Ecommerce Data – An 360° View

Date: 24/11/2016

Developer: Oliver Prasanna R

Objective: To Create an Easy & Customized Report for an Organization which becomes an easy tool for every employees to visualize the Organizational Performance as well as their Individual Contribution.

Abstract: Given Set of Test data based on Individual Transaction made by the Customer.

1. **Transaction data** – Obtained on Every User Individual Progress.

TransactionID	DOT	UserID	Amount	Category	Accessories	Place	City	Payment Mode
0	06-26-2015	4007024	40.33	Exercise & Fitness	Cardio Machine Accessories	Clarksville	Tennessee	credit
1	05-26-2015	4006742	198.44	Exercise & Fitness	Weightlifting Gloves	Long Beach	California	credit

2. **Customer Data** - A Secondary table stored as a result of Transaction.

CID	Fname	Lname	Age	Profession
4000001	Kristina	Chung	55	Pilot
4000002	Paige	Chen	74	Teacher

Hardware Requirements:

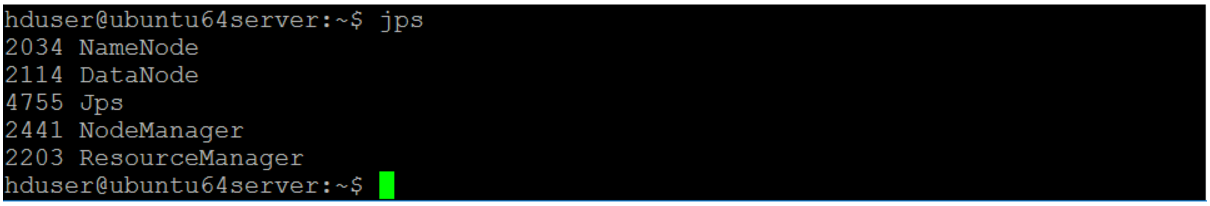
- 8 GB RAM
- 64 Bit OS

Software Requirements:

- Oracle Virtual Box
- Ubuntu
- Hadoop
- Putty

Assumptions:

- Oracle Virtual Box – Configurations are set correctly.
- Ubuntu is lying on the Virtual Box and it is powered on
- Putty is configured with the IP address of Ubuntu.
- Hadoop Folder must be extracted and all the services of the hadoop is running. Configuration to be made in the XML are set.
- Confirmation Box Below that Everything is Set Right .



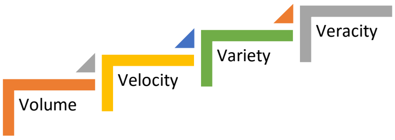
```
hduser@ubuntu64server:~$ jps
2034 NameNode
2114 DataNode
4755 Jps
2441 NodeManager
2203 ResourceManager
hduser@ubuntu64server:~$
```

A Brief Introduction about Various Technologies used in our Project:

Big Data:

Big data is a term for data sets that are so large or complex that traditional data processing applications are inadequate to deal with them.

4 V's of Big Data:



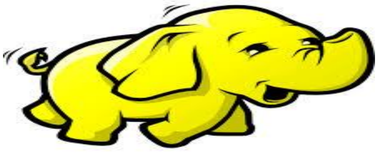
Volume

Velocity

Variety

Veracity

Hadoop



- **Apache Hadoop** is an open-source software framework used for distributed storage and processing of very large data sets.
- It consists of computer clusters built from commodity hardware.
- All the modules in Hadoop are designed with a fundamental assumption that hardware failures are a common occurrence and should be automatically handled by the framework.

HIVE

- **Apache Hive** is a data warehouse infrastructure built on top of Hadoop for providing data summarization, query, and analysis
- Hive gives an SQL-like interface to query data stored in various databases and file systems that integrate with Hadoop.

PIG

- **Apache Pig** is a high-level platform for creating programs that run on Apache Hadoop. The language for this platform is called **Pig Latin**.
- Pig can execute its Hadoop jobs in Map Reduce, Apache Tez, or Apache Spark. Pig Latin abstracts the programming from the Java Map Reduce idiom into a notation which makes Map Reduce programming high level, similar to that of SQL for RDBMSs
- Pig Latin can be extended using User Defined Functions (UDFs) which the user can write in Java, Python, JavaScript, Ruby or Groovy[2] and then call directly from the language.

Use – Case Generation

Use Case 1: ***Constraint Based Amount Scenario***

- **Input**: Amount [Custom Input from the User]
- **Key**: ID, **Value**: Amount
- **Output**: Set of User ID and Amount Based on the Input provided by the user
- **Data Validation**: Yes.

  Constraint: User Input Can be Only Numbers
- **Concept used:** Advanced Map Reduce, HIVE and PIG.

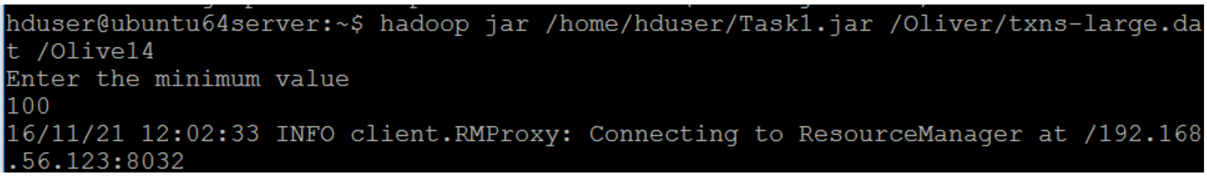
Description: We need to find the Set of User ID and their Associated Amount Based on the Amount Specified by the User.

Why this Report: To find the set of customers who does a purchase for a minimum amount comparing to the standards set by the organization.

Progress: Organization will have an opportunity for creating a new offers or benefits of these set of customers. These benefits can be either through mail or message. The benefit offers can be also displayed to the customer once they open their authenticated site.

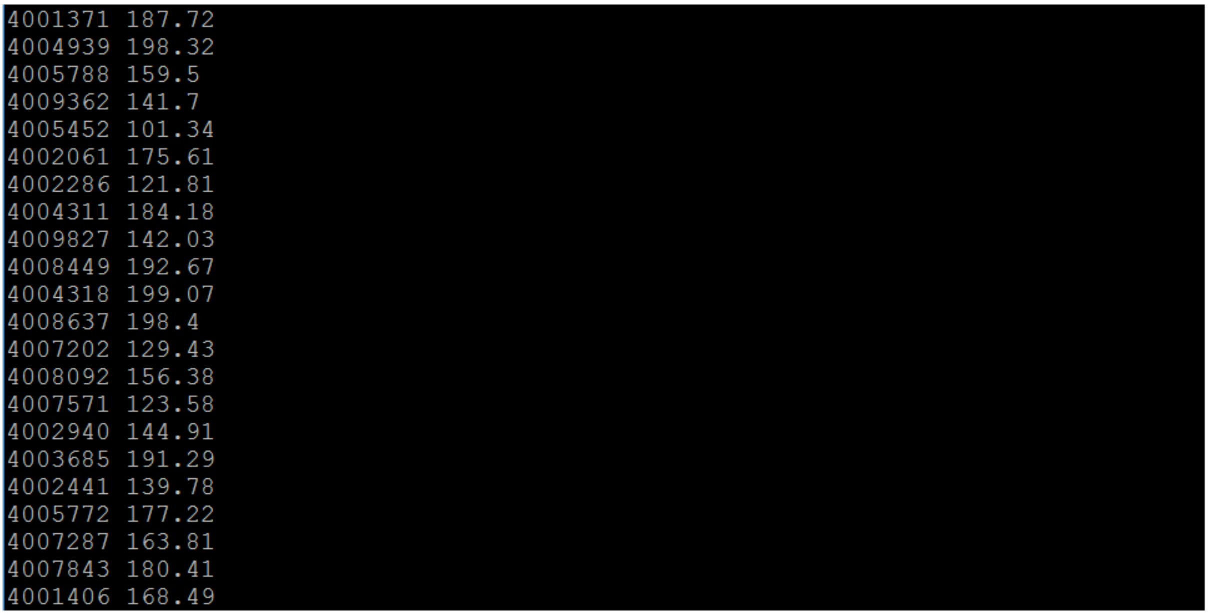
1.1 Advanced Map Reduce:

Screen Shot 1.1.1: Input Window



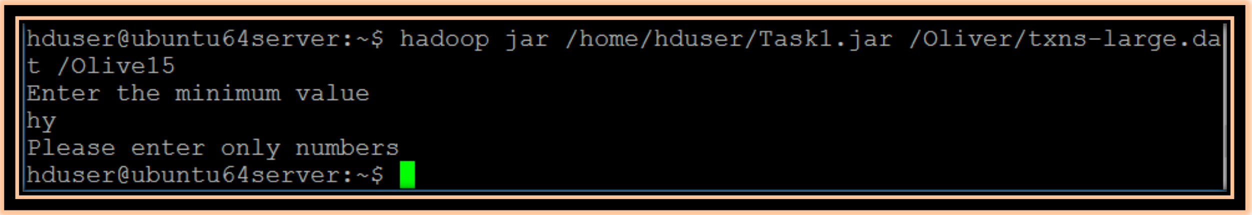
```
hduser@ubuntu64server:~$ hadoop jar /home/hduser/Task1.jar /Oliver/txns-large.da
t /Olive14
Enter the minimum value
100
16/11/21 12:02:33 INFO client.RMProxy: Connecting to ResourceManager at /192.168
.56.123:8032
```

Screen Shot 1.1.2: Output Window



```
4001371 187.72
4004939 198.32
4005788 159.5
4009362 141.7
4005452 101.34
4002061 175.61
4002286 121.81
4004311 184.18
4009827 142.03
4008449 192.67
4004318 199.07
4008637 198.4
4007202 129.43
4008092 156.38
4007571 123.58
4002940 144.91
4003685 191.29
4002441 139.78
4005772 177.22
4007287 163.81
4007843 180.41
4001406 168.49
```

Screen Shot 1.1.2: Data Validation



```
hduser@ubuntu64server:~$ hadoop jar /home/hduser/Task1.jar /Oliver/txns-large.da
t /Olive15
Enter the minimum value
hy
Please enter only numbers
hduser@ubuntu64server:~$
```

