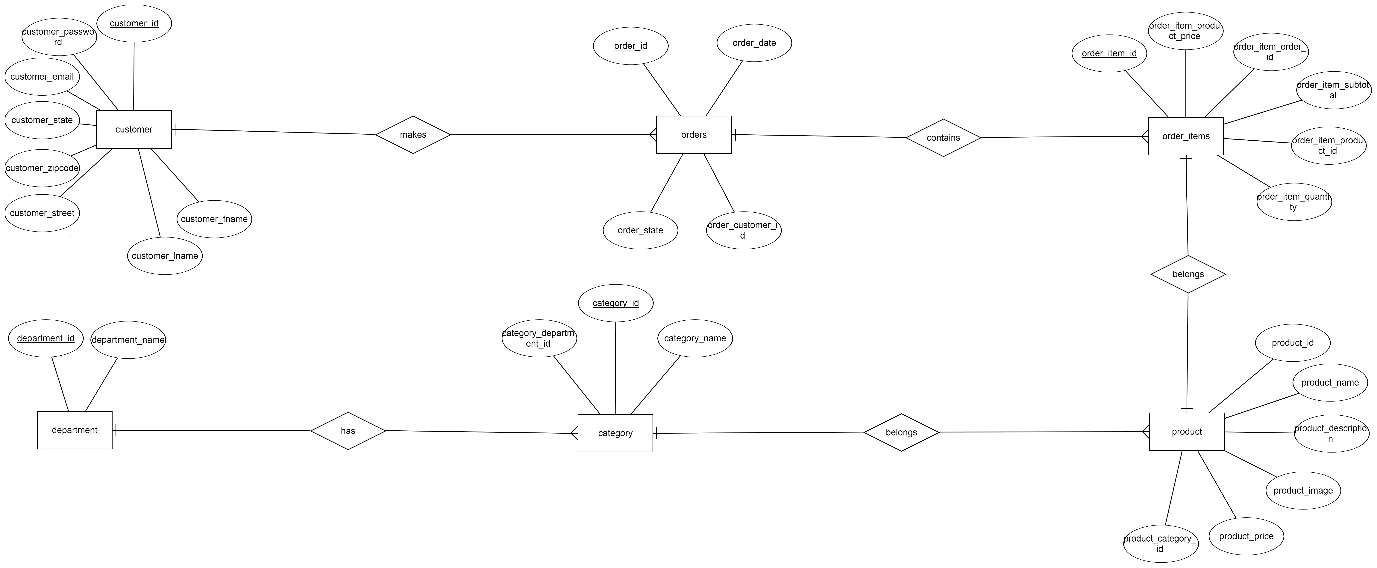
**Day-1**

>mysql -u root –p //u= username p=password.

>show databases; //lists the databases.

>use surya\_project; // user defined database.

**ER DIAGRAM**:



**INDENTIFYING RELATIONSHIP**

**CUSTOMER table:**

Primary key : customer\_id

Foreign key : none

References Table :none

//main table

**ORDERS table:**

Primary key : order\_id

Foreign key : order\_customer-id

References Table : CUSTOMER

**ORDER\_ITEM table:**

Primary key : order\_item\_id

Foreign key :order\_item\_order\_id,order\_item\_product\_id

References Table : ORDERS,PRODUCT

**PRODUCT table:**

Primary key : product\_id

Foreign key : product\_category\_id

References Table : CATEGORIES

**CATEGORIES:**

Primary key : Category\_name

Foreign key : category\_department\_id

References Table : DEPARTMENT

**DEPARTMENT:**

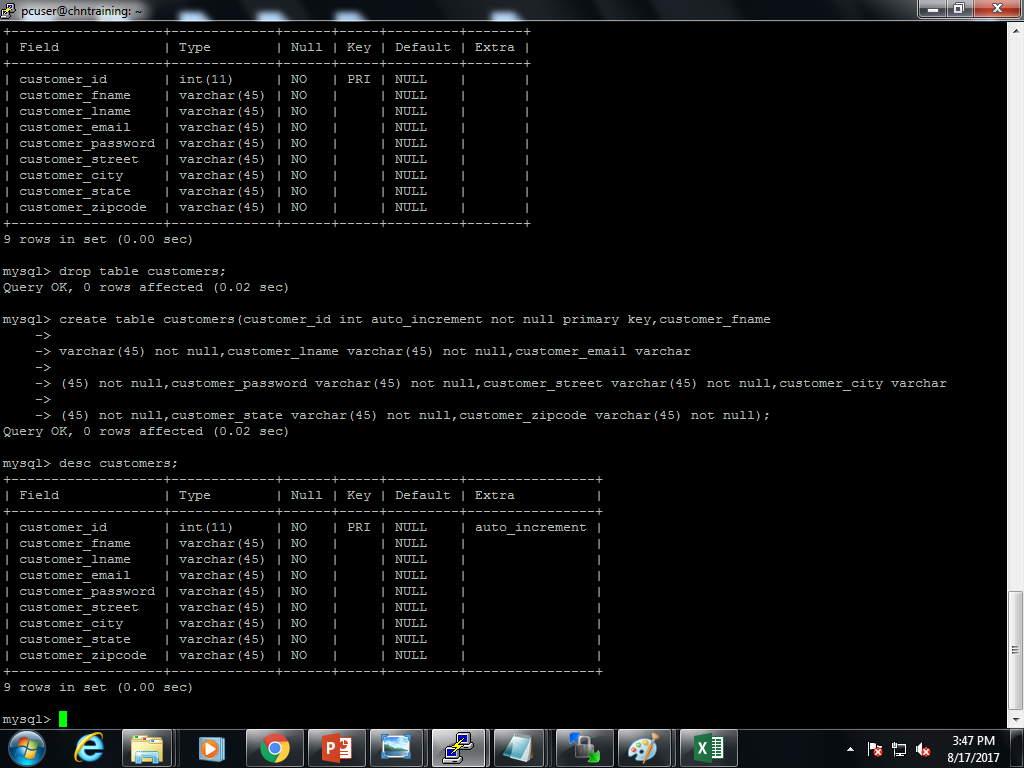
Primary key : department\_id

Foreign key : none

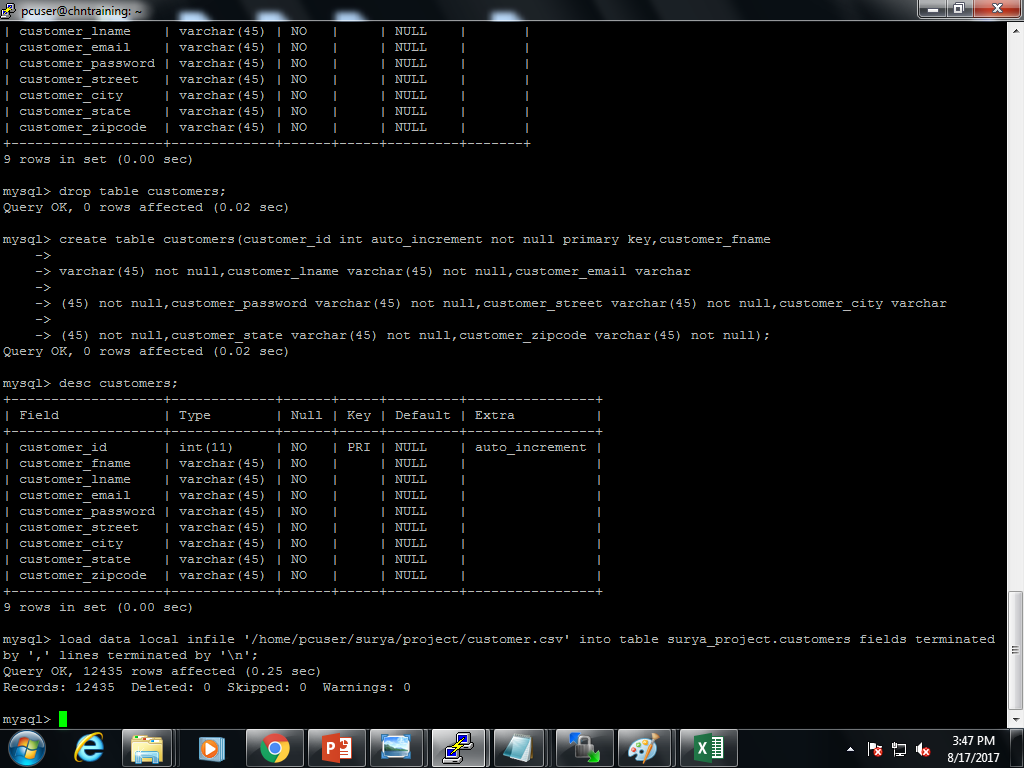
References Table :none

**Creating tables in mysql:**

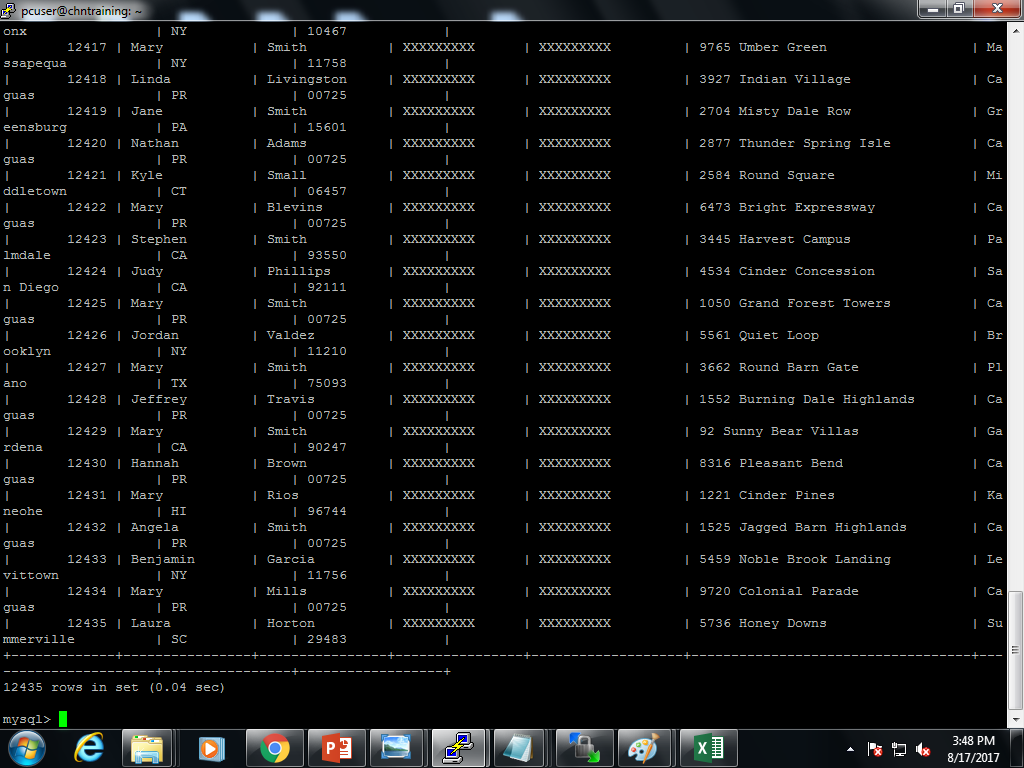
1)>create table customers(customer\_id int auto\_increment not null primary key,customer\_fname varchar(45) not null,customer\_lname varchar(45) not null,customer\_email varchar(45) not null,customer\_password varchar(45) not null,customer\_street varchar(45) not null,customer\_city varchar(45) not null,customer\_state varchar(45) not null,customer\_zipcode varchar(45) not null);



