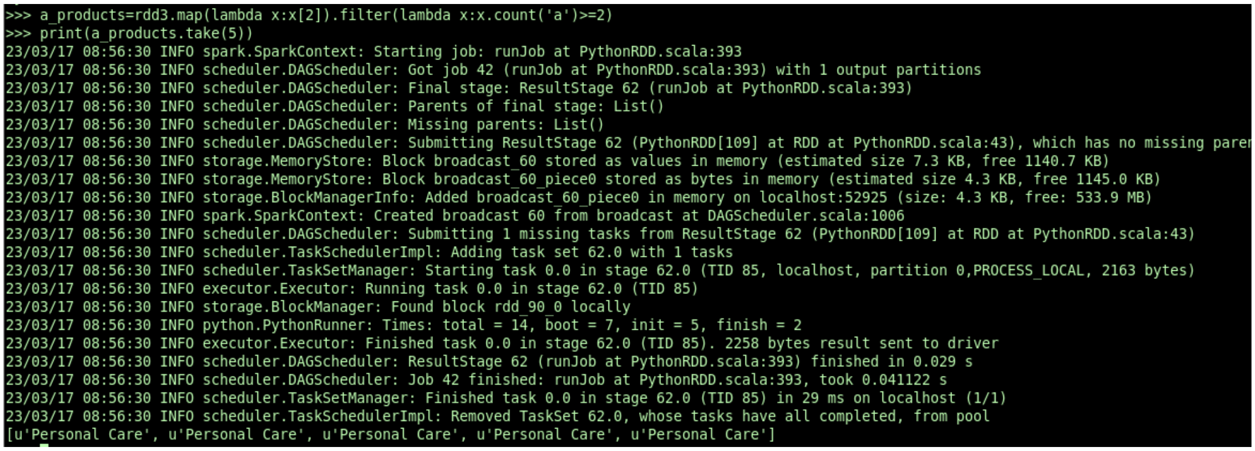
```
>>> from datetime import datetime
>>> dt3 = rdd3.map(lambda row: (row[:5]+ [datetime.strptime( row[5],'%m/%d/%Y')]+row[6:]))
>>> dt3.sortBy(lambda row : row[5],ascending=False).map(lambda row : row[:5] + [row[5].strftime('%-m/%-d/%Y')] +row[6:]).take(10)
23/03/17 08:54:42 INFO spark.SparkContext: Starting job: sortBy at <stdin>:1
23/03/17 08:54:42 INFO scheduler.DAGScheduler: Got job 39 (sortBy at <stdin>:1) with 2 output partitions
23/03/17 08:54:42 INFO scheduler.DAGScheduler: Final stage: ResultStage 58 (sortBy at <stdin>:1)
23/03/17 08:54:42 INFO scheduler.DAGScheduler: Parents of final stage: List()
```



```
23/03/17 08:54:43 INFO scheduler.TaskSchedulerImpl: Removed TaskSet 61.0, whose tasks have all completed, from pool
[[u'Asia', u'Bhutan', u'Cereal', u'Offline', u'M', '7/28/2017', u'223854434', u'8/25/2017', 2356, 205.69999999999999, 117.11, 484629.20000000001, 275911.15999999997, 208718.04000000001], [u'Sub-Saharan Africa', u'Senegal', u'Cosmetics',
u'Online', u'C', '7/26/2017', u'537970721', u'8/18/2017', 6346, 437.19999999999999, 263.32999999999998, 2774471.2000000002, 1671092.1799999999, 1103379.02], [u'Middle East and North Africa', u'United Arab Emirates', u'Household', u'Onlin
e', u'C', '7/26/2017', u'419542396', u'8/8/2017', 773, 668.26999999999998, 502.54000000000002, 516572.71000000002, 388463.41999999998, 128109.28999999999], [u'Australia and Oceania', u'Australia', u'Beverages', u'Online', u'L', '7/26/201
7', u'631485402', u'8/12/2017', 9418, 47.450000000000003, 31.789999999999999, 446884.09999999998, 299398.21999999997, 147485.88], [u'Sub-Saharan Africa', u"Cote d'Ivoire", u'Vegetables', u'Online', u'H', '7/24/2017', u'588388097', u'8/25
/2017', 5968, 154.06, 90.930000000000007, 919430.07999999996, 542670.23999999999, 376759.84000000003], [u'Sub-Saharan Africa', u'Chad', u'Household', u'Online', u'L', '7/24/2017', u'586341464', u'7/31/2017', 324, 668.26999999999998, 502.
54000000000002, 216519.48000000001, 162822.95999999999, 53696.519999999997], [u'Australia and Oceania', u'Vanuatu', u'Office Supplies', u'Online', u'C', '7/24/2017', u'480310952', u'8/11/2017', 3539, 651.21000000000004, 524.9600000000000
4, 2304632.1899999999, 1857833.4399999999, 446798.75], [u'Europe', u'Kosovo', u'Vegetables', u'Offline', u'C', '7/23/2017', u'975080668', u'8/20/2017', 6893, 154.06, 90.930000000000007, 1061935.5800000001, 626780.48999999999, 435155.0900
0000003], [u'Europe', u'San Marino', u'Snacks', u'Offline', u'C', '7/22/2017', u'476453721', u'8/10/2017', 2099, 152.58000000000001, 97.439999999999998, 320265.41999999998, 204526.56, 115738.86], [u'Australia and Oceania', u'Palau', u'Ba
by Food', u'Offline', u'H', '7/21/2017', u'956778991', u'8/25/2017', 1020, 255.28, 159.41999999999999, 260385.60000000001, 162608.39999999999, 97777.199999999997]]
>>> 
```