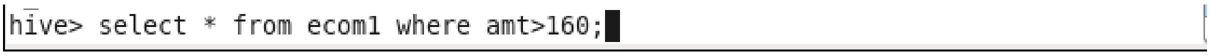
Output Path:

hadoop fs -cat /Olive15/part-m-00000

1.2 HIVE:

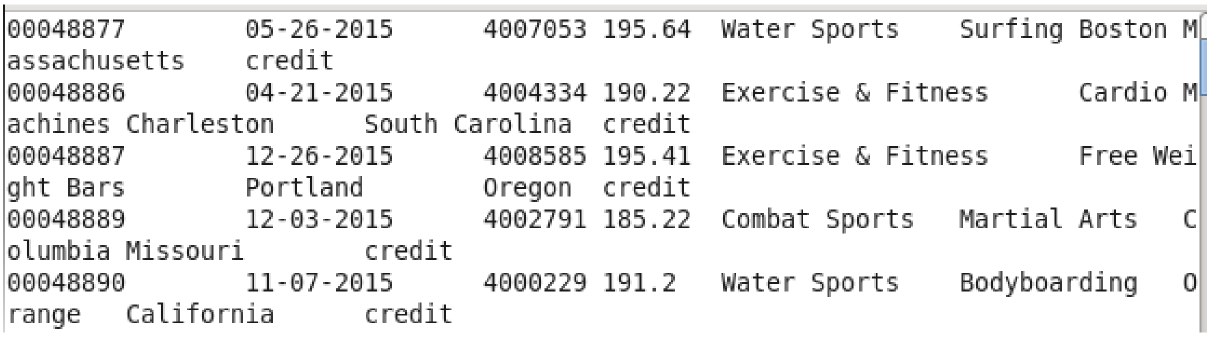
Screen Shot 1.2.1:

HiveQL: select * from Ecom1 where amt>160;



```
hive> select * from ecom1 where amt>160;
```

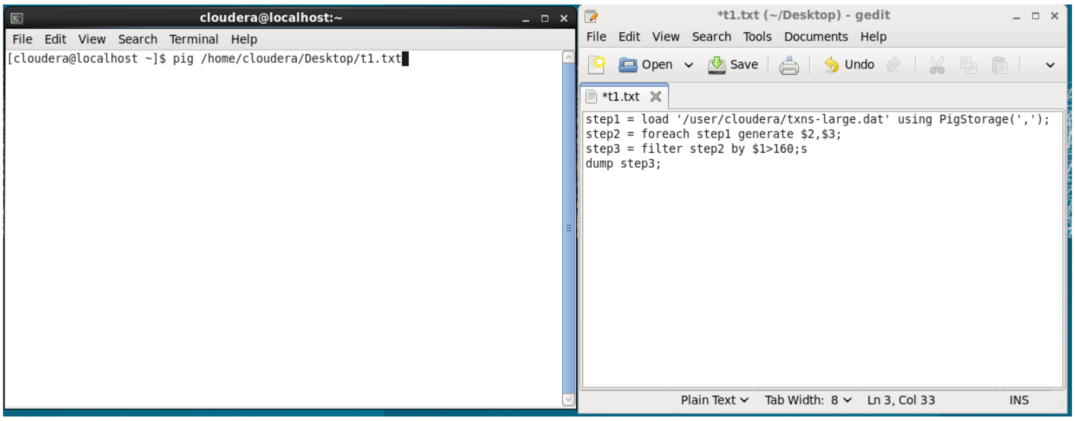
HiveQL Result Window:



```
00048877        05-26-2015      4007053 195.64  Water Sports    Surfing Boston M
assachusetts    credit
00048886        04-21-2015      4004334 190.22  Exercise & Fitness      Cardio M
achines Charleston      South Carolina  credit
00048887        12-26-2015      4008585 195.41  Exercise & Fitness      Free Wei
ght Bars        Portland        Oregon  credit
00048889        12-03-2015      4002791 185.22  Combat Sports   Martial Arts    C
olumbia Missouri        credit
00048890        11-07-2015      4000229 191.2   Water Sports    Bodyboarding    O
range   California      credit
```

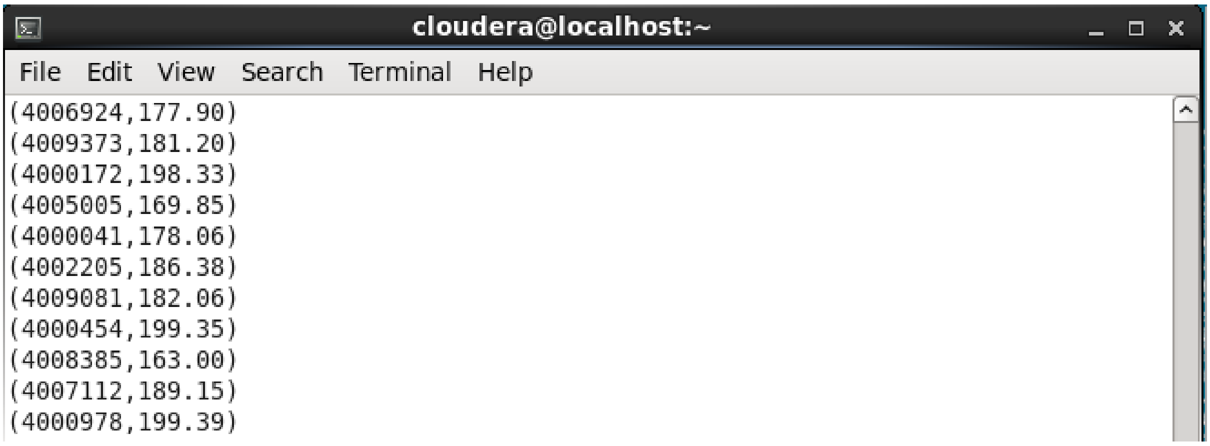
1.3 PIG:

Screen Shot 1.3.1:


cloudera@localhost:~
File Edit View Search Terminal Help
*t1.txt (~/Desktop) - gedit
File Edit View Search Tools Documents Help
Open
Save
Undo
*t1.txt
Plain Text
Tab Width: 8
Ln 3, Col 33
INS

```
[cloudera@localhost ~]$ pig /home/cloudera/Desktop/t1.txt
```

```
step1 = load '/user/cloudera/txns-large.dat' using PigStorage(',');
step2 = foreach step1 generate $2,$3;
step3 = filter step2 by $1>160;s
dump step3;
```



```
(4006924,177.90)
(4009373,181.20)
(4000172,198.33)
(4005005,169.85)
(4000041,178.06)
(4002205,186.38)
(4009081,182.06)
(4000454,199.35)
(4008385,163.00)
(4007112,189.15)
(4000978,199.39)
```

Use Case 2: *A Single Spot with a Range*

- **Input File**: txns-large.dat
- **Input:** Upper and Lower Limit [Custom Input from the User]
- **Key:** ID, Name **Value:** Amount
- **Output:** A Single Count that match the Range
- **Data Validation:** Yes.

  Constraint: User Input Can be Only Numbers
- **Concept :** Advanced Map Reduce, Hive & PIG

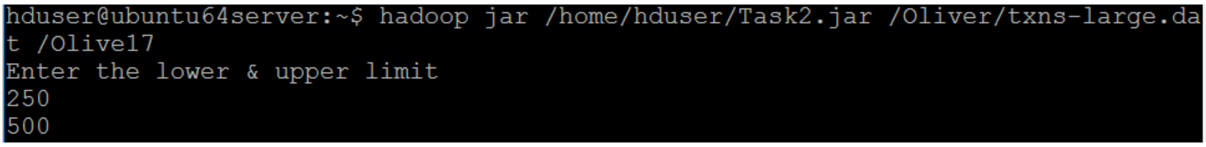
Description: To Obtain the Exact Count of number of Amount Transaction within a Particular Range

Why this Report: To find a number made in Sales.

Progress: Report from this can be used by Sales Manager. The report an exact view of the number of Transaction made in a particular limit range. If in a particular range, the sales count is less, then the Manager should move a step forward to identify the gap, and plan a mechanism to increase sales.

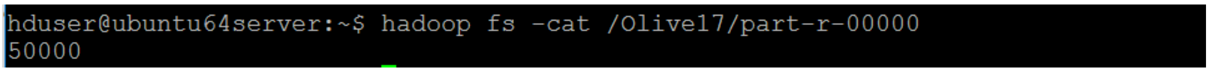
2.1 Advanced Map Reduce:

Screen Shot 2.1.1: Input Window



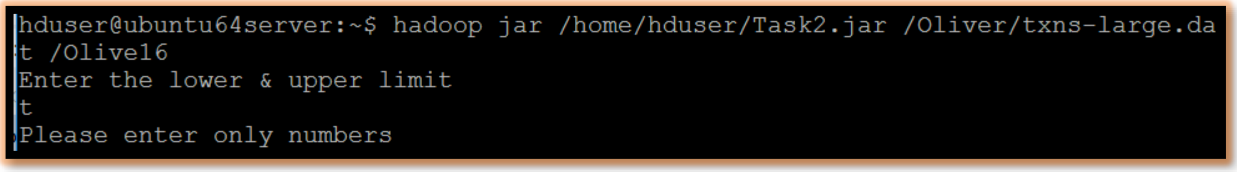
```
hduser@ubuntu64server:~$ hadoop jar /home/hduser/Task2.jar /Oliver/txns-large.da
t /Olive17
Enter the lower & upper limit
250
500
```

Screen Shot 2.1.2: Output Window



```
hduser@ubuntu64server:~$ hadoop fs -cat /Olive17/part-r-00000
50000
```

Screen Shot 2.1.3: Data Validation



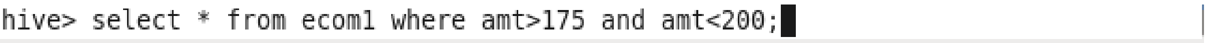
```
hduser@ubuntu64server:~$ hadoop jar /home/hduser/Task2.jar /Oliver/txns-large.da
t /Olive16
Enter the lower & upper limit
t
Please enter only numbers
```

Output Path:

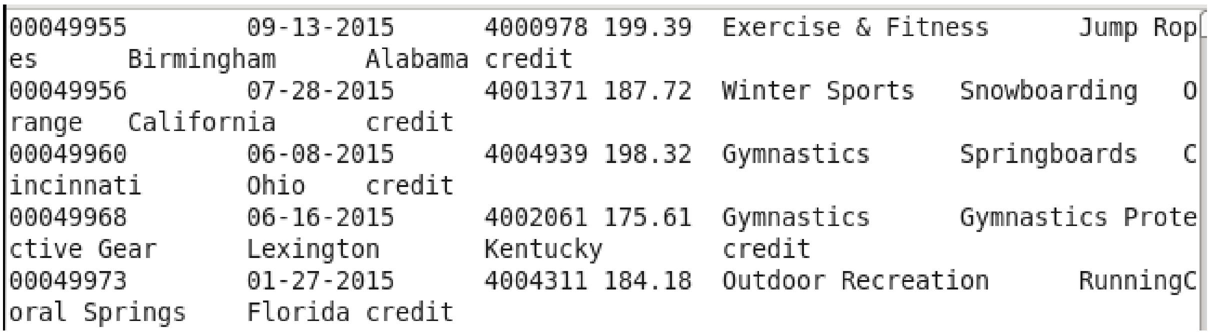
Hadoop fs –cat /Olive17/part-r-00000

2.2 HIVE

Screen Shot 2.2.1:



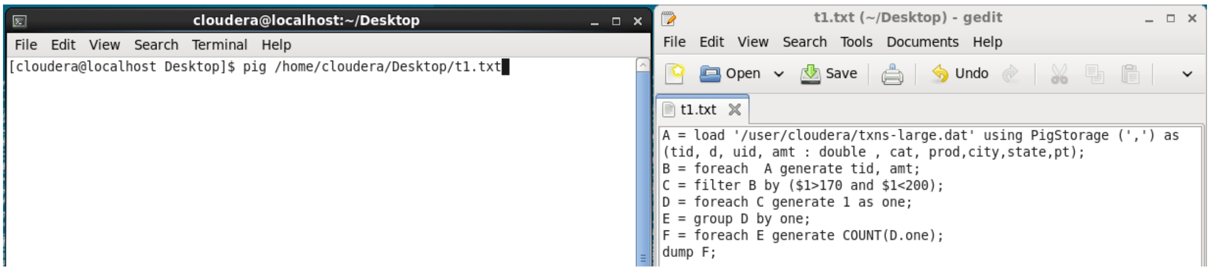
```
hive> select * from ecom1 where amt>175 and amt<200;
```



```
00049955        09-13-2015      4000978 199.39  Exercise & Fitness      Jump Rop
es      Birmingham      Alabama credit
00049956        07-28-2015      4001371 187.72  Winter Sports   Snowboarding    O
range   California      credit
00049960        06-08-2015      4004939 198.32  Gymnastics      Springboards    C
incinnati       Ohio    credit
00049968        06-16-2015      4002061 175.61  Gymnastics      Gymnastics Prote
ctive Gear      Lexington       Kentucky        credit
00049973        01-27-2015      4004311 184.18  Outdoor Recreation      RunningC
oral Springs    Florida credit
```

2.3 PIG

Screen Shot 2.3.1:

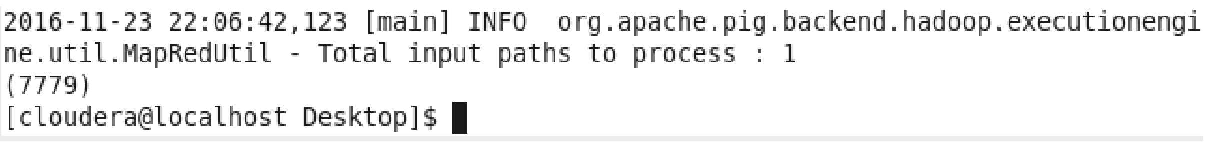


cloudera@localhost:~/Desktop
File Edit View Search Terminal Help

```
[cloudera@localhost Desktop]$ pig /home/cloudera/Desktop/t1.txt
```

t1.txt (~/Desktop) - gedit
File Edit View Search Tools Documents Help
Open
Save
Undo
t1.txt

```
A = load '/user/cloudera/txns-large.dat' using PigStorage (',') as
(tid, d, uid, amt : double , cat, prod,city,state,pt);
B = foreach  A generate tid, amt;
C = filter B by ($1>170 and $1<200);
D = foreach C generate 1 as one;
E = group D by one;
F = foreach E generate COUNT(D.one);
dump F;
```



```
2016-11-23 22:06:42,123 [main] INFO  org.apache.pig.backend.hadoop.executionengi
ne.util.MapRedUtil - Total input paths to process : 1
(7779)
[cloudera@localhost Desktop]$ 
```

Use Case 3: ***A User Aggregate***

- **Input File**: txns-large.dat
- **Input**: ID [Custom Input from the User]
- **Key**: ID, **Value**: Amount
- **Output**: Count, Sum & Average Transaction of a User.
- **Data Validation**: Yes.

  Constraint: User Input Can be Only Numbers
- **Concept** : Advanced Map Reduce, Hive & PIG

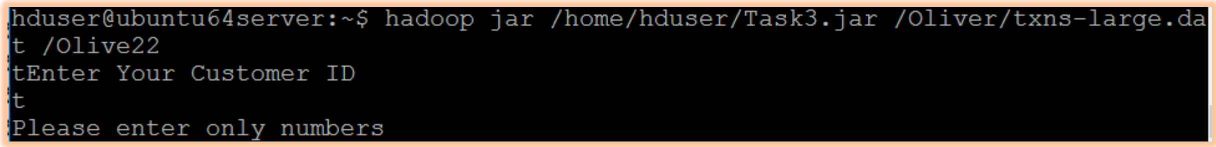
Description: To obtain a summarized data of a user about the transaction completed.

Why this Report: To get an Individual Customer Performance

Progress: This report can be used any Employee in the Organization to identify their helpdesk queries. However this can be also used to identify the Summarized Customer Performance till date.

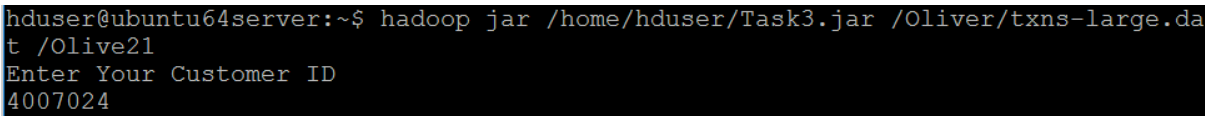
3.1 Advanced Map Reduce

Screen Shot 3.1.1: Data Validation



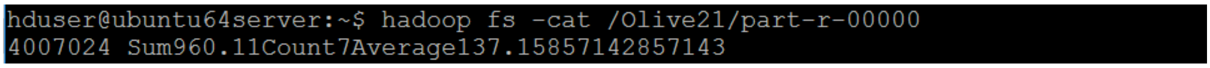
```
hduser@ubuntu64server:~$ hadoop jar /home/hduser/Task3.jar /Oliver/txns-large.da
t /Olive22
tEnter Your Customer ID
t
Please enter only numbers
```

Screen Shot 3.1.2: Input Window



```
hduser@ubuntu64server:~$ hadoop jar /home/hduser/Task3.jar /Oliver/txns-large.da
t /Olive21
Enter Your Customer ID
4007024
```

Screen Shot 3.1.3: Output Window



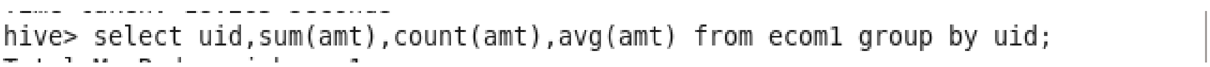
```
hduser@ubuntu64server:~$ hadoop fs -cat /Olive21/part-r-00000
4007024 Sum960.11Count7Average137.15857142857143
```

Output Path:

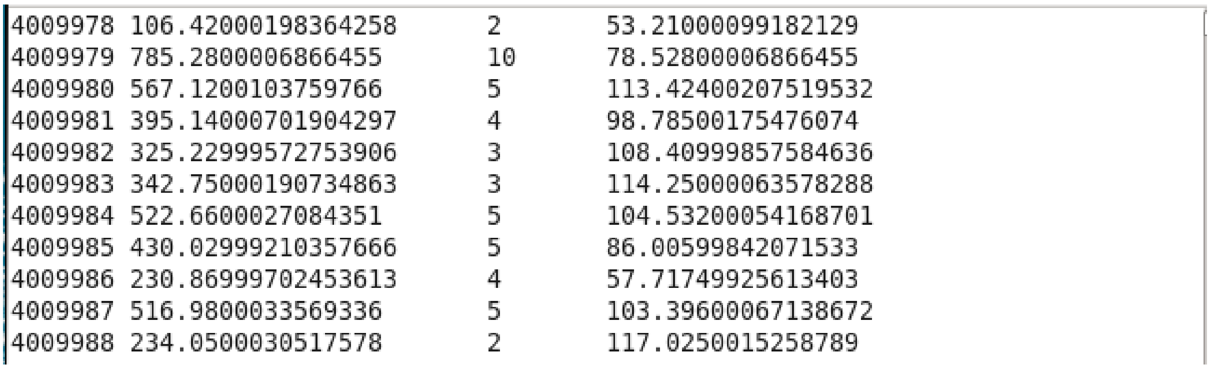
Hadoop fs –cat /Olive21/part-r-00000

3.2 HIVE:

Screen Shot 3.2.1:



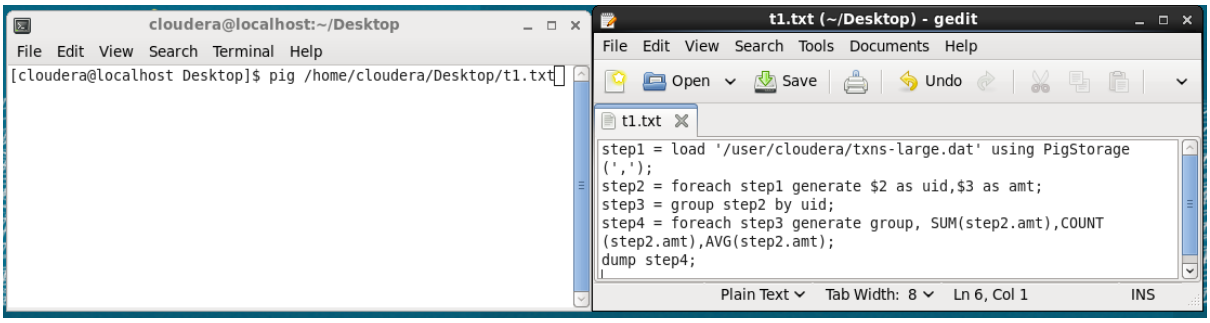
```
hive> select uid,sum(amt),count(amt),avg(amt) from ecom1 group by uid;
```



```
4009978 106.42000198364258      2       53.21000099182129
4009979 785.2800006866455       10      78.52800006866455
4009980 567.1200103759766       5       113.42400207519532
4009981 395.14000701904297      4       98.78500175476074
4009982 325.22999572753906      3       108.40999857584636
4009983 342.75000190734863      3       114.25000063578288
4009984 522.6600027084351       5       104.53200054168701
4009985 430.02999210357666      5       86.00599842071533
4009986 230.86999702453613      4       57.71749925613403
4009987 516.9800033569336       5       103.39600067138672
4009988 234.0500030517578       2       117.0250015258789
```