> load data local infile '/home/pcuser/surya/project/customer.csv' into table surya\_project.customers fields terminated by ',' lines terminated by '\n';



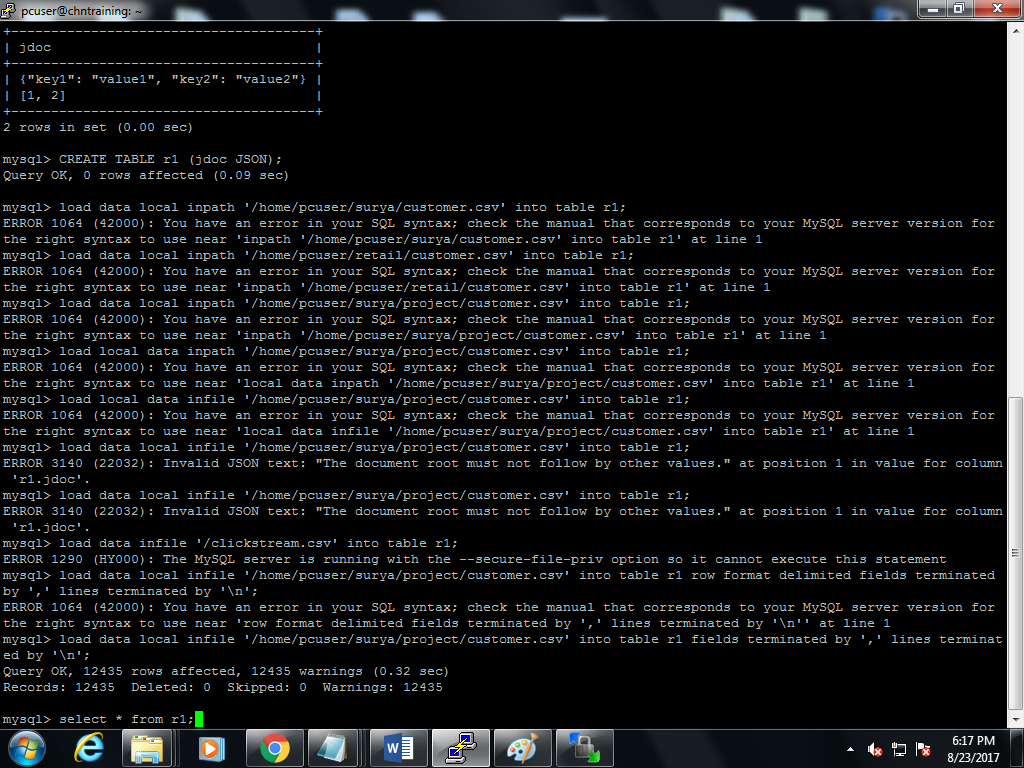
Select \* from customers;

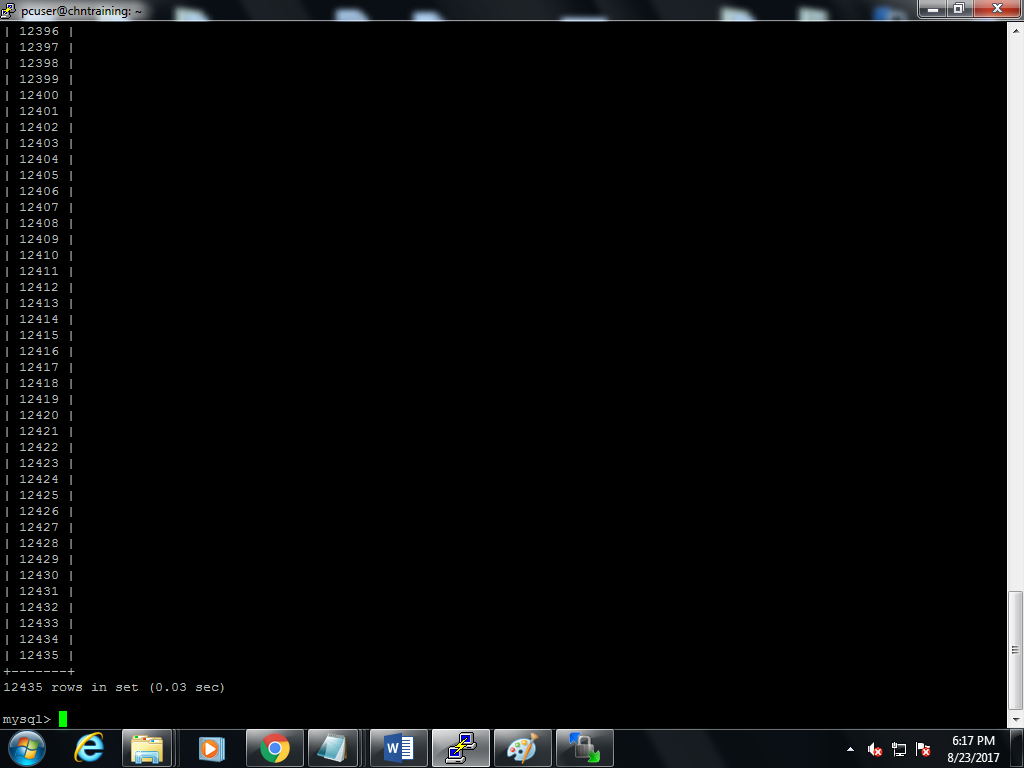
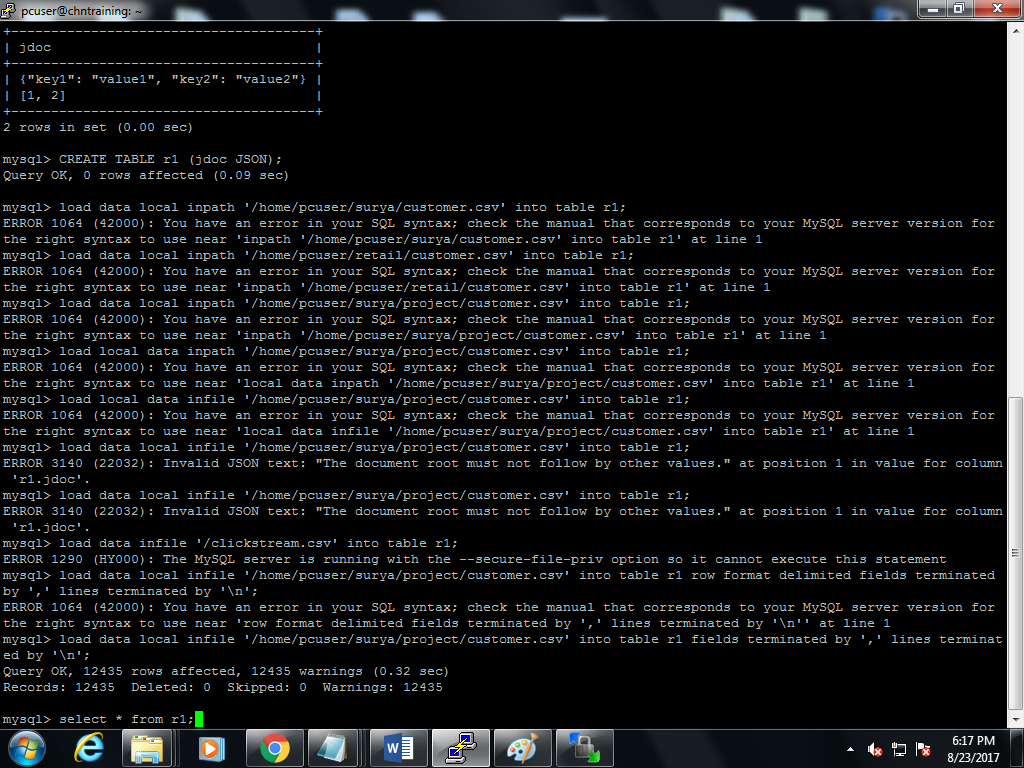


//supposed to be tried in hive because of complex datatype.

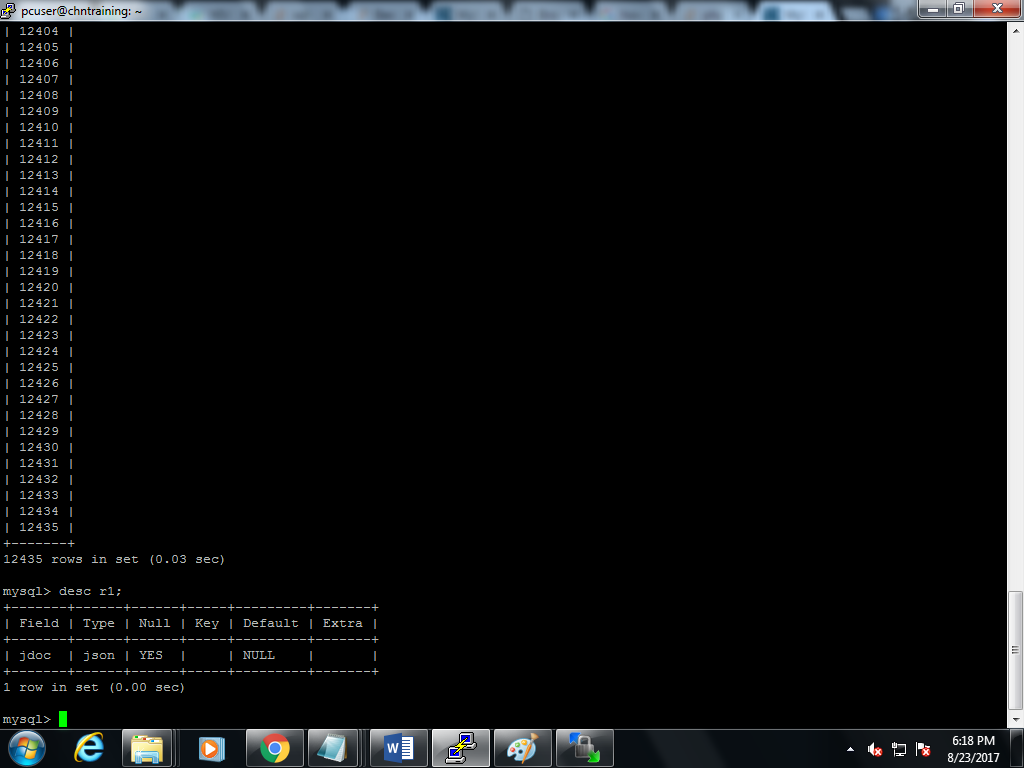
//implemented in mysql until discovered that it had to be done in hive,because mysql doesn’t support complex datatypes.