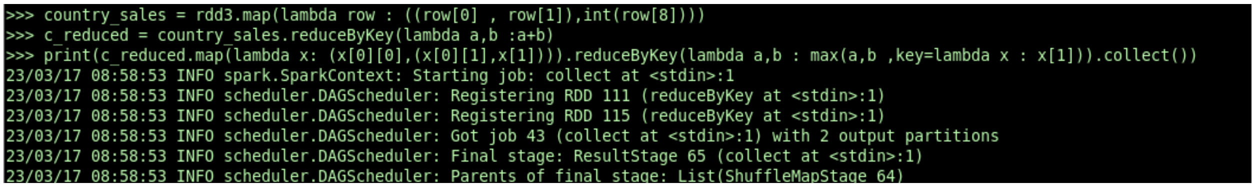
# 4.Display the products with atleast 2 occurences of 'a'

a_products=rdd3.map(lambda x:x[2]).filter(lambda x:x.count('a')>=2)
print(a_products.take(5))

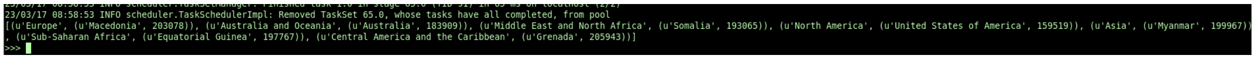


```
>>> a_products=rdd3.map(lambda x:x[2]).filter(lambda x:x.count('a')>=2)
>>> print(a_products.take(5))
23/03/17 08:56:30 INFO spark.SparkContext: Starting job: runJob at PythonRDD.scala:393
23/03/17 08:56:30 INFO scheduler.DAGScheduler: Got job 42 (runJob at PythonRDD.scala:393) with 1 output partitions
23/03/17 08:56:30 INFO scheduler.DAGScheduler: Final stage: ResultStage 62 (runJob at PythonRDD.scala:393)
23/03/17 08:56:30 INFO scheduler.DAGScheduler: Parents of final stage: List()
23/03/17 08:56:30 INFO scheduler.DAGScheduler: Missing parents: List()
23/03/17 08:56:30 INFO scheduler.DAGScheduler: Submitting ResultStage 62 (PythonRDD[109] at RDD at PythonRDD.scala:43), which has no missing paren
23/03/17 08:56:30 INFO storage.MemoryStore: Block broadcast_60 stored as values in memory (estimated size 7.3 KB, free 1140.7 KB)
23/03/17 08:56:30 INFO storage.MemoryStore: Block broadcast_60_piece0 stored as bytes in memory (estimated size 4.3 KB, free 1145.0 KB)
23/03/17 08:56:30 INFO storage.BlockManagerInfo: Added broadcast_60_piece0 in memory on localhost:52925 (size: 4.3 KB, free: 533.9 MB)
23/03/17 08:56:30 INFO spark.SparkContext: Created broadcast 60 from broadcast at DAGScheduler.scala:1006
23/03/17 08:56:30 INFO scheduler.DAGScheduler: Submitting 1 missing tasks from ResultStage 62 (PythonRDD[109] at RDD at PythonRDD.scala:43)
23/03/17 08:56:30 INFO scheduler.TaskSchedulerImpl: Adding task set 62.0 with 1 tasks
23/03/17 08:56:30 INFO scheduler.TaskSetManager: Starting task 0.0 in stage 62.0 (TID 85, localhost, partition 0,PROCESS_LOCAL, 2163 bytes)
23/03/17 08:56:30 INFO executor.Executor: Running task 0.0 in stage 62.0 (TID 85)
23/03/17 08:56:30 INFO storage.BlockManager: Found block rdd_90_0 locally
23/03/17 08:56:30 INFO python.PythonRunner: Times: total = 14, boot = 7, init = 5, finish = 2
23/03/17 08:56:30 INFO executor.Executor: Finished task 0.0 in stage 62.0 (TID 85). 2258 bytes result sent to driver
23/03/17 08:56:30 INFO scheduler.DAGScheduler: ResultStage 62 (runJob at PythonRDD.scala:393) finished in 0.029 s
23/03/17 08:56:30 INFO scheduler.DAGScheduler: Job 42 finished: runJob at PythonRDD.scala:393, took 0.041122 s
23/03/17 08:56:30 INFO scheduler.TaskSetManager: Finished task 0.0 in stage 62.0 (TID 85) in 29 ms on localhost (1/1)
23/03/17 08:56:30 INFO scheduler.TaskSchedulerImpl: Removed TaskSet 62.0, whose tasks have all completed, from pool
[u'Personal Care', u'Personal Care', u'Personal Care', u'Personal Care', u'Personal Care']
```