3.3 PIG

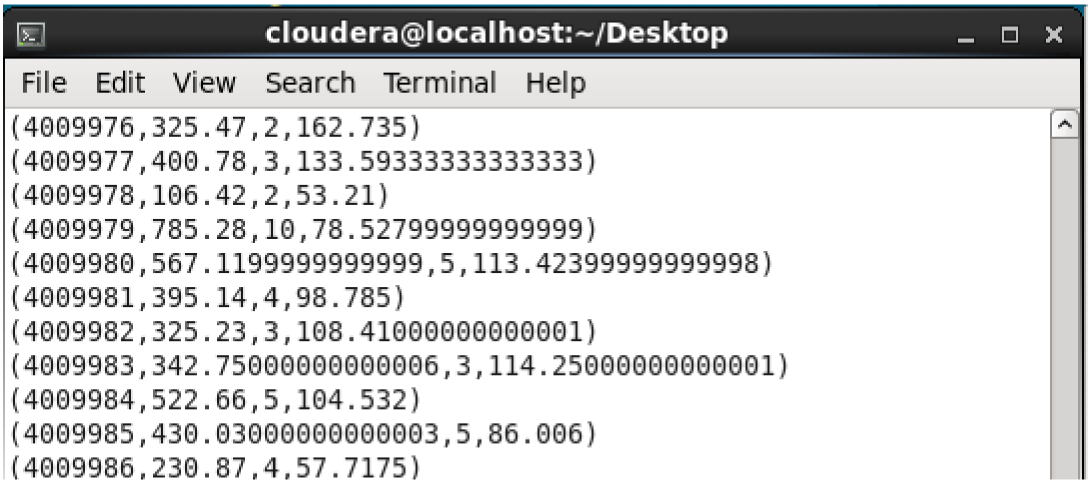
Screen Shot 3.3.1:



cloudera@localhost:~/Desktop
File Edit View Search Terminal Help
t1.txt (~/Desktop) - gedit
File Edit View Search Tools Documents Help
Open
Save
Undo
t1.txt
Plain Text
Tab Width: 8
Ln 6, Col 1
INS

```
[cloudera@localhost Desktop]$ pig /home/cloudera/Desktop/t1.txt
```

```
step1 = load '/user/cloudera/txns-large.dat' using PigStorage
(',');
step2 = foreach step1 generate $2 as uid,$3 as amt;
step3 = group step2 by uid;
step4 = foreach step3 generate group, SUM(step2.amt),COUNT
(step2.amt),AVG(step2.amt);
dump step4;
```



cloudera@localhost:~/Desktop
File Edit View Search Terminal Help

```
(4009976,325.47,2,162.735)
(4009977,400.78,3,133.59333333333333)
(4009978,106.42,2,53.21)
(4009979,785.28,10,78.52799999999999)
(4009980,567.1199999999999,5,113.42399999999998)
(4009981,395.14,4,98.785)
(4009982,325.23,3,108.41000000000001)
(4009983,342.75000000000006,3,114.25000000000001)
(4009984,522.66,5,104.532)
(4009985,430.03000000000003,5,86.006)
(4009986,230.87,4,57.7175)
```

Use Case 4: ***Quick Month Sales Review***

- **Input File**: txns-large.dat
- **Input**: Month [Custom Input from the User]
- **Key**: Month, **Value**: Amount
- **Output** : Total Sales of Month
- **Data Validation**: Yes.

Constraints 1: Month can be only between 1-12
Constraints 2: Month can be only Positive Value
Constraints 3: Month can be only in Numbers

- **Concept**: Advanced Map Reduce & PIG

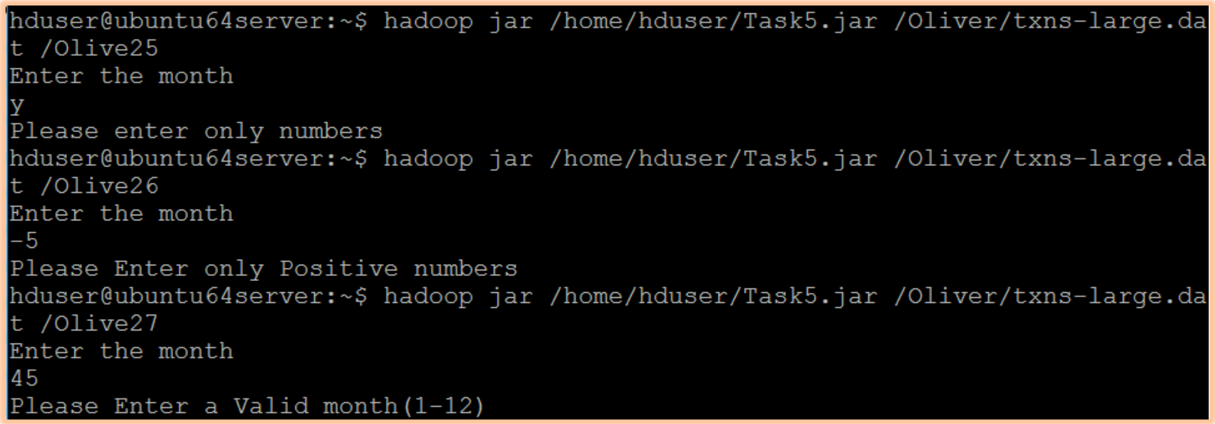
Description: To obtain a summarized view of a Month.

Why this Report: For an effective Analysis

Progress: Every month it is the responsibility for a Business head to analyse the sales performance. The data obtained from the base is very large and hence the filtering of an individual data as per the requirement is complex. This report will give an overall sales made in the month handy to the Business head to understand where do they stand and bring out innovative ideas to move further.

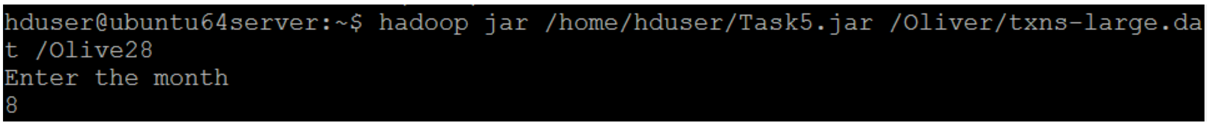
4.1. Advanced Map Reduce

Screen Shot 4.1.1. Data Validation



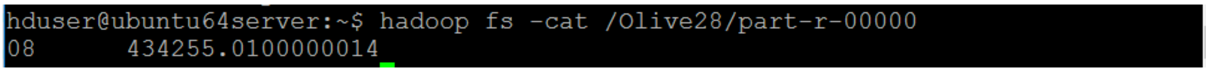
```
hduser@ubuntu64server:~$ hadoop jar /home/hduser/Task5.jar /Oliver/txns-large.da
t /Olive25
Enter the month
y
Please enter only numbers
hduser@ubuntu64server:~$ hadoop jar /home/hduser/Task5.jar /Oliver/txns-large.da
t /Olive26
Enter the month
-5
Please Enter only Positive numbers
hduser@ubuntu64server:~$ hadoop jar /home/hduser/Task5.jar /Oliver/txns-large.da
t /Olive27
Enter the month
45
Please Enter a Valid month(1-12)
```

Screen Shot 4.1.2: Input Window



```
hduser@ubuntu64server:~$ hadoop jar /home/hduser/Task5.jar /Oliver/txns-large.da
t /Olive28
Enter the month
8
```

Screen Shot 4.1.3: Output Window

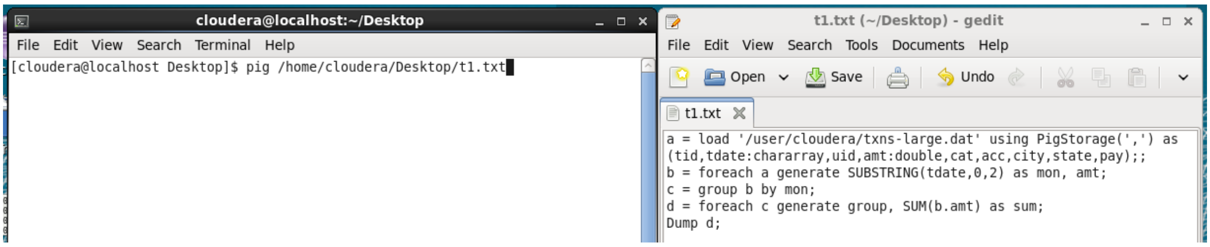


```
hduser@ubuntu64server:~$ hadoop fs -cat /Olive28/part-r-00000
08      434255.0100000014
```

Output Path:

Hadoop fs –cat /Olive28/part-r-00000

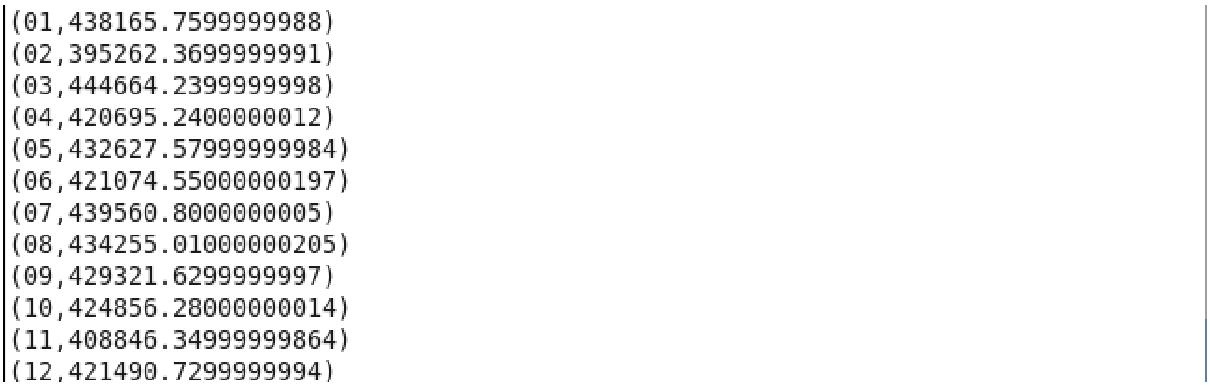
4.2 PIG:



cloudera@localhost:~/Desktop
File Edit View Search Terminal Help
t1.txt (~/Desktop) - gedit
File Edit View Search Tools Documents Help
Open
Save
Undo
t1.txt

```
[cloudera@localhost Desktop]$ pig /home/cloudera/Desktop/t1.txt
```

```
a = load '/user/cloudera/txns-large.dat' using PigStorage(',') as
(tid,tdate:chararray,uid,amt:double,cat,acc,city,state,pay);;
b = foreach a generate SUBSTRING(tdate,0,2) as mon, amt;
c = group b by mon;
d = foreach c generate group, SUM(b.amt) as sum;
Dump d;
```



```
(01,438165.7599999988)
(02,395262.3699999991)
(03,444664.2399999998)
(04,420695.2400000012)
(05,432627.57999999984)
(06,421074.55000000197)
(07,439560.8000000005)
(08,434255.01000000205)
(09,429321.6299999997)
(10,424856.28000000014)
(11,408846.34999999864)
(12,421490.7299999994)
```