**//tried json datatype in mysql:**



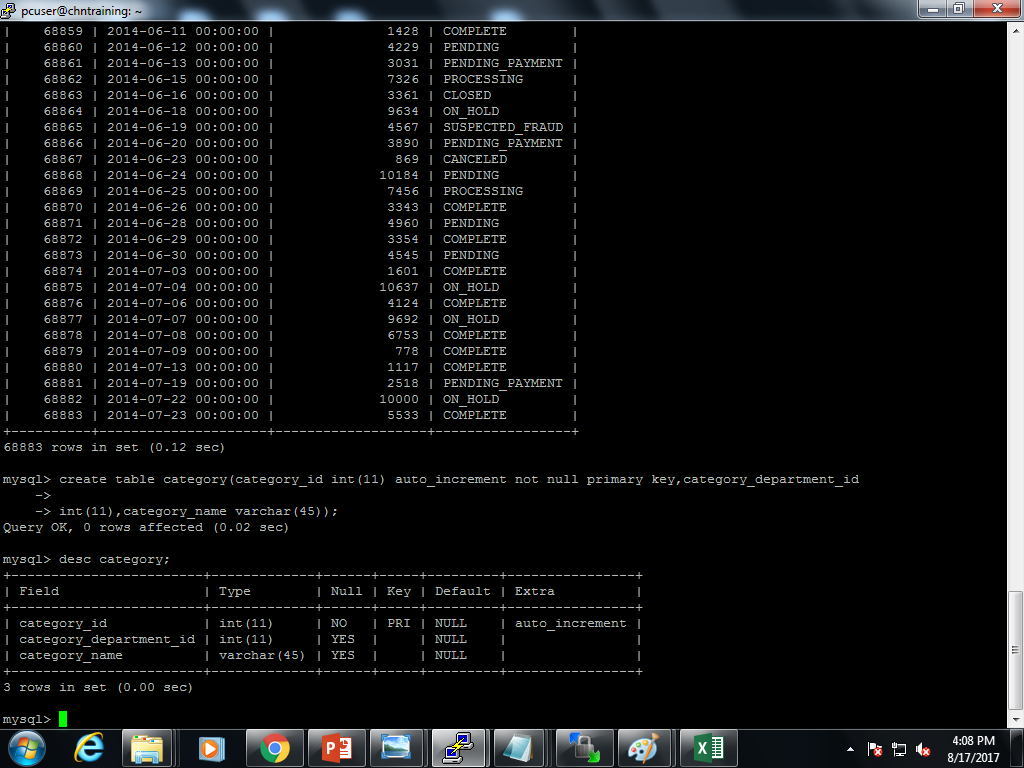


>Select \* from r1;

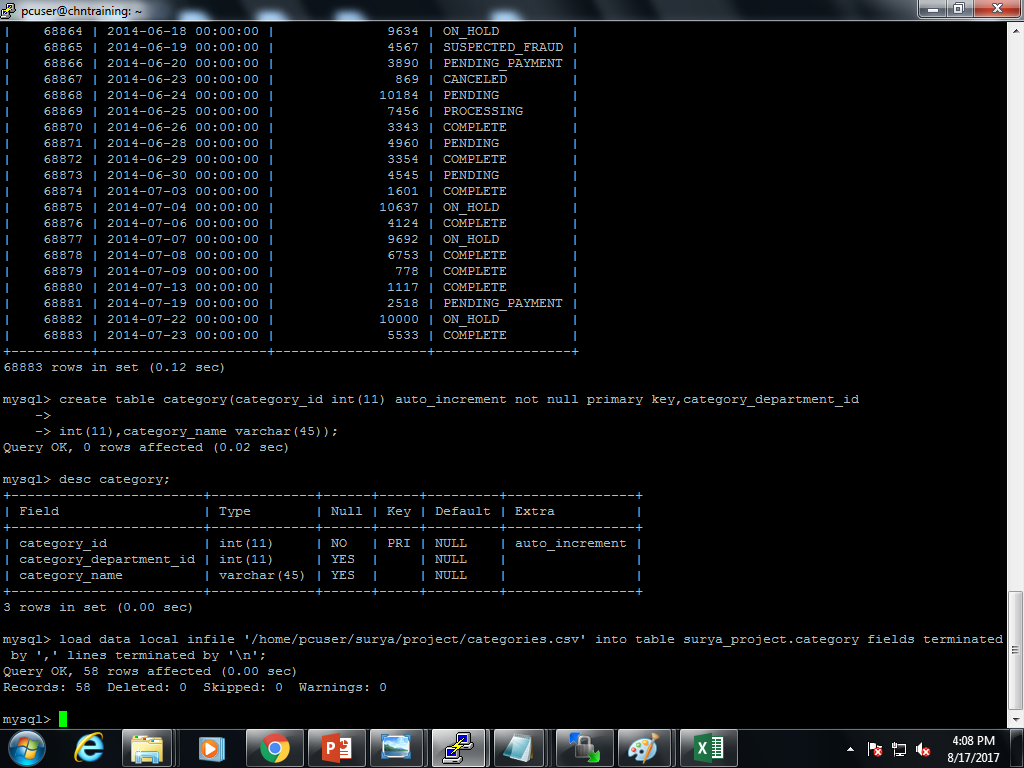
>desc r1;

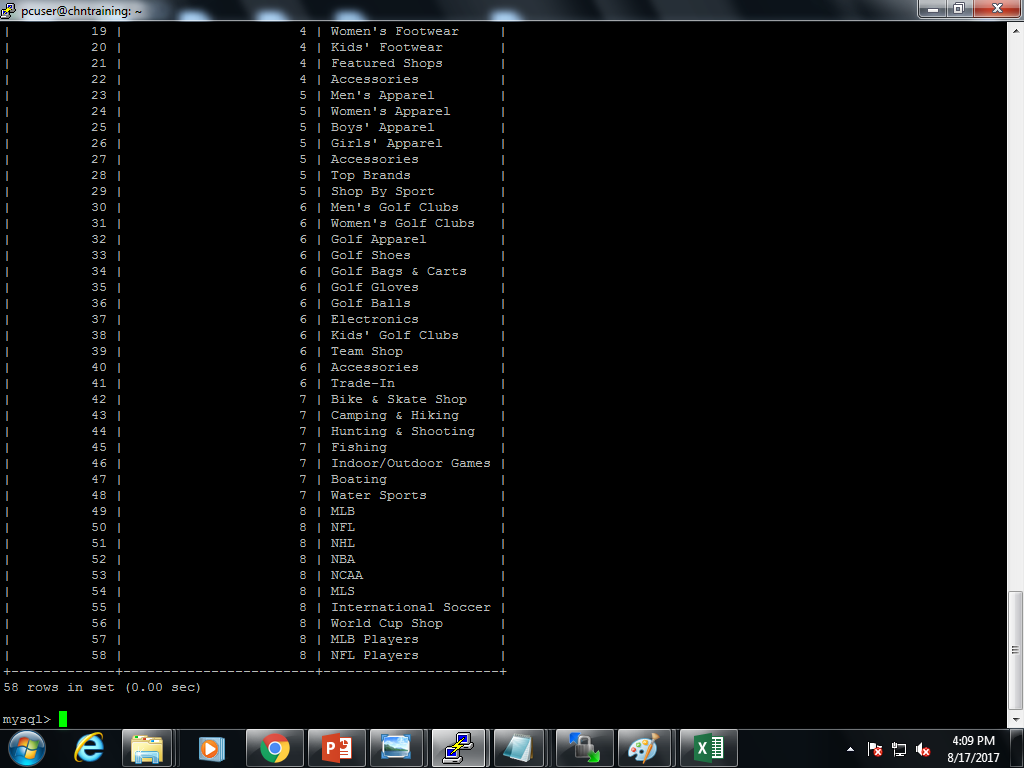
Only the customer\_id is fetched thus the last custmomer\_id =12435.

2) >create table category (category\_id int(11) auto\_increment not null primary key,category\_department\_id int(11),category\_name varchar(45));

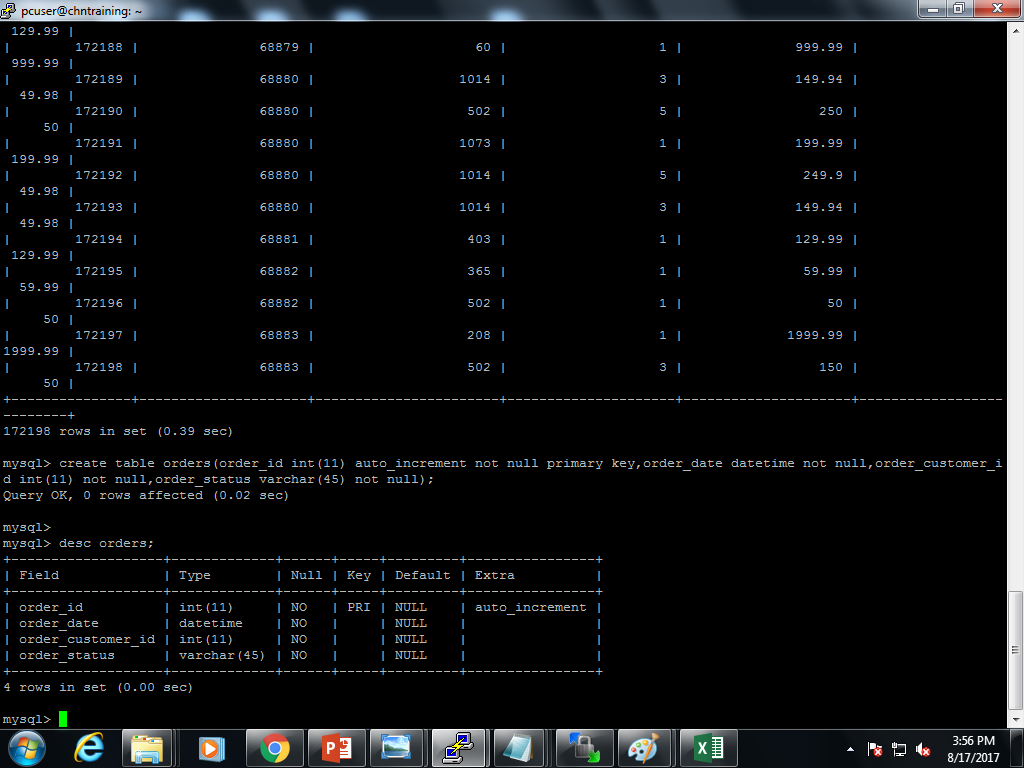


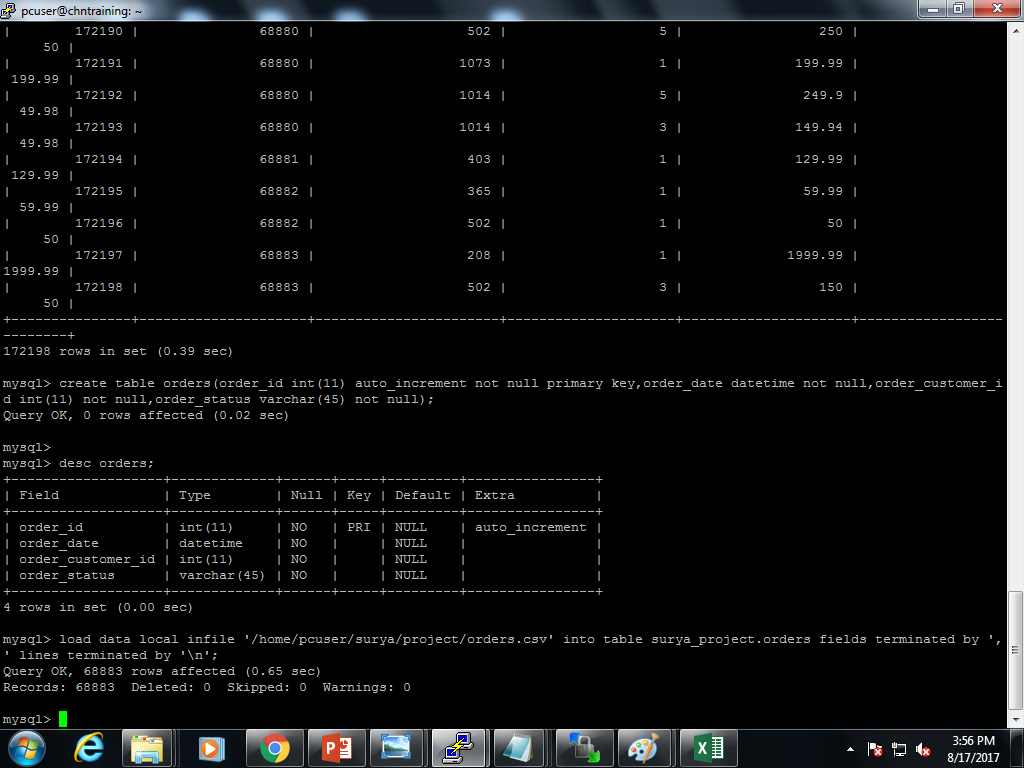
>load data local infile '/home/pcuser/surya/project/categories.csv' into table surya\_project.category fields terminated by ',' lines terminated by '\n';