#5.Display country in each region with highest units sold. (Using spark)
country_sales = rdd3.map(lambda row : ((row[0] , row[1]),int(row[8])))
c_reduced = country_sales.reduceByKey(lambda a,b :a+b)
print(c_reduced.map(lambda x: (x[0][0],(x[0][1],x[1]))).reduceByKey(lambda a,b : max(a,b ,key=lambda x : x[1])).collect())



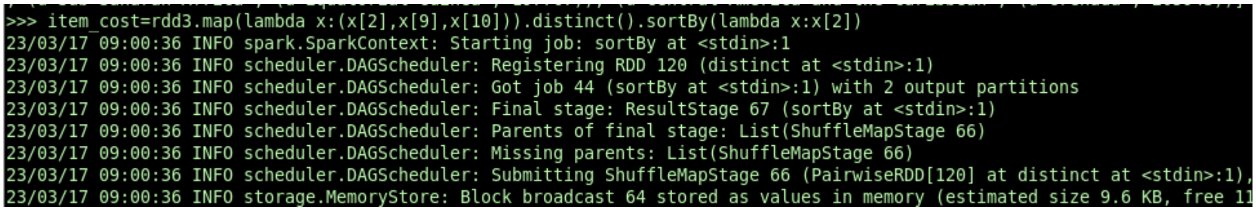
```
>>> country_sales = rdd3.map(lambda row : ((row[0] , row[1]),int(row[8])))
>>> c_reduced = country_sales.reduceByKey(lambda a,b :a+b)
>>> print(c_reduced.map(lambda x: (x[0][0],(x[0][1],x[1]))).reduceByKey(lambda a,b : max(a,b ,key=lambda x : x[1])).collect())
23/03/17 08:58:53 INFO spark.SparkContext: Starting job: collect at <stdin>:1
23/03/17 08:58:53 INFO scheduler.DAGScheduler: Registering RDD 111 (reduceByKey at <stdin>:1)
23/03/17 08:58:53 INFO scheduler.DAGScheduler: Registering RDD 115 (reduceByKey at <stdin>:1)
23/03/17 08:58:53 INFO scheduler.DAGScheduler: Got job 43 (collect at <stdin>:1) with 2 output partitions
23/03/17 08:58:53 INFO scheduler.DAGScheduler: Final stage: ResultStage 65 (collect at <stdin>:1)
23/03/17 08:58:53 INFO scheduler.DAGScheduler: Parents of final stage: List(ShuffleMapStage 64)
```