Use Case 5: *Large Module – Small Module Visualization*

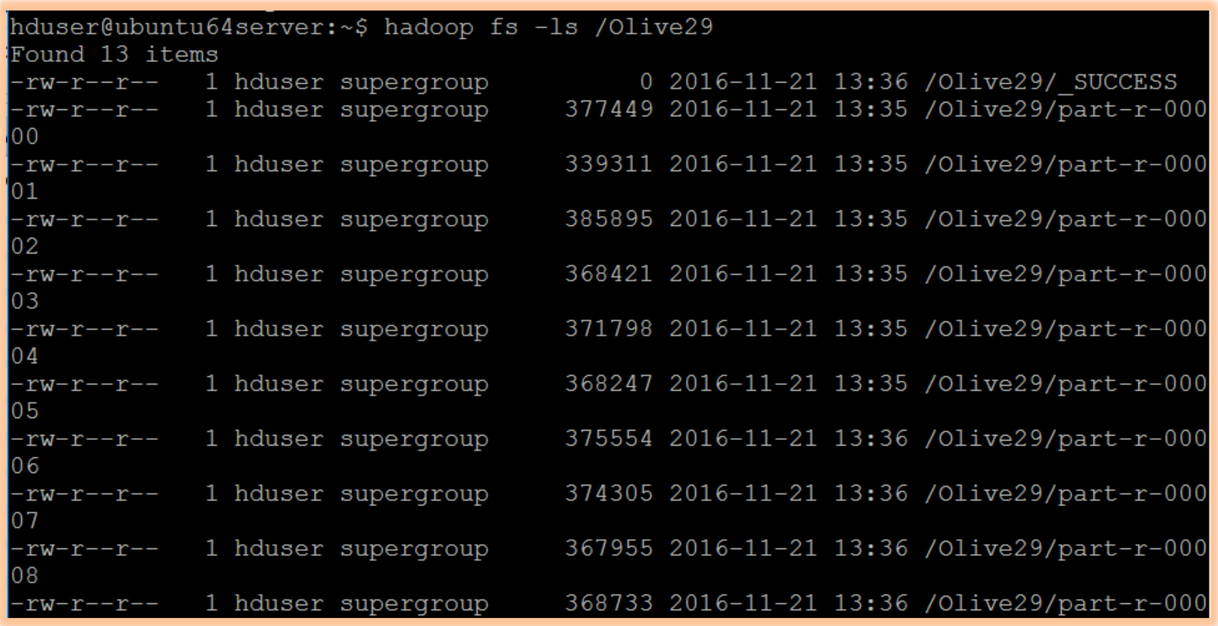
- **Input File**: txns-large.dat
- **Input** : Predefined
- **Key**: Month, **Value**: Entire Line of Data
- **Output** : Partitioned Month
- **Data Validation**: NA
- **Concept** : Map Reduce with Partitioner & PIG

Description: To Obtain a Distributed output for every month specific Data. Each of the output will be stored in a separate file based on month.

Why this Report: For modularity

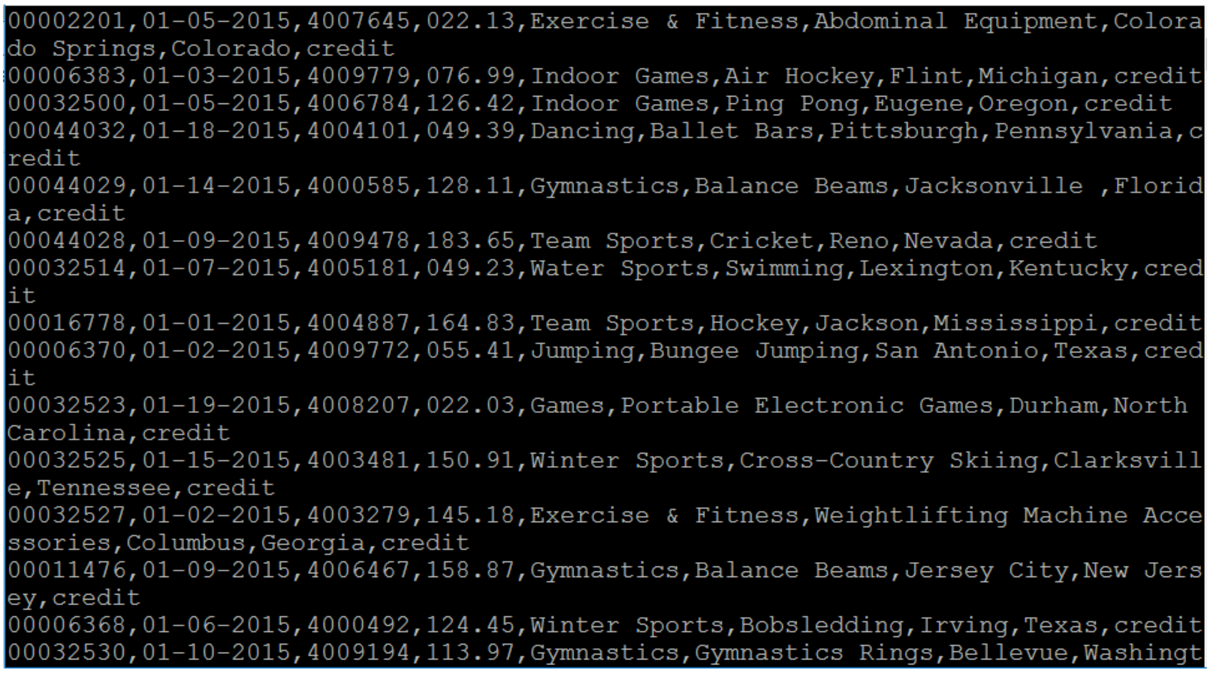
Progress: The server used by the organization streams various data from the Clients. The frequency of the data will be unimaginable. All these data to the server is dumped together. Hadoop developer of the company can help the admin to partition the data based on the month. Now this can be a dual purpose way. Admin can maintain back up of data for every month as well an over view for the managers.

Screen Shot 5.1: ***Output Window [Multiple Partitioned Files]***



```
hduser@ubuntu64server:~$ hadoop fs -ls /Olive29
Found 13 items
-rw-r--r--   1 hduser supergroup          0 2016-11-21 13:36 /Olive29/_SUCCESS
-rw-r--r--   1 hduser supergroup     377449 2016-11-21 13:35 /Olive29/part-r-000
00
-rw-r--r--   1 hduser supergroup     339311 2016-11-21 13:35 /Olive29/part-r-000
01
-rw-r--r--   1 hduser supergroup     385895 2016-11-21 13:35 /Olive29/part-r-000
02
-rw-r--r--   1 hduser supergroup     368421 2016-11-21 13:35 /Olive29/part-r-000
03
-rw-r--r--   1 hduser supergroup     371798 2016-11-21 13:35 /Olive29/part-r-000
04
-rw-r--r--   1 hduser supergroup     368247 2016-11-21 13:35 /Olive29/part-r-000
05
-rw-r--r--   1 hduser supergroup     375554 2016-11-21 13:36 /Olive29/part-r-000
06
-rw-r--r--   1 hduser supergroup     374305 2016-11-21 13:36 /Olive29/part-r-000
07
-rw-r--r--   1 hduser supergroup     367955 2016-11-21 13:36 /Olive29/part-r-000
08
-rw-r--r--   1 hduser supergroup     368733 2016-11-21 13:36 /Olive29/part-r-000
```

Screen Shot 5.2: ***Output Window [Specific Month View]***



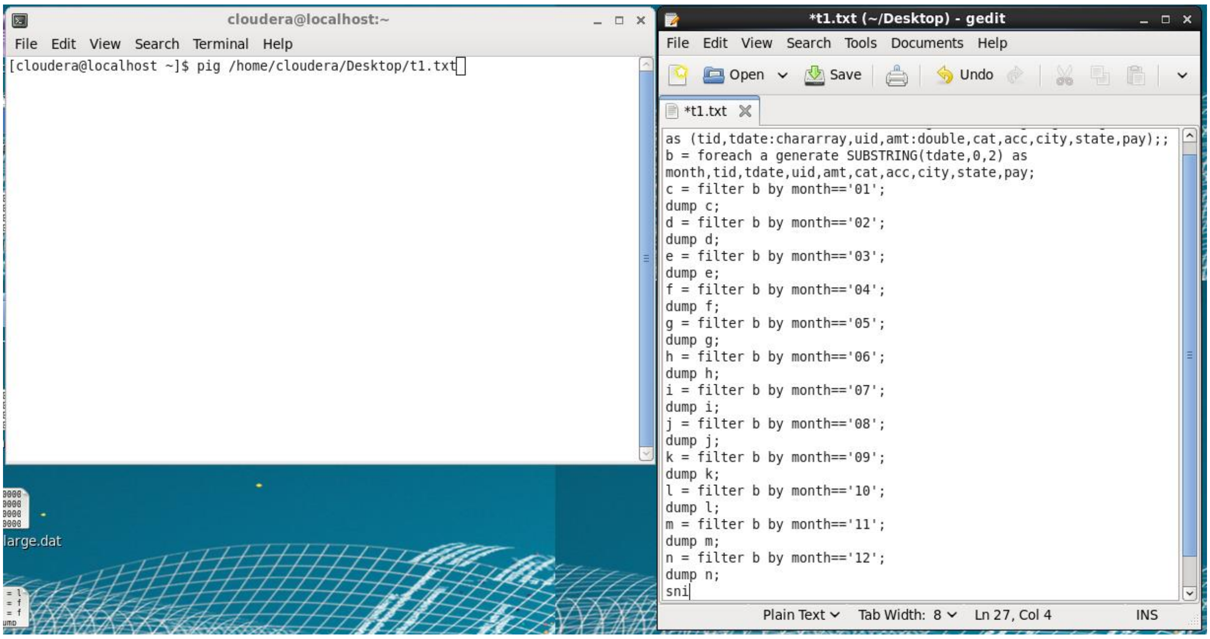
```
00002201,01-05-2015,4007645,022.13,Exercise & Fitness,Abdominal Equipment,Colora
do Springs,Colorado,credit
00006383,01-03-2015,4009779,076.99,Indoor Games,Air Hockey,Flint,Michigan,credit
00032500,01-05-2015,4006784,126.42,Indoor Games,Ping Pong,Eugene,Oregon,credit
00044032,01-18-2015,4004101,049.39,Dancing,Ballet Bars,Pittsburgh,Pennsylvania,c
redit
00044029,01-14-2015,4000585,128.11,Gymnastics,Balance Beams,Jacksonville ,Florid
a,credit
00044028,01-09-2015,4009478,183.65,Team Sports,Cricket,Reno,Nevada,credit
00032514,01-07-2015,4005181,049.23,Water Sports,Swimming,Lexington,Kentucky,cred
it
00016778,01-01-2015,4004887,164.83,Team Sports,Hockey,Jackson,Mississippi,credit
00006370,01-02-2015,4009772,055.41,Jumping,Bungee Jumping,San Antonio,Texas,cred
it
00032523,01-19-2015,4008207,022.03,Games,Portable Electronic Games,Durham,North
Carolina,credit
00032525,01-15-2015,4003481,150.91,Winter Sports,Cross-Country Skiing,Clarksvill
e,Tennessee,credit
00032527,01-02-2015,4003279,145.18,Exercise & Fitness,Weightlifting Machine Acce
ssories,Columbus,Georgia,credit
00011476,01-09-2015,4006467,158.87,Gymnastics,Balance Beams,Jersey City,New Jers
ey,credit
00006368,01-06-2015,4000492,124.45,Winter Sports,Bobsledding,Irving,Texas,credit
00032530,01-10-2015,4009194,113.97,Gymnastics,Gymnastics Rings,Bellevue,Washingt
```