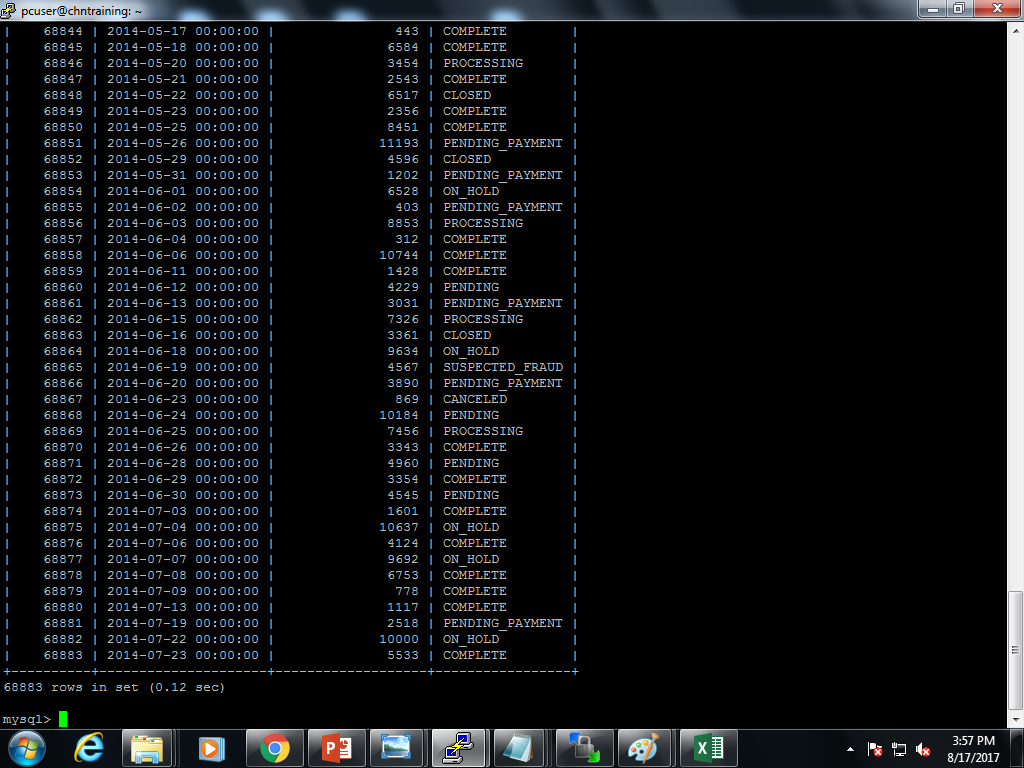




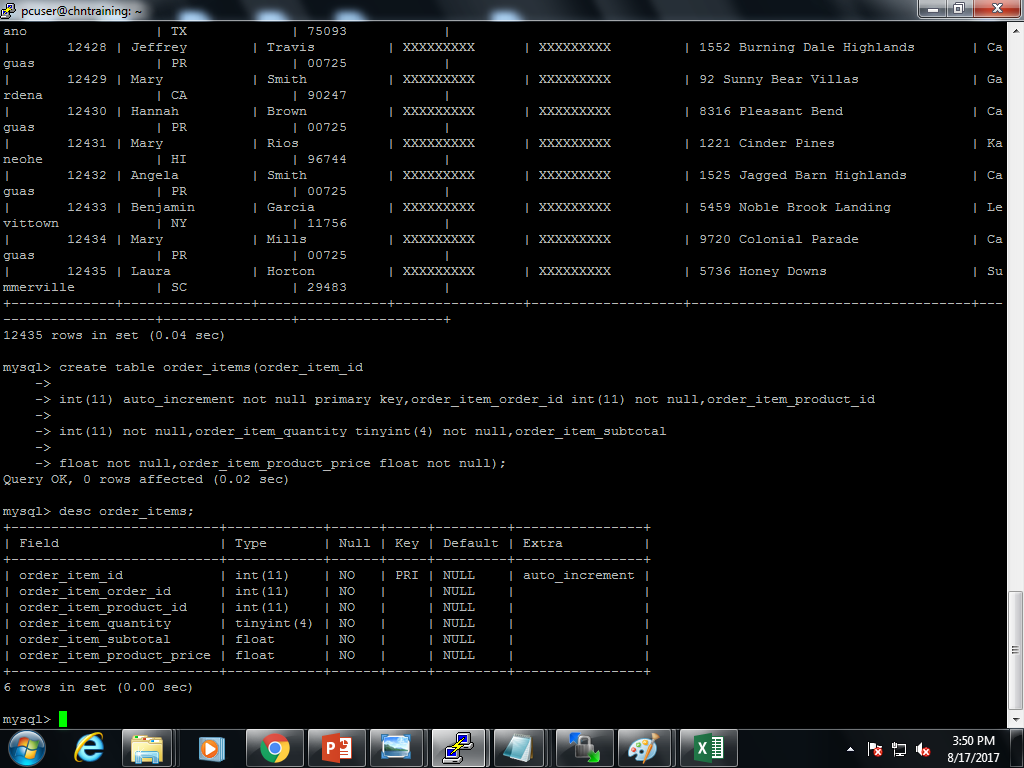
>select \* from category;

3) Create table orders (order\_id int(11) auto\_increment not null primary key,order\_date datetime not null,order\_customer\_id int(11) not null, order\_status varchar(45) not null); 

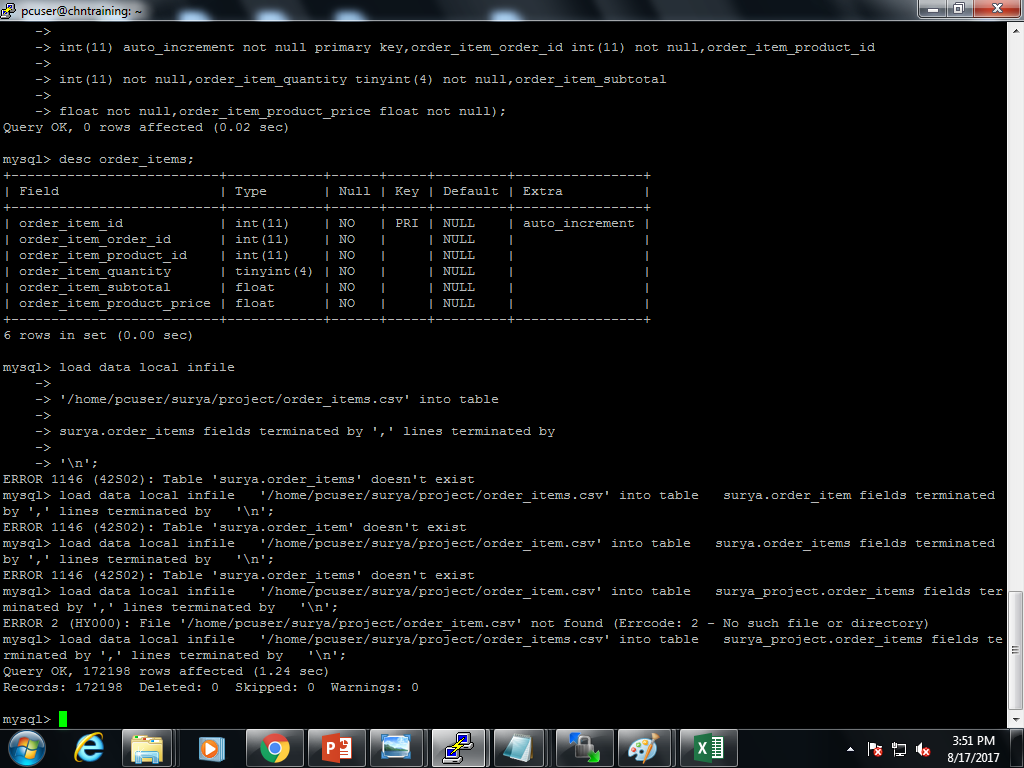
>load data local infile '/home/pcuser/surya/project/orders.csv' into table surya\_project.orders fields terminated by ',' lines terminated by '\n'; 



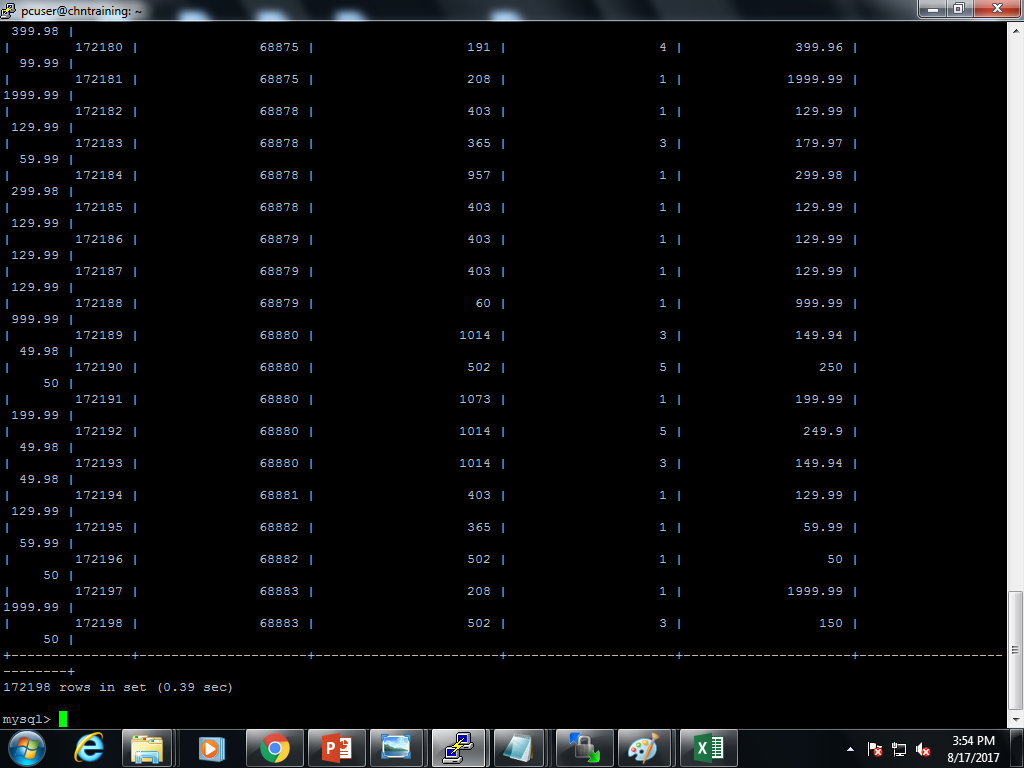
>select \* from order;