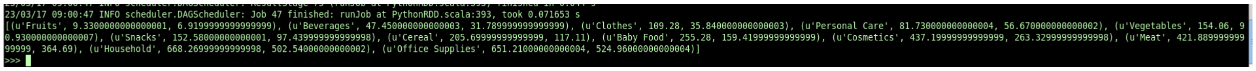
```
23/03/17 08:58:53 INFO scheduler.TaskSchedulerImpl: Removed TaskSet 65.0, whose tasks have all completed, from pool
[(u'Europe', (u'Macedonia', 203078)), (u'Australia and Oceania', (u'Australia', 183909)), (u'Middle East and North Africa', (u'Somalia', 193065)), (u'North America', (u'United States of America', 159519)), (u'Asia', (u'Myanmar', 199967))
, (u'Sub-Saharan Africa', (u'Equatorial Guinea', 197767)), (u'Central America and the Caribbean', (u'Grenada', 205943))]
>>> 
```

# 6.Display the unit price and unit cost of each item in ascending order.

item_cost=rdd3.map(lambda x:(x[2],x[9],x[10])).distinct().sortBy(lambda x:x[2])
print(item_cost.take(20))



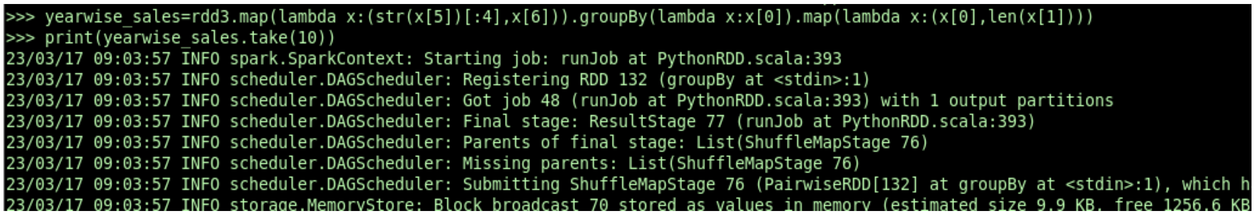
```
>>> item_cost=rdd3.map(lambda x:(x[2],x[9],x[10])).distinct().sortBy(lambda x:x[2])
23/03/17 09:00:36 INFO spark.SparkContext: Starting job: sortBy at <stdin>:1
23/03/17 09:00:36 INFO scheduler.DAGScheduler: Registering RDD 120 (distinct at <stdin>:1)
23/03/17 09:00:36 INFO scheduler.DAGScheduler: Got job 44 (sortBy at <stdin>:1) with 2 output partitions
23/03/17 09:00:36 INFO scheduler.DAGScheduler: Final stage: ResultStage 67 (sortBy at <stdin>:1)
23/03/17 09:00:36 INFO scheduler.DAGScheduler: Parents of final stage: List(ShuffleMapStage 66)
23/03/17 09:00:36 INFO scheduler.DAGScheduler: Missing parents: List(ShuffleMapStage 66)
23/03/17 09:00:36 INFO scheduler.DAGScheduler: Submitting ShuffleMapStage 66 (PairwiseRDD[120] at distinct at <stdin>:1),
23/03/17 09:00:36 INFO storage.MemoryStore: Block broadcast 64 stored as values in memory (estimated size 9.6 KB, free 11
```



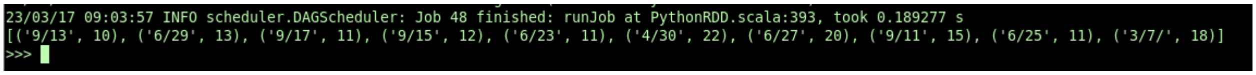
```
23/03/17 09:00:47 INFO scheduler.DAGScheduler: Job 47 finished: runJob at PythonRDD.scala:393, took 0.071653 s
[(u'Fruits', 9.3300000000000001, 6.9199999999999999), (u'Beverages', 47.450000000000003, 31.789999999999999), (u'Clothes', 109.28, 35.840000000000003), (u'Personal Care', 81.730000000000004, 56.670000000000002), (u'Vegetables', 154.06, 9
0.930000000000007), (u'Snacks', 152.58000000000001, 97.439999999999998), (u'Cereal', 205.69999999999999, 117.11), (u'Baby Food', 255.28, 159.41999999999999), (u'Cosmetics', 437.19999999999999, 263.32999999999998), (u'Meat', 421.889999999
99999, 364.69), (u'Household', 668.26999999999998, 502.54000000000002), (u'Office Supplies', 651.21000000000004, 524.96000000000004)]
>>> 
```

# 7.Display the number of sales yearwise.

yearwise_sales=rdd3.map(lambda x:(str(x[5])[:4],x[6])).groupBy(lambda x:x[0]).map(lambda x:(x[0],len(x[1])))
print(yearwise_sales.take(10))