Output Path:

Hadoop fs –cat /Olive29/part-r-00000

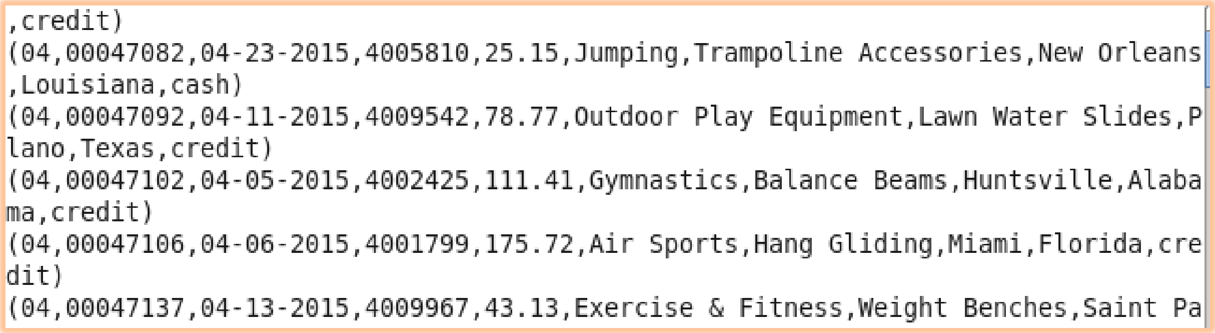
5.2 PIG

Screen Shot 5.2.1:

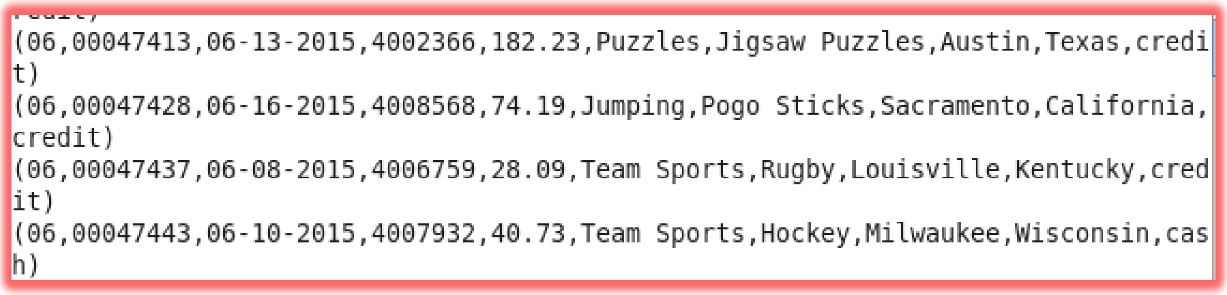

cloudera@localhost:~
File Edit View Search Terminal Help
*t1.txt (~/Desktop) - gedit
File Edit View Search Tools Documents Help
Open
Save
Undo
*t1.txt
large.dat
Plain Text
Tab Width: 8
Ln 27, Col 4
INS

```
[cloudera@localhost ~]$ pig /home/cloudera/Desktop/t1.txt
```

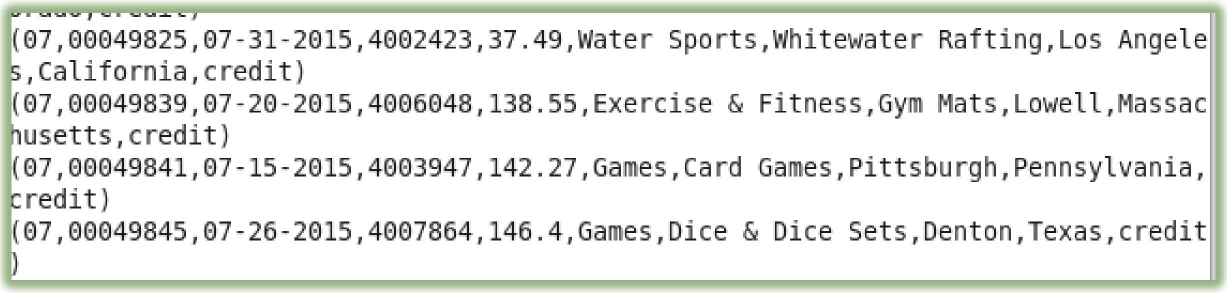
```
as (tid,tdate:chararray,uid,amt:double,cat,acc,city,state,pay);;
b = foreach a generate SUBSTRING(tdate,0,2) as
month,tid,tdate,uid,amt,cat,acc,city,state,pay;
c = filter b by month=='01';
dump c;
d = filter b by month=='02';
dump d;
e = filter b by month=='03';
dump e;
f = filter b by month=='04';
dump f;
g = filter b by month=='05';
dump g;
h = filter b by month=='06';
dump h;
i = filter b by month=='07';
dump i;
j = filter b by month=='08';
dump j;
k = filter b by month=='09';
dump k;
l = filter b by month=='10';
dump l;
m = filter b by month=='11';
dump m;
n = filter b by month=='12';
dump n;
sni
```



```
,credit)
(04,00047082,04-23-2015,4005810,25.15,Jumping,Trampoline Accessories,New Orleans
,Louisiana,cash)
(04,00047092,04-11-2015,4009542,78.77,Outdoor Play Equipment,Lawn Water Slides,P
lano,Texas,credit)
(04,00047102,04-05-2015,4002425,111.41,Gymnastics,Balance Beams,Huntsville,Alaba
ma,credit)
(04,00047106,04-06-2015,4001799,175.72,Air Sports,Hang Gliding,Miami,Florida,cre
dit)
(04,00047137,04-13-2015,4009967,43.13,Exercise & Fitness,Weight Benches,Saint Pa
```



```
(06,00047413,06-13-2015,4002366,182.23,Puzzles,Jigsaw Puzzles,Austin,Texas,credi
t)
(06,00047428,06-16-2015,4008568,74.19,Jumping,Pogo Sticks,Sacramento,California,
credit)
(06,00047437,06-08-2015,4006759,28.09,Team Sports,Rugby,Louisville,Kentucky,cred
it)
(06,00047443,06-10-2015,4007932,40.73,Team Sports,Hockey,Milwaukee,Wisconsin,cas
h)
```



```
(07,00049825,07-31-2015,4002423,37.49,Water Sports,Whitewater Rafting,Los Angele
s,California,credit)
(07,00049839,07-20-2015,4006048,138.55,Exercise & Fitness,Gym Mats,Lowell,Massac
husetts,credit)
(07,00049841,07-15-2015,4003947,142.27,Games,Card Games,Pittsburgh,Pennsylvania,
credit)
(07,00049845,07-26-2015,4007864,146.4,Games,Dice & Dice Sets,Denton,Texas,credit
)
```

Use Case 6: Take me to and From the Beginning

- **Input File**: txns-large.dat
- **Input** : AMT
- **Key**: AMT, **Value**: Entire line of a data
- **Output** : SORTED data based by Amount
- **Concept** : Simple Map Reduce & PIG

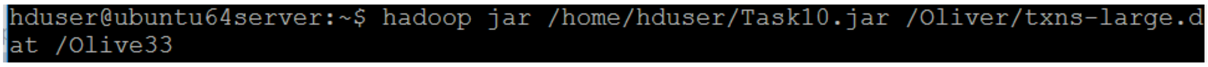
Description: To sort the output based on the Amount

Why this Report: Range of Results.

Progress: This report will be handy to managers of various departments to identify the different set of products sold in various ranges. This can be partitioned to multiple employees to concentrate on the product which has been sold quickly / more, identify the scarcity of the product and meet the demands of the customer

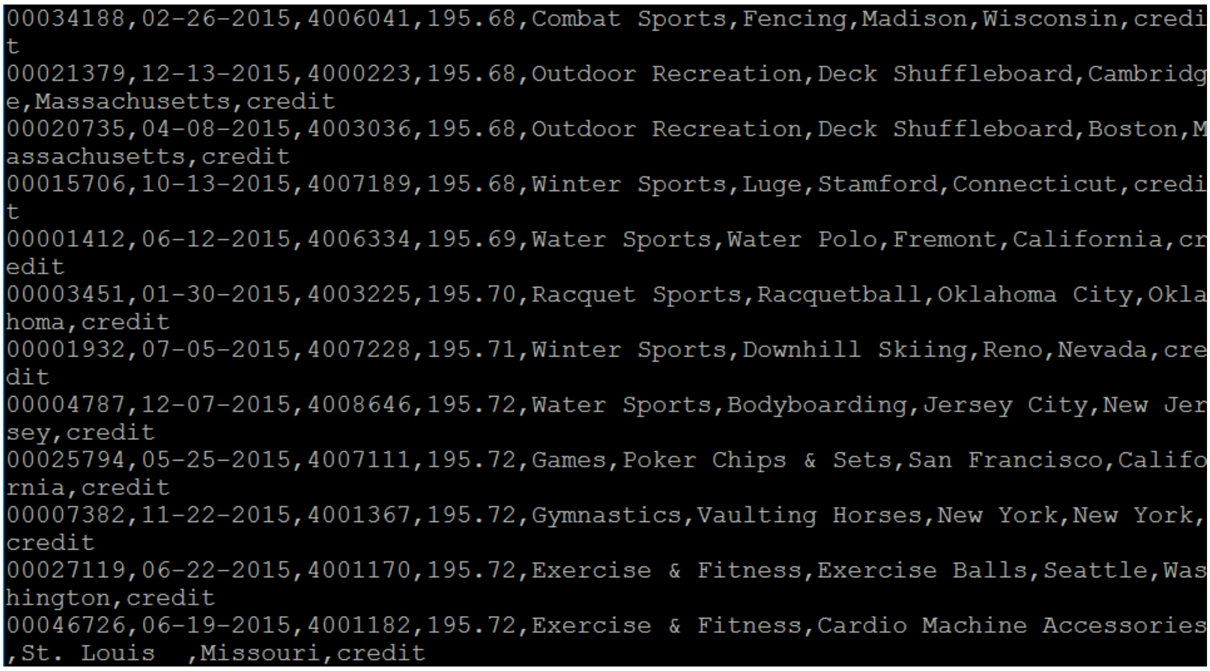
6.1 Simple Map Reduce

Screen Shot 6.1.1: Input Window



```
hduser@ubuntu64server:~$ hadoop jar /home/hduser/Task10.jar /Oliver/txns-large.d
at /Olive33
```