4) >create table order\_items(order\_item\_id int(11) auto\_increment not null primary key,order\_item\_order\_id int(11) not null,order\_item\_product\_id int(11) not null,order\_item\_quantity tinyint(4) not null,order\_item\_subtotal float not null,order\_item\_product\_price float not null); 

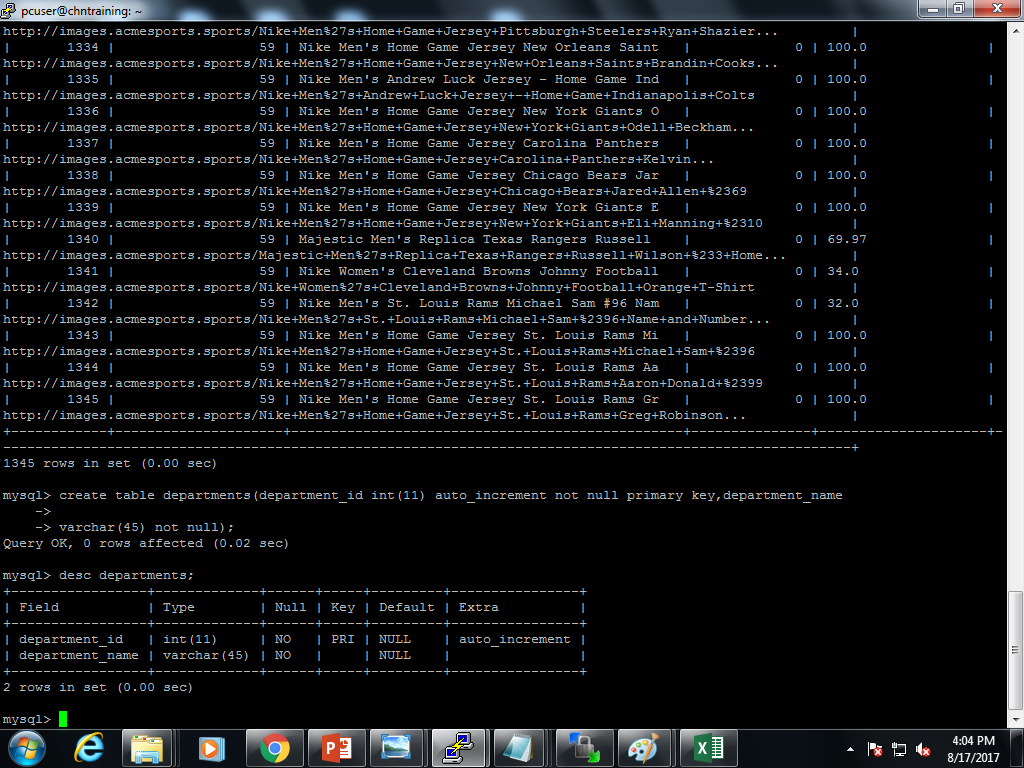
>load data local infile '/home/pcuser/surya/project/order\_items.csv' into table surya\_project.order\_items fields terminated by ',' lines terminated by '\n';



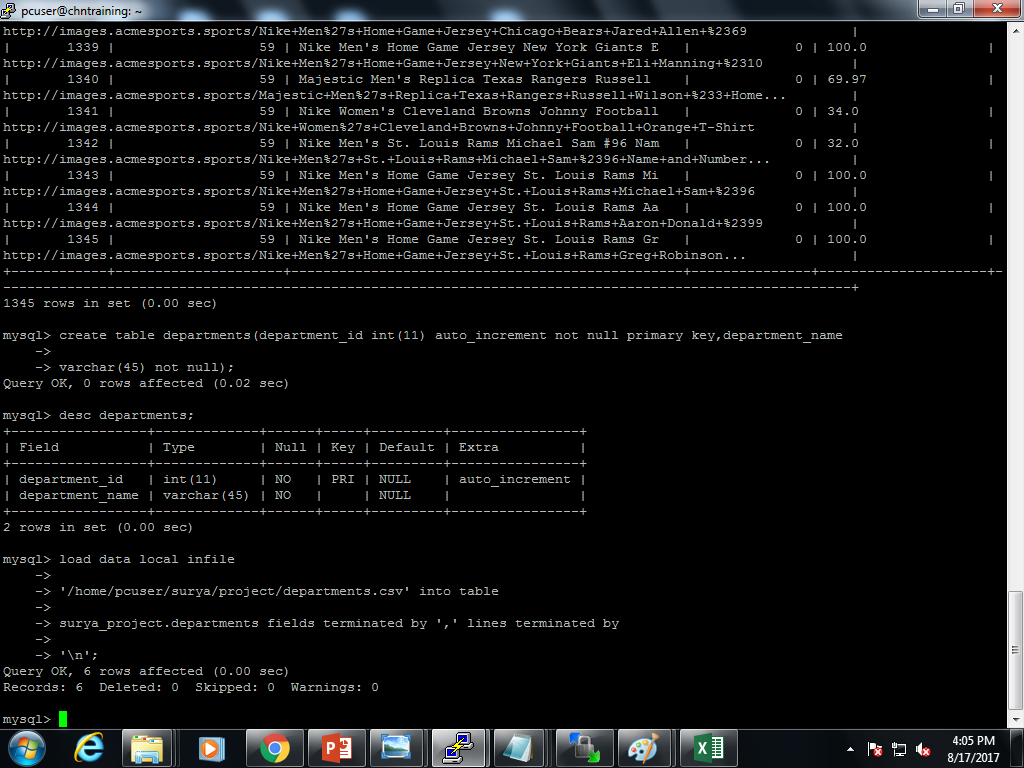
>select \* from order\_items;



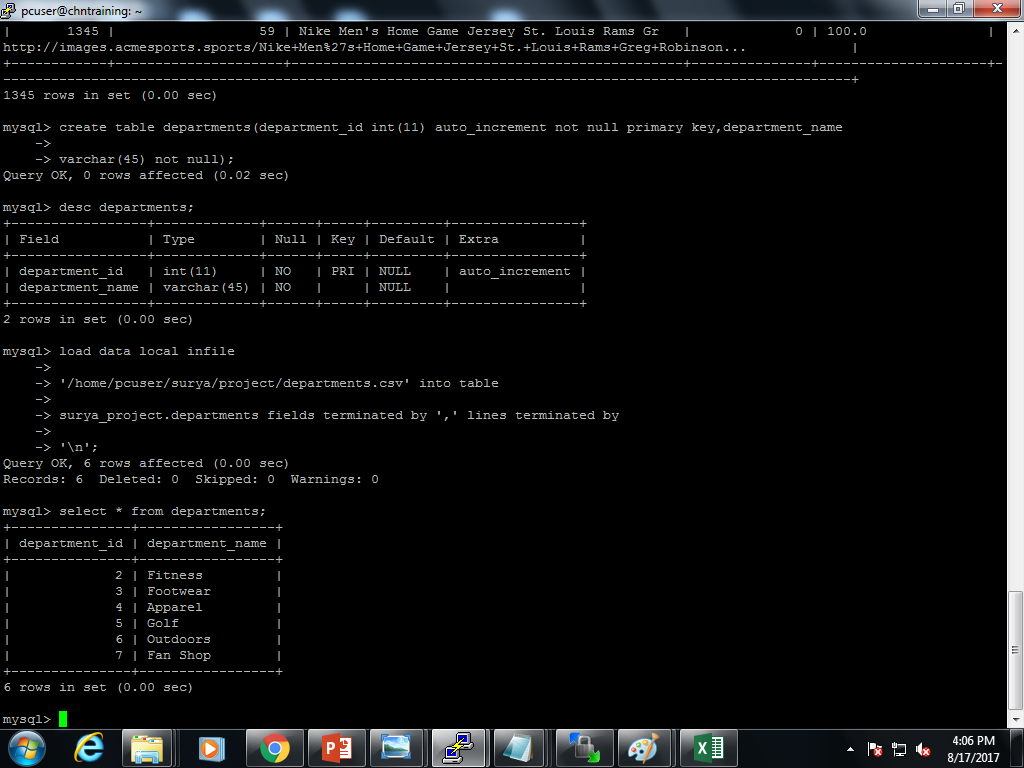
5) >create table departments (department\_id int(11) auto\_increment not null primary key,department\_name varchar(45) not null);



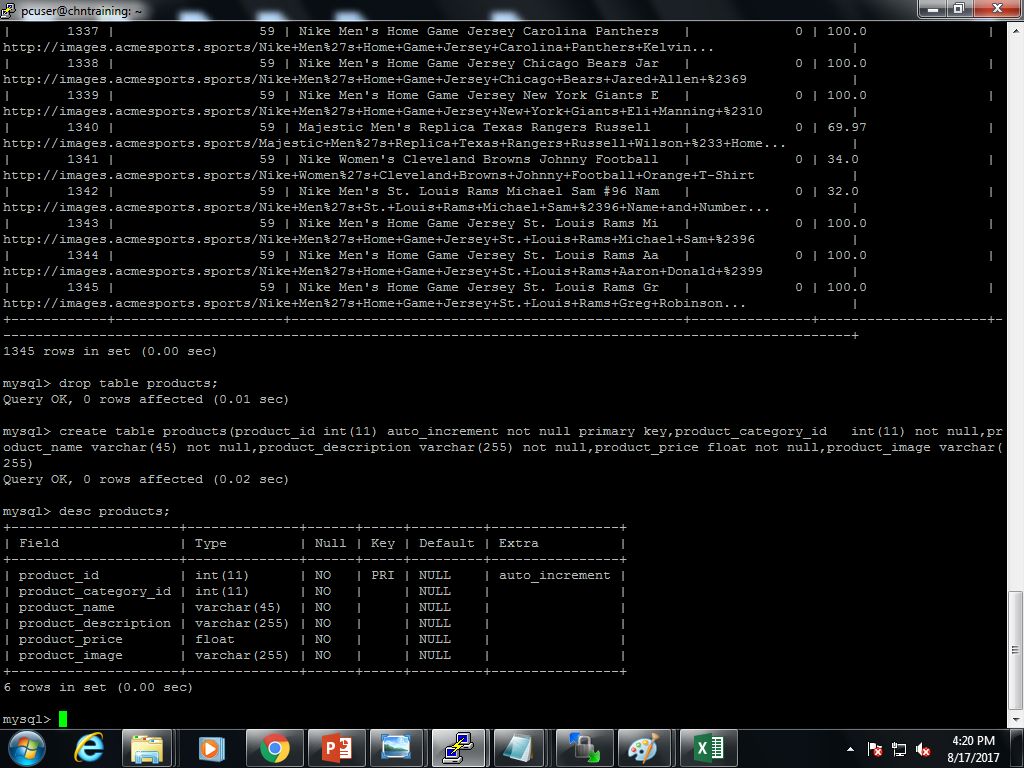
>load data local infile '/home/pcuser/surya/project/departments.csv' into table surya\_project.departments fields terminated by ',' lines terminated by '\n';