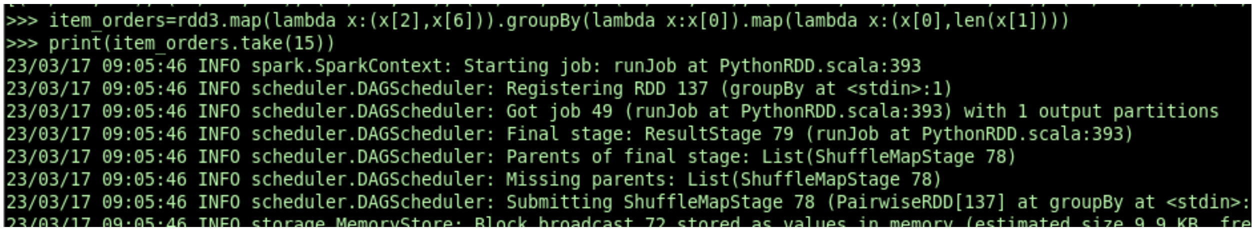
```
>>> yearwise_sales=rdd3.map(lambda x:(str(x[5])[:4],x[6])).groupBy(lambda x:x[0]).map(lambda x:(x[0],len(x[1])))
>>> print(yearwise_sales.take(10))
23/03/17 09:03:57 INFO spark.SparkContext: Starting job: runJob at PythonRDD.scala:393
23/03/17 09:03:57 INFO scheduler.DAGScheduler: Registering RDD 132 (groupBy at <stdin>:1)
23/03/17 09:03:57 INFO scheduler.DAGScheduler: Got job 48 (runJob at PythonRDD.scala:393) with 1 output partitions
23/03/17 09:03:57 INFO scheduler.DAGScheduler: Final stage: ResultStage 77 (runJob at PythonRDD.scala:393)
23/03/17 09:03:57 INFO scheduler.DAGScheduler: Parents of final stage: List(ShuffleMapStage 76)
23/03/17 09:03:57 INFO scheduler.DAGScheduler: Missing parents: List(ShuffleMapStage 76)
23/03/17 09:03:57 INFO scheduler.DAGScheduler: Submitting ShuffleMapStage 76 (PairwiseRDD[132] at groupBy at <stdin>:1), which h
23/03/17 09:03:57 INFO storage.MemoryStore: Block broadcast 70 stored as values in memory (estimated size 9.9 KB, free 1256.6 KB
```



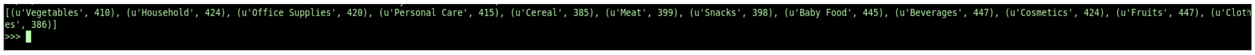
```
23/03/17 09:03:57 INFO scheduler.DAGScheduler: Job 48 finished: runJob at PythonRDD.scala:393, took 0.189277 s
[('9/13', 10), ('6/29', 13), ('9/17', 11), ('9/15', 12), ('6/23', 11), ('4/30', 22), ('6/27', 20), ('9/11', 15), ('6/25', 11), ('3/7/', 18)]
>>> 
```

# 8.Display the number of orders for each item.

item_orders=rdd3.map(lambda x:(x[2],x[6])).groupBy(lambda x:x[0]).map(lambda x:(x[0],len(x[1])))
print(item_orders.take(15))