Screen Shot 6.1.2 : Output Window

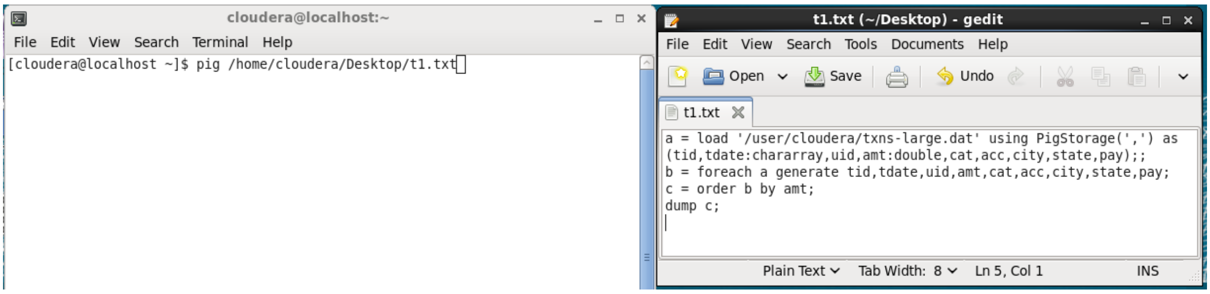


```
00034188,02-26-2015,4006041,195.68,Combat Sports,Fencing,Madison,Wisconsin,credi
t
00021379,12-13-2015,4000223,195.68,Outdoor Recreation,Deck Shuffleboard,Cambridg
e,Massachusetts,credit
00020735,04-08-2015,4003036,195.68,Outdoor Recreation,Deck Shuffleboard,Boston,M
assachusetts,credit
00015706,10-13-2015,4007189,195.68,Winter Sports,Luge,Stamford,Connecticut,credi
t
00001412,06-12-2015,4006334,195.69,Water Sports,Water Polo,Fremont,California,cr
edit
00003451,01-30-2015,4003225,195.70,Racquet Sports,Racquetball,Oklahoma City,Okla
homa,credit
00001932,07-05-2015,4007228,195.71,Winter Sports,Downhill Skiing,Reno,Nevada,cre
dit
00004787,12-07-2015,4008646,195.72,Water Sports,Bodyboarding,Jersey City,New Jer
sey,credit
00025794,05-25-2015,4007111,195.72,Games,Poker Chips & Sets,San Francisco,Califo
rnia,credit
00007382,11-22-2015,4001367,195.72,Gymnastics,Vaulting Horses,New York,New York,
credit
00027119,06-22-2015,4001170,195.72,Exercise & Fitness,Exercise Balls,Seattle,Was
hington,credit
00046726,06-19-2015,4001182,195.72,Exercise & Fitness,Cardio Machine Accessories
,St. Louis  ,Missouri,credit
```

Output Path:

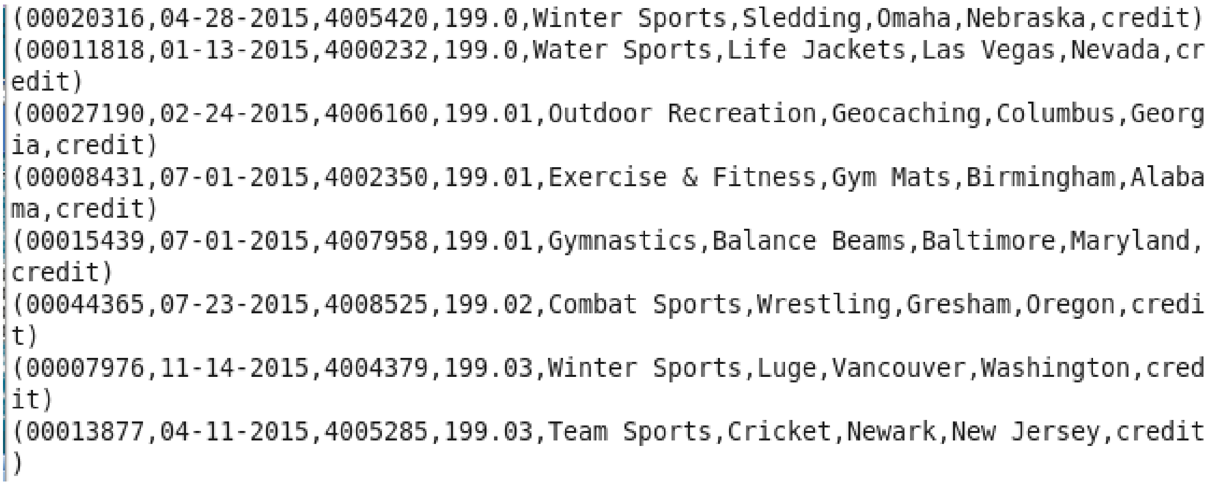
Hadoop fs –cat /Olive33/part-r-0000

6.2 PIG


cloudera@localhost:~
File Edit View Search Terminal Help
t1.txt (~/Desktop) - gedit
File Edit View Search Tools Documents Help
Open
Save
Undo
t1.txt
Plain Text
Tab Width: 8
Ln 5, Col 1
INS

```
[cloudera@localhost ~]$ pig /home/cloudera/Desktop/t1.txt
```

```
a = load '/user/cloudera/txns-large.dat' using PigStorage(',') as
(tid,tdate:chararray,uid,amt:double,cat,acc,city,state,pay);;
b = foreach a generate tid,tdate,uid,amt,cat,acc,city,state,pay;
c = order b by amt;
dump c;
```



```
(00020316,04-28-2015,4005420,199.0,Winter Sports,Sledding,Omaha,Nebraska,credit)
(00011818,01-13-2015,4000232,199.0,Water Sports,Life Jackets,Las Vegas,Nevada,cr
edit)
(00027190,02-24-2015,4006160,199.01,Outdoor Recreation,Geocaching,Columbus,Georg
ia,credit)
(00008431,07-01-2015,4002350,199.01,Exercise & Fitness,Gym Mats,Birmingham,Alaba
ma,credit)
(00015439,07-01-2015,4007958,199.01,Gymnastics,Balance Beams,Baltimore,Maryland,
credit)
(00044365,07-23-2015,4008525,199.02,Combat Sports,Wrestling,Gresham,Oregon,credi
t)
(00007976,11-14-2015,4004379,199.03,Winter Sports,Luge,Vancouver,Washington,cred
it)
(00013877,04-11-2015,4005285,199.03,Team Sports,Cricket,Newark,New Jersey,credit
)
```

Use Case 7: *Top 3 Contributors*

- **Input File**: txns-large.dat, Customer.dat
- **Input** : ID, AMT, NAME
- **Key**: NAME, **Value**: AMT
- **Output** : NAME
- **Concept** : Mapper Side Join & PIG

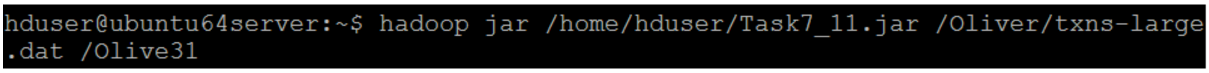
Description: To find the top 3 Customers who has spent the MAX Transaction

Why this Report: To Identify Category wise Performers.

Progress: This report is just to identify top customers who has done good amount of transactions at various products. This will help to identify the fast moving products and will help to promote advertisement about the product as soon the customer opens the website.

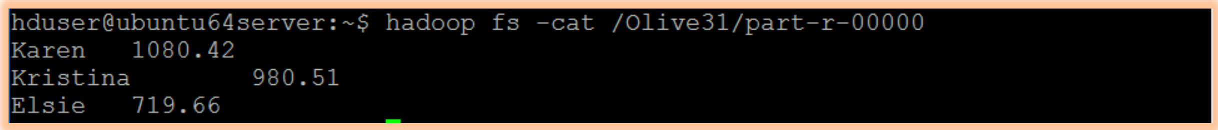
7.1 Simple Map Reduce – Mapper Side Join

Screen Shot 7. 1: Input Window



```
hduser@ubuntu64server:~$ hadoop jar /home/hduser/Task7_11.jar /Oliver/txns-large
.dat /Olive31
```

Screen Shot 7. 2: Output Window

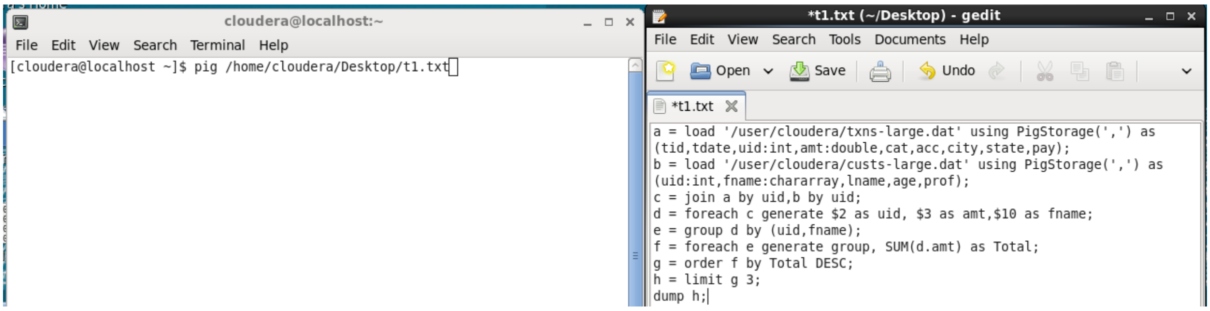


```
hduser@ubuntu64server:~$ hadoop fs -cat /Olive31/part-r-00000
Karen    1080.42
Kristina        980.51
Elsie    719.66
```

Output Path:

Hadoop fs –cat /Olive31/part-r-00000

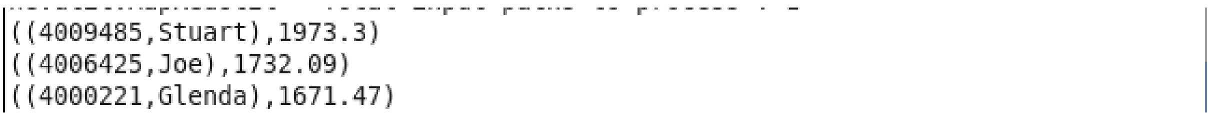
7.2 PIG



cloudera@localhost:~
File Edit View Search Terminal Help
*t1.txt (~/Desktop) - gedit
File Edit View Search Tools Documents Help
Open
Save
Undo
*t1.txt

```
[cloudera@localhost ~]$ pig /home/cloudera/Desktop/t1.txt
```

```
a = load '/user/cloudera/txns-large.dat' using PigStorage(',') as
(tid,tdate,uid:int,amt:double,cat,acc,city,state,pay);
b = load '/user/cloudera/custs-large.dat' using PigStorage(',') as
(uid:int,fname:chararray,lname,age,prof);
c = join a by uid,b by uid;
d = foreach c generate $2 as uid, $3 as amt,$10 as fname;
e = group d by (uid,fname);
f = foreach e generate group, SUM(d.amt) as Total;
g = order f by Total DESC;
h = limit g 3;
dump h;
```



```
((4009485,Stuart),1973.3)
((4006425,Joe),1732.09)
((4000221,Glenda),1671.47)
```

Use Case 8: ***Rock Star***

- **Input File**: txns-large.dat, Customer.dat
- **Input** : ID, AMT, NAME
- **Key**: NAME, **Value**: AMT
- **Output** : NAME [BASED ON MONTH]
- **Concept** : Mapper Side Join & PIG

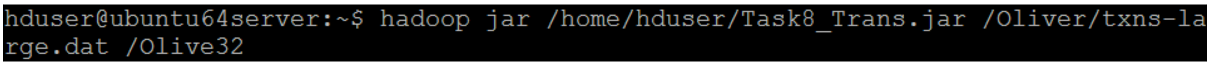
Description: To find the User who has done more contribution

Why this Report: To identify Luckiest Person.