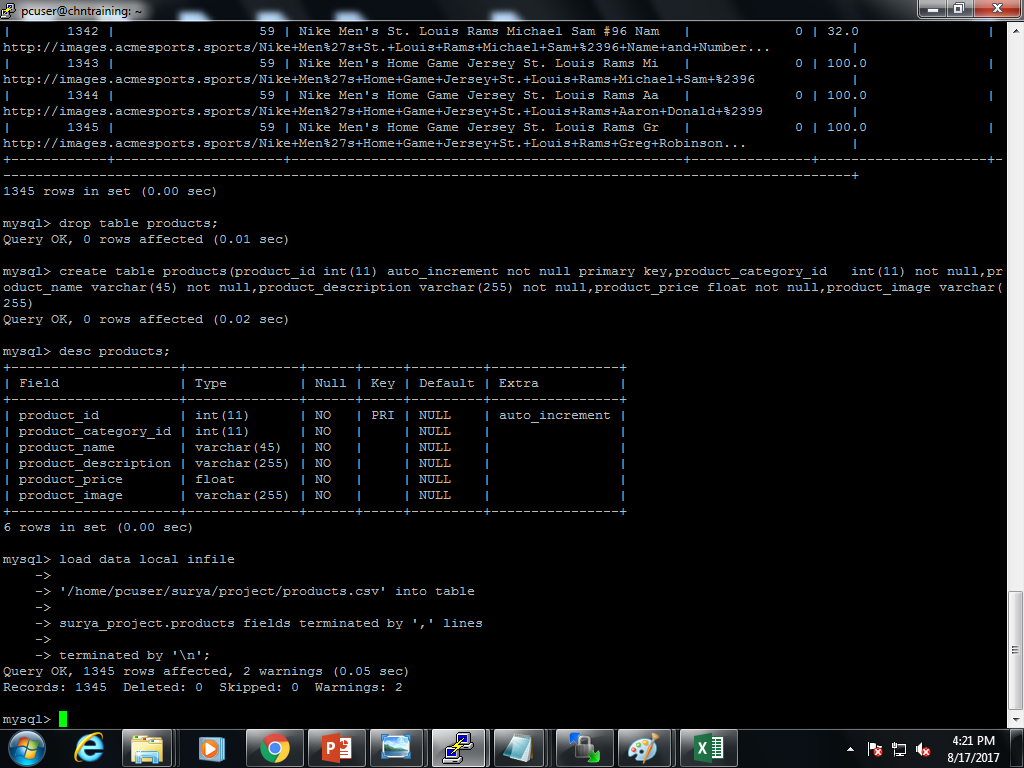
>select \* from departments;

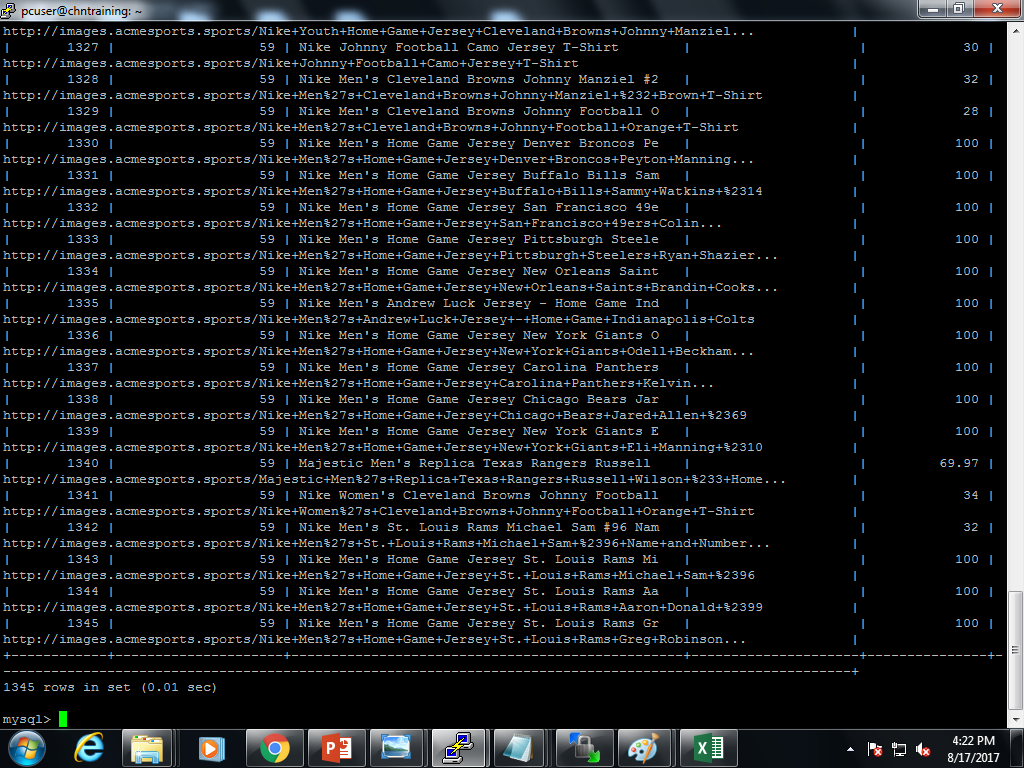


6) >create table products(product\_id int(11) auto\_increment not null primary key,product\_category\_id int(11) not null,product\_name varchar(45) not null,product\_description varchar(255) not null,product\_price float not null,product\_image varchar(255) not null);



>load data local infile '/home/pcuser/surya/project/products.csv' into table surya\_project.products fields terminated by ',' lines terminated by '\n';



>Select \* from products;

**HIVE**

>hive

>show databases;

>use surya\_project;

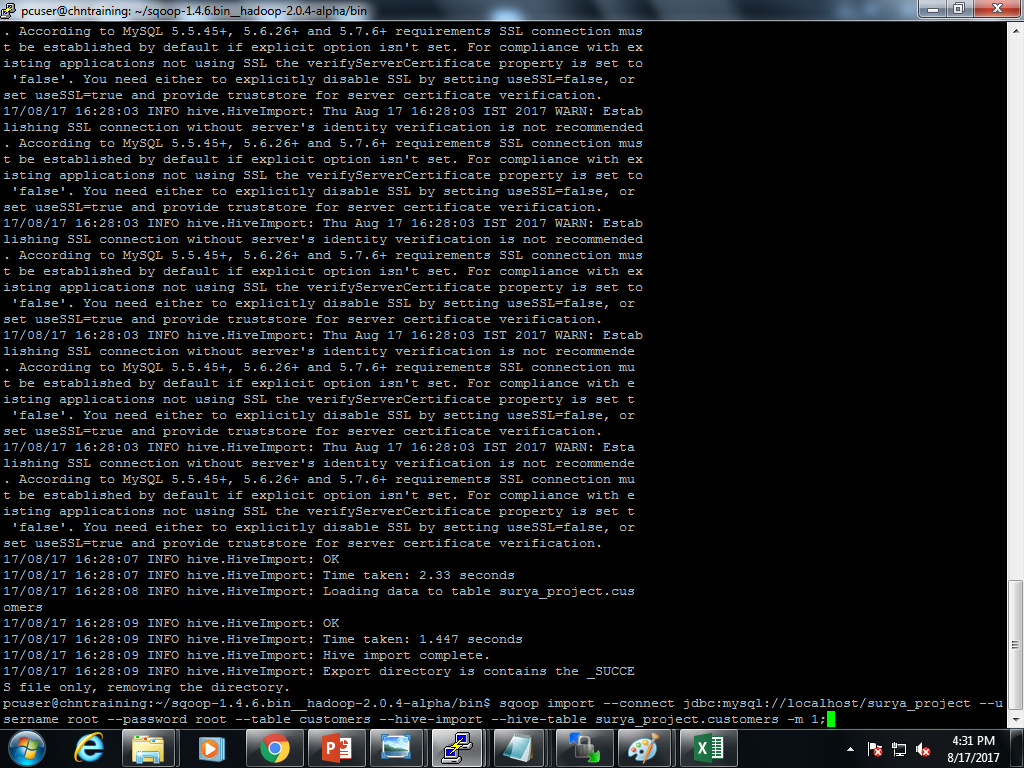
//after importing tables from sqoop

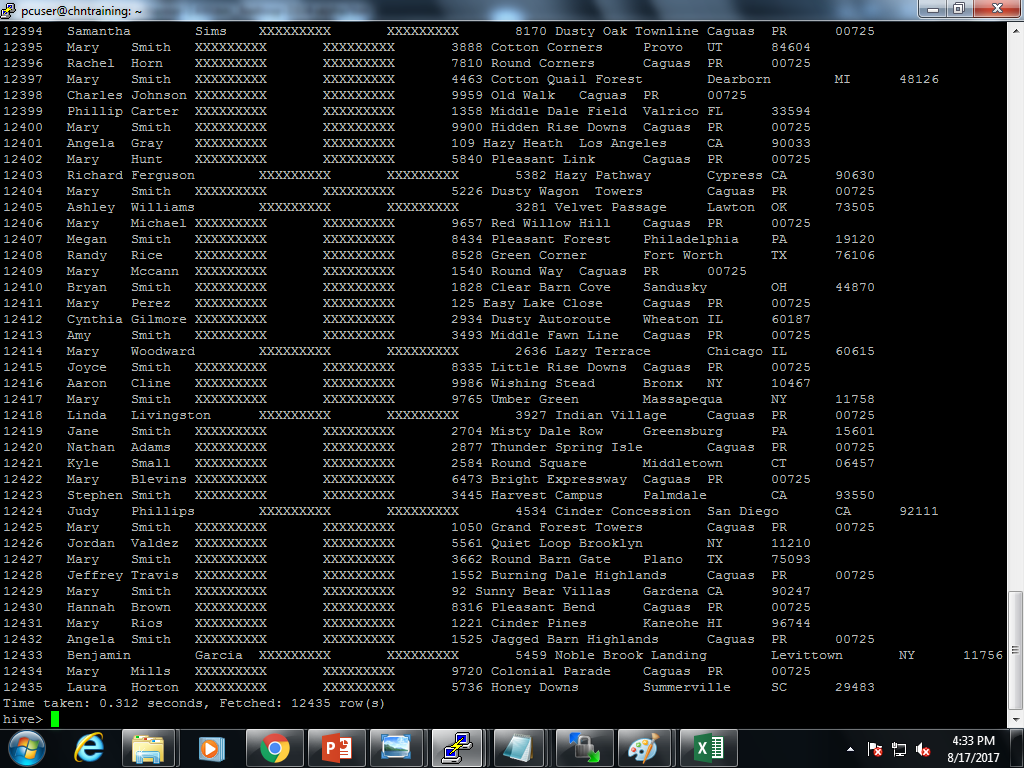
>select \* from customers;

>select \* from category;

***Loading the data from mysql to hive:***

1)sqoop bin$>sqoop import --connect jdbc:mysql://localhost/surya\_project --username root --password root --table customers --hive-import --hive-table surya\_project.customers -m 1;