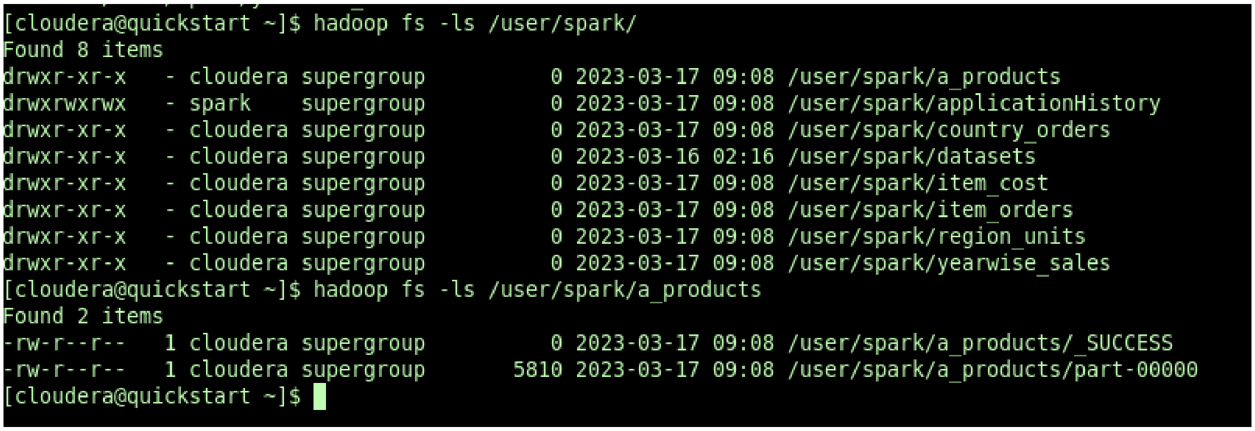
```
>>> item_orders=rdd3.map(lambda x:(x[2],x[6])).groupBy(lambda x:x[0]).map(lambda x:(x[0],len(x[1])))
>>> print(item_orders.take(15))
23/03/17 09:05:46 INFO spark.SparkContext: Starting job: runJob at PythonRDD.scala:393
23/03/17 09:05:46 INFO scheduler.DAGScheduler: Registering RDD 137 (groupBy at <stdin>:1)
23/03/17 09:05:46 INFO scheduler.DAGScheduler: Got job 49 (runJob at PythonRDD.scala:393) with 1 output partitions
23/03/17 09:05:46 INFO scheduler.DAGScheduler: Final stage: ResultStage 79 (runJob at PythonRDD.scala:393)
23/03/17 09:05:46 INFO scheduler.DAGScheduler: Parents of final stage: List(ShuffleMapStage 78)
23/03/17 09:05:46 INFO scheduler.DAGScheduler: Missing parents: List(ShuffleMapStage 78)
23/03/17 09:05:46 INFO scheduler.DAGScheduler: Submitting ShuffleMapStage 78 (PairwiseRDD[137] at groupBy at <stdin>:
23/03/17 09:05:46 INFO storage.MemoryStore: Block broadcast 72 stored as values in memory (estimated size 9.9 KB, fre
```



```
[(u'Vegetables', 410), (u'Household', 424), (u'Office Supplies', 420), (u'Personal Care', 415), (u'Cereal', 385), (u'Meat', 399), (u'Snacks', 398), (u'Baby Food', 445), (u'Beverages', 447), (u'Cosmetics', 424), (u'Fruits', 447), (u'Cloth
es', 386)]
>>> 
```

Storing the analysis results as text files in hdfs

```
country_orders.saveAsTextFile('/user/spark/country_orders')

region_units.coalesce(1).saveAsTextFile('/user/spark/region_units')

a_products.coalesce(1).saveAsTextFile('/user/spark/a_products')

item_cost.coalesce(1).saveAsTextFile('/user/spark/item_cost')

yearwise_sales.coalesce(1).saveAsTextFile('/user/spark/yearwise_sales')

item_orders.coalesce(1).saveAsTextFile('/user/spark/item_orders')
```



```
[cloudera@quickstart ~]$ hadoop fs -ls /user/spark/
Found 8 items
drwxr-xr-x   - cloudera supergroup          0 2023-03-17 09:08 /user/spark/a_products
drwxrwxrwx   - spark    supergroup          0 2023-03-17 09:08 /user/spark/applicationHistory
drwxr-xr-x   - cloudera supergroup          0 2023-03-17 09:08 /user/spark/country_orders
drwxr-xr-x   - cloudera supergroup          0 2023-03-16 02:16 /user/spark/datasets
drwxr-xr-x   - cloudera supergroup          0 2023-03-17 09:08 /user/spark/item_cost
drwxr-xr-x   - cloudera supergroup          0 2023-03-17 09:08 /user/spark/item_orders
drwxr-xr-x   - cloudera supergroup          0 2023-03-17 09:08 /user/spark/region_units
drwxr-xr-x   - cloudera supergroup          0 2023-03-17 09:08 /user/spark/yearwise_sales
[cloudera@quickstart ~]$ hadoop fs -ls /user/spark/a_products
Found 2 items
-rw-r--r--   1 cloudera supergroup          0 2023-03-17 09:08 /user/spark/a_products/_SUCCESS
-rw-r--r--   1 cloudera supergroup       5810 2023-03-17 09:08 /user/spark/a_products/part-00000
[cloudera@quickstart ~]$
```