Progress: To gift the customer with an exciting prize for the contribution made and the prize details to be published on the login site of the application, to attract customers.

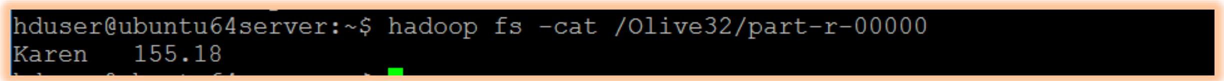
8.1 Simple Map Reduce - Mapper Side Join

Screen Shot 8. 1.1: Input Window



```
hduser@ubuntu64server:~$ hadoop jar /home/hduser/Task8_Trans.jar /Oliver/txns-la
rge.dat /Olive32
```

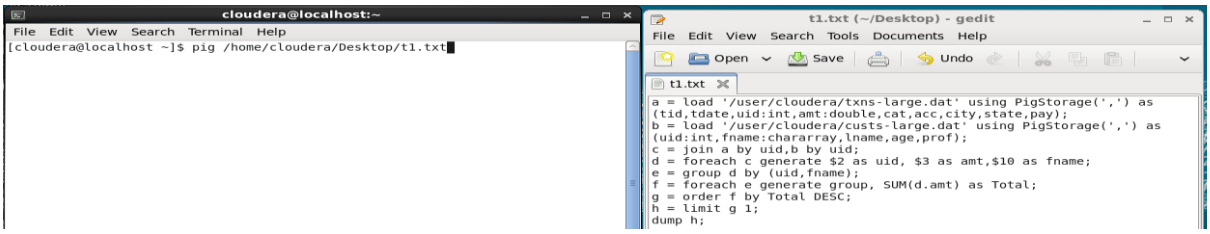
Screen Shot 8.1.2 : Output Window



```
hduser@ubuntu64server:~$ hadoop fs -cat /Olive32/part-r-00000
Karen   155.18
```

8.2 PIG:

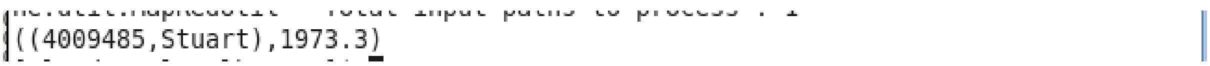
Screen 8.2.1 :



cloudera@localhost:~
File Edit View Search Terminal Help
t1.txt (~/Desktop) - gedit
File Edit View Search Tools Documents Help
Open
Save
Undo
t1.txt

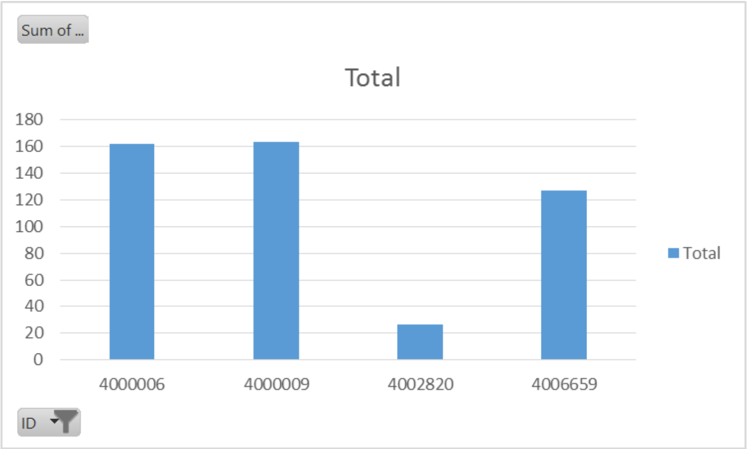
```
[cloudera@localhost ~]$ pig /home/cloudera/Desktop/t1.txt
```

```
a = load '/user/cloudera/txns-large.dat' using PigStorage(',') as
(tid,tdate,uid:int,amt:double,cat,acc,city,state,pay);
b = load '/user/cloudera/custs-large.dat' using PigStorage(',') as
(uid:int,fname:chararray,lname,age,prof);
c = join a by uid,b by uid;
d = foreach c generate $2 as uid, $3 as amt,$10 as fname;
e = group d by (uid,fname);
f = foreach e generate group, SUM(d.amt) as Total;
g = order f by Total DESC;
h = limit g 1;
dump h;
```



```
ne.util.MapRedUtil - Total input paths to process : 1
((4009485,Stuart),1973.3)
```

Analysis:



Use Case 9: Month Specific Customer

- **Input File**: txns-large.dat, Customer.dat
- **Input** : ID, AMT, NAME
- **Key**: NAME, **Value**: AMT
- **Output** : NAME [BASED ON MONTH]
- **Concept** : Mapper Side Join & PIG

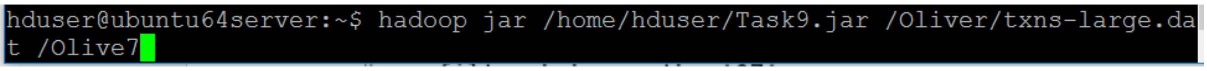
Description: To find the User who has Spent MAX Amount in the month of July / Vary according to the preferences.

Why this Report: To identify Customer Month Wise Specific.

Progress: This report is not only to identify the customer who has done more contribution in the month of July, it is also to understand what kind of products that customer has purchased more in various categories so that the organization can promote the products in a better way.

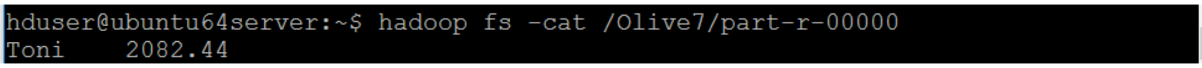
9.1: Simple Map Reduce – Mapper Side Join.

Screen Shot 9.1: Input Window



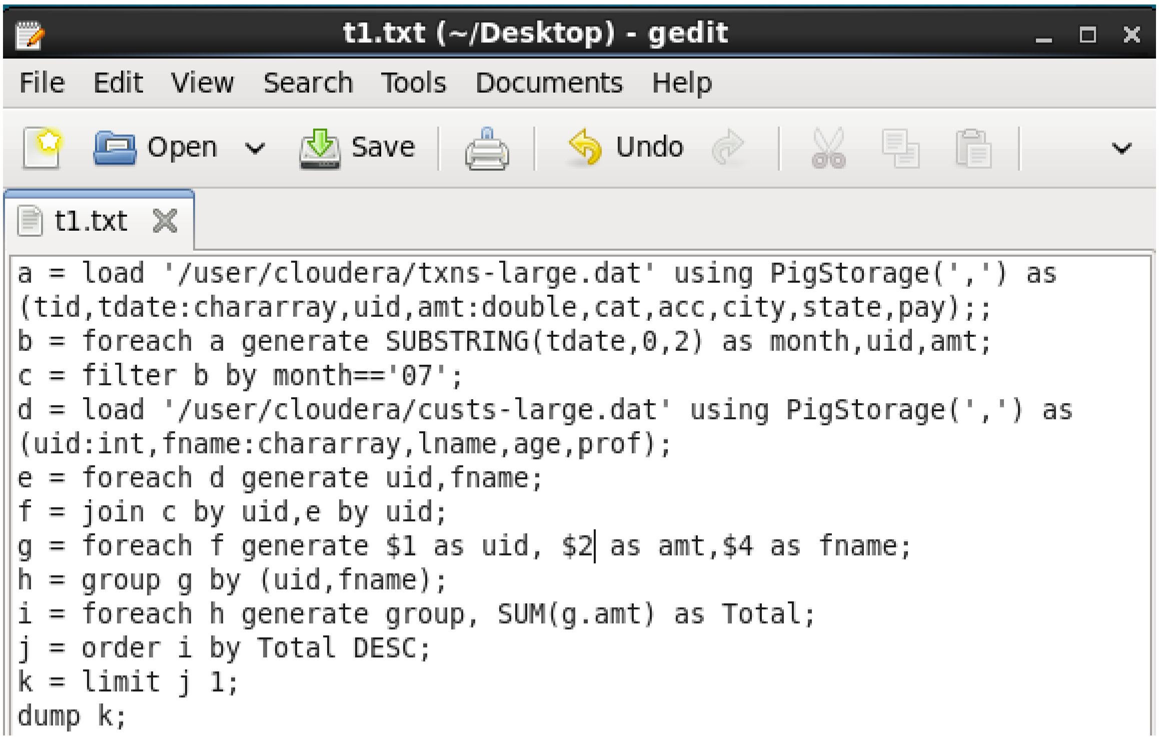
```
hduser@ubuntu64server:~$ hadoop jar /home/hduser/Task9.jar /Oliver/txns-large.da
t /Olive7
```

Screen Shot 9.2: Output Window

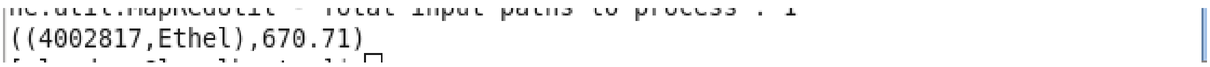


```
hduser@ubuntu64server:~$ hadoop fs -cat /Olive7/part-r-00000
Toni     2082.44
```

9.2: PIG:



```
a = load '/user/cloudera/txns-large.dat' using PigStorage(',') as
(tid,tdate:chararray,uid,amt:double,cat,acc,city,state,pay);;
b = foreach a generate SUBSTRING(tdate,0,2) as month,uid,amt;
c = filter b by month=='07';
d = load '/user/cloudera/custs-large.dat' using PigStorage(',') as
(uid:int,fname:chararray,lname,age,prof);
e = foreach d generate uid,fname;
f = join c by uid,e by uid;
g = foreach f generate $1 as uid, $2 as amt,$4 as fname;
h = group g by (uid,fname);
i = foreach h generate group, SUM(g.amt) as Total;
j = order i by Total DESC;
k = limit j 1;
dump k;
```



```
ne.util.MapRedUtil - Total input paths to process : 1
((4002817,Ethel),670.71)
```

Output Path:

Hadoop fs –cat /Olive32/part-r-00000

Conclusion:

A handy tool that makes the life easier of employee in the organization is now ready. This Project, as title says, as 360 Degree report generation, that can be used by the Organization to make their day – to – day activities easier and overcomes the time spent to analyse the data manually.