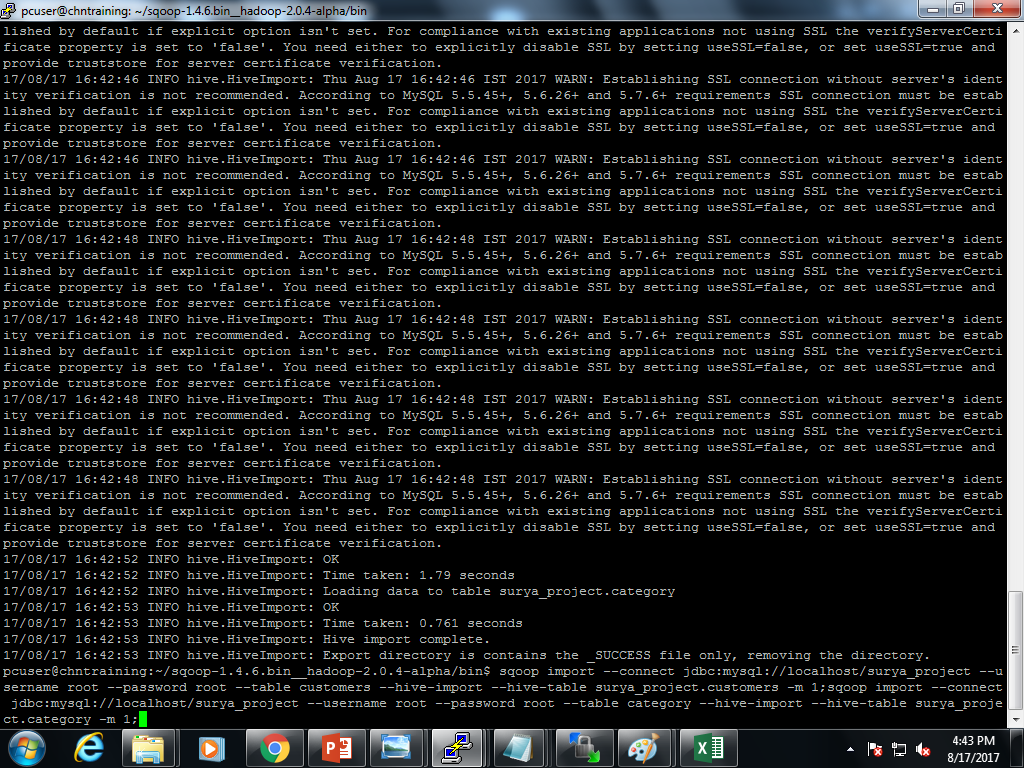
****

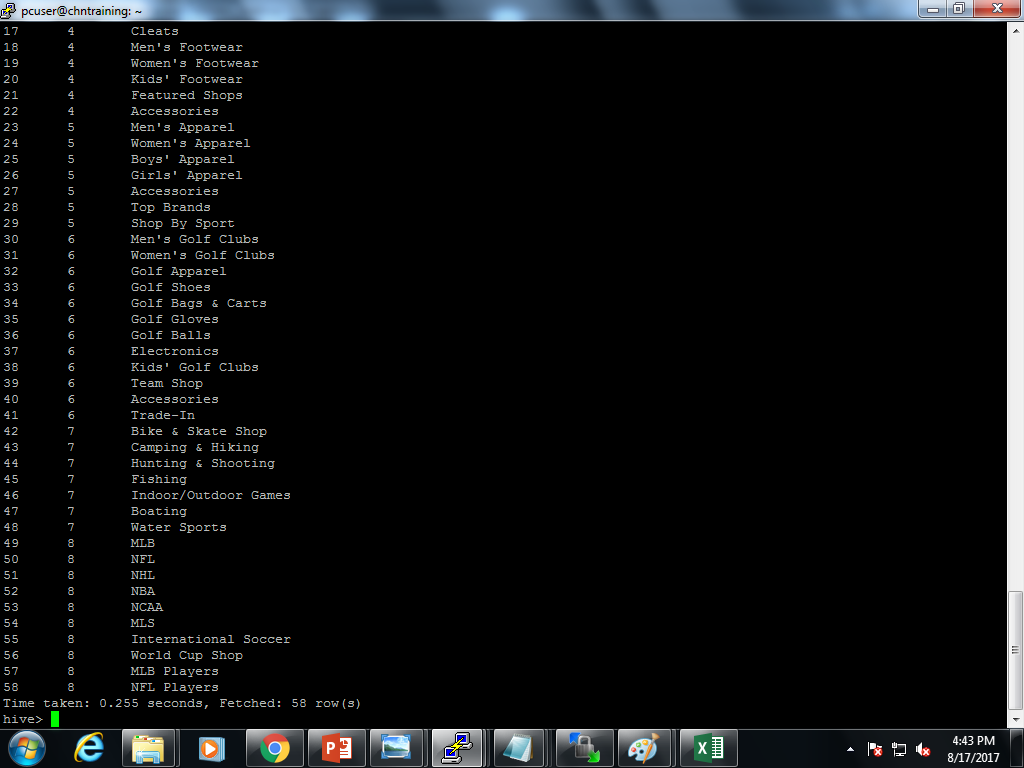
****

2)sqoop bin$>sqoop import --connect jdbc:mysql://localhost/surya\_project --username root --password root --table category --hive-import --hive-table surya\_project.category -m 1;

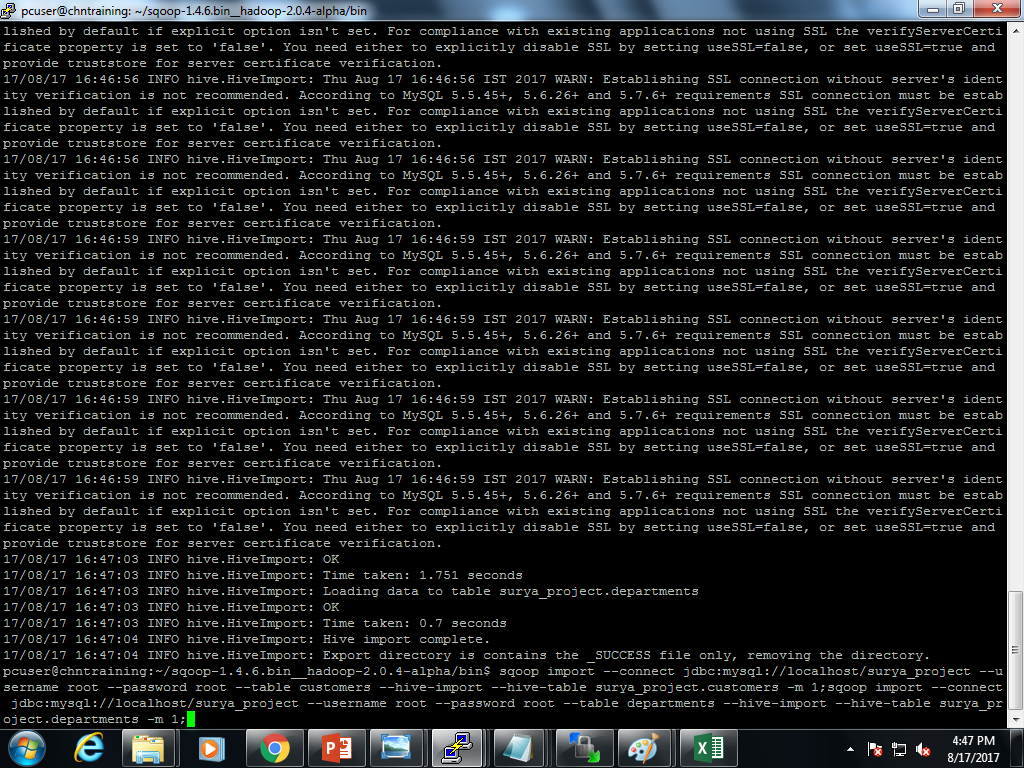
//username is set as root.

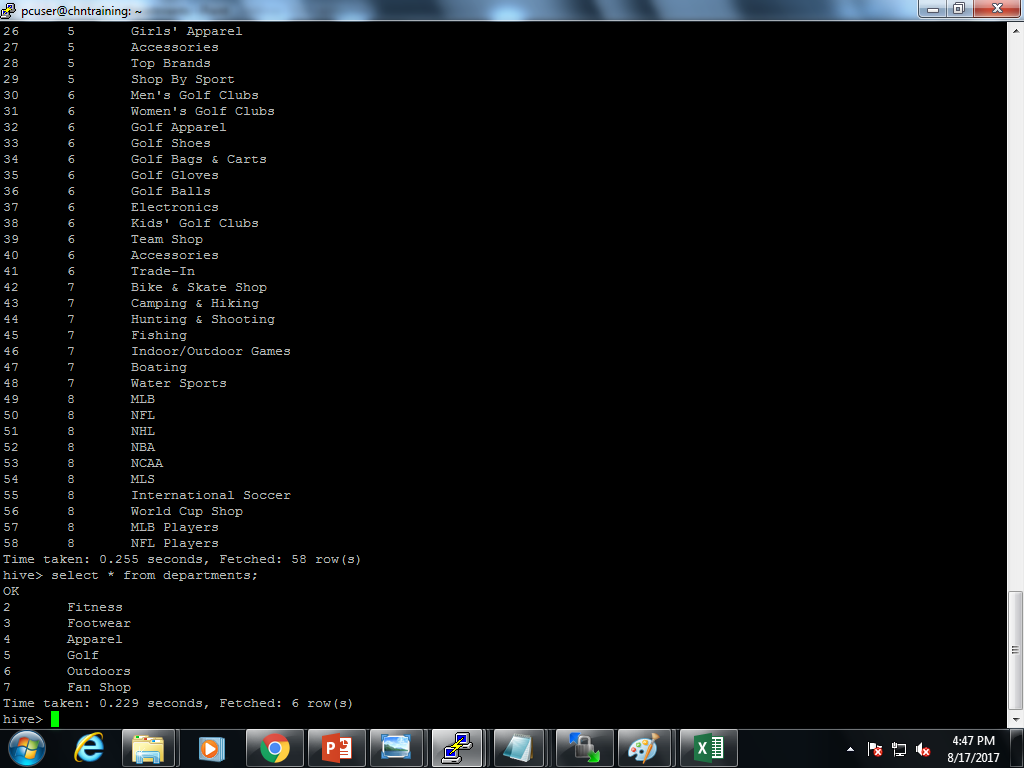
//password is set as root.

****

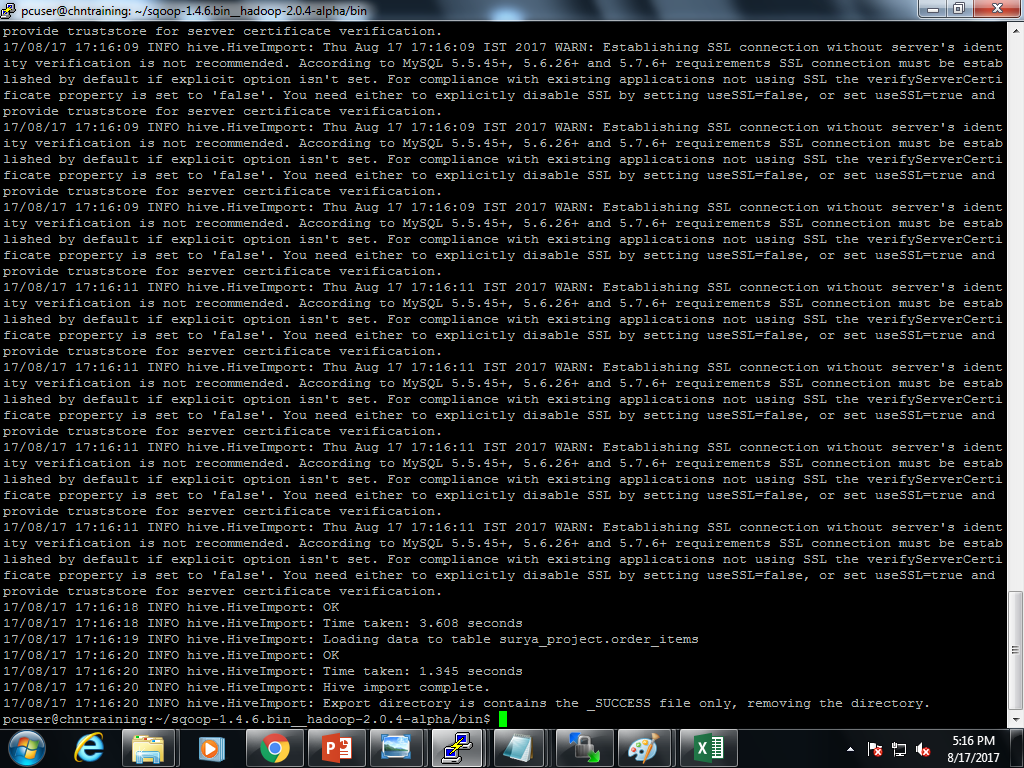
****

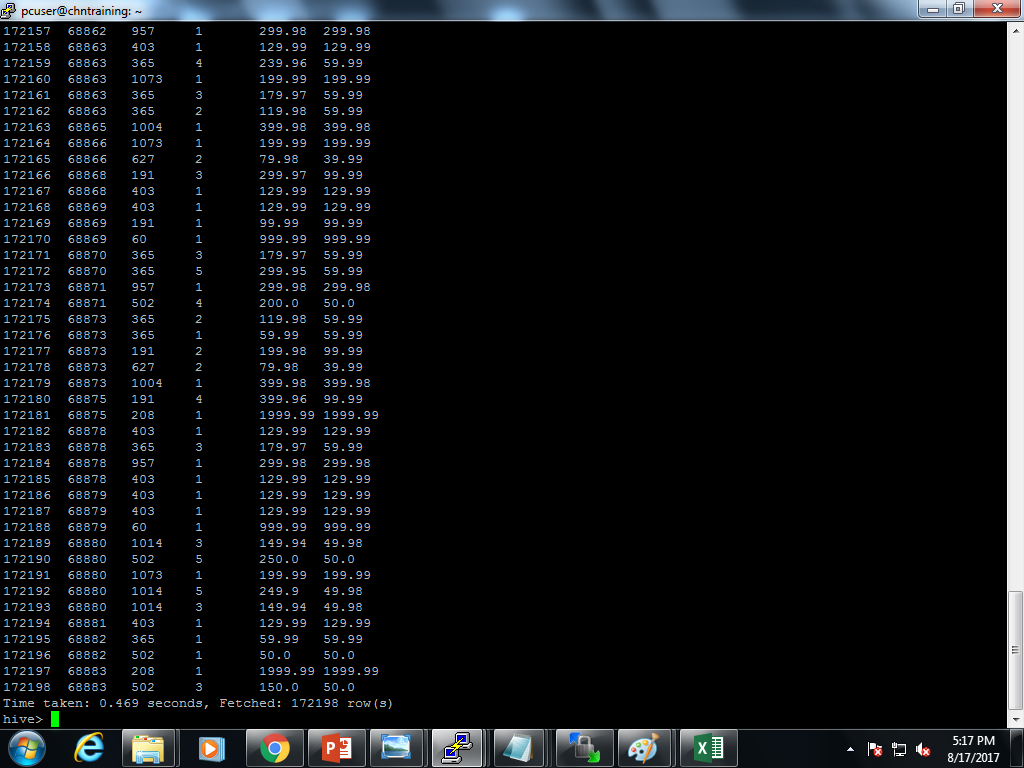
3)sqoop bin$>sqoop import --connect jdbc:mysql://localhost/surya\_project --username root --password root --table departments --hive-import --hive-table surya\_project.departments -m 1;

****

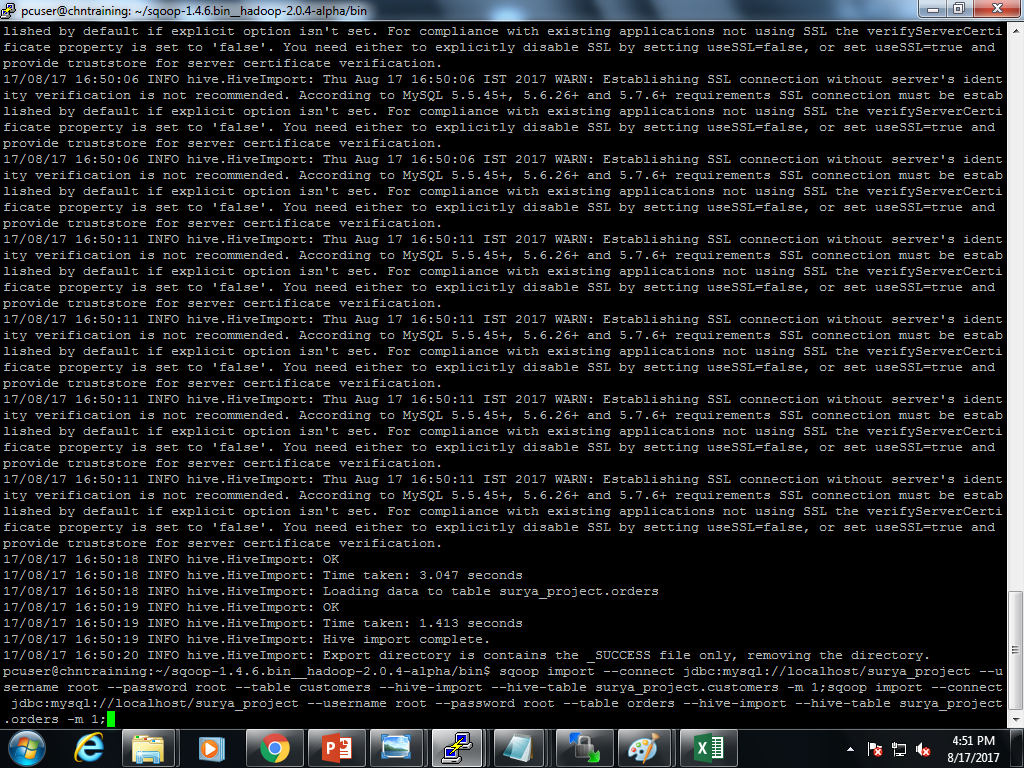
****

4)sqoop bin$>sqoop import --connect jdbc:mysql://localhost/surya\_project --username root --password root --table order\_items --hive-import --hive-table surya\_project.order\_items -m 1;

****

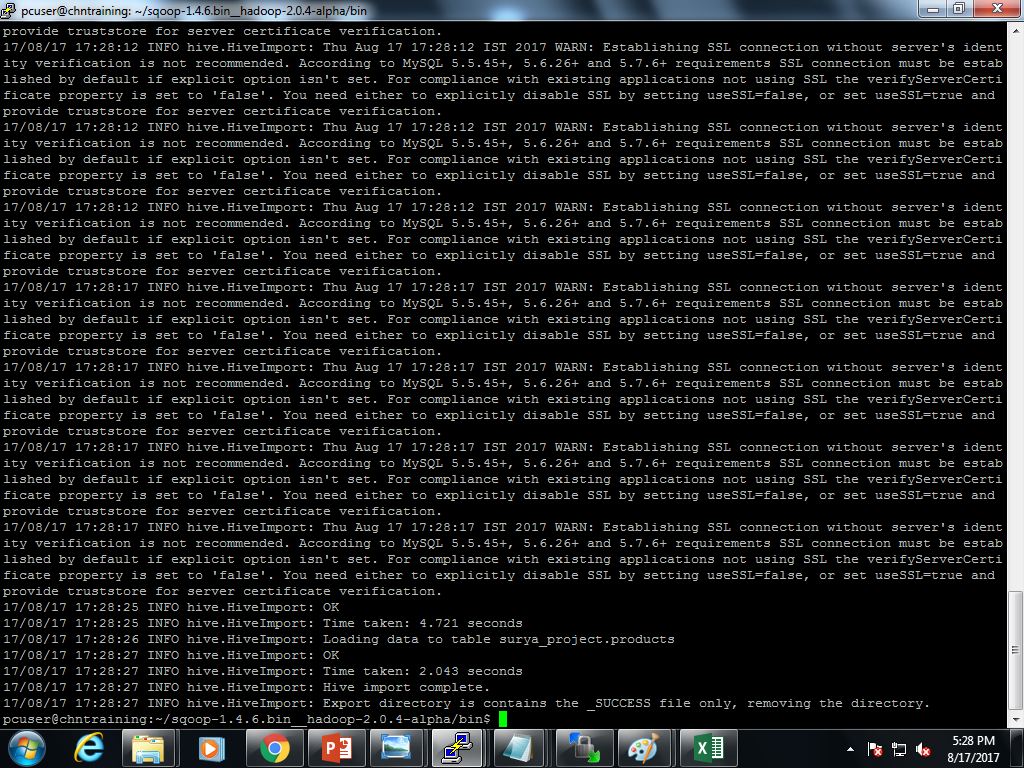
****

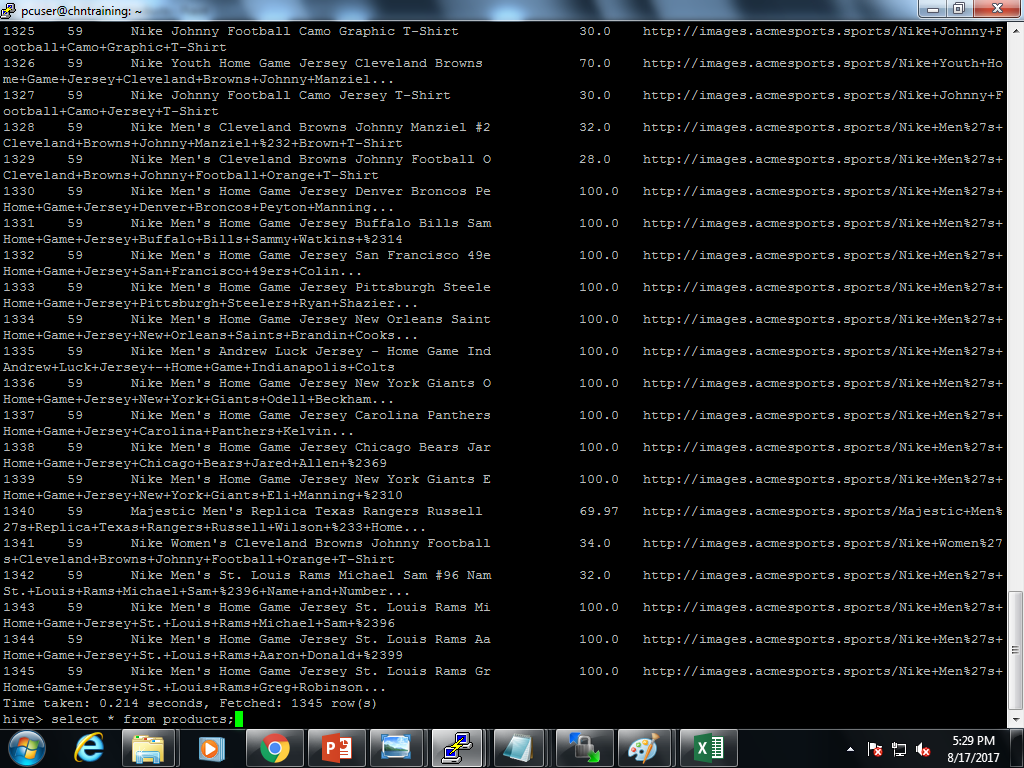
5)sqoop bin$>sqoop import --connect jdbc:mysql://localhost/surya\_project --username root --password root --table orders --hive-import --hive-table surya\_project.orders -m 1;

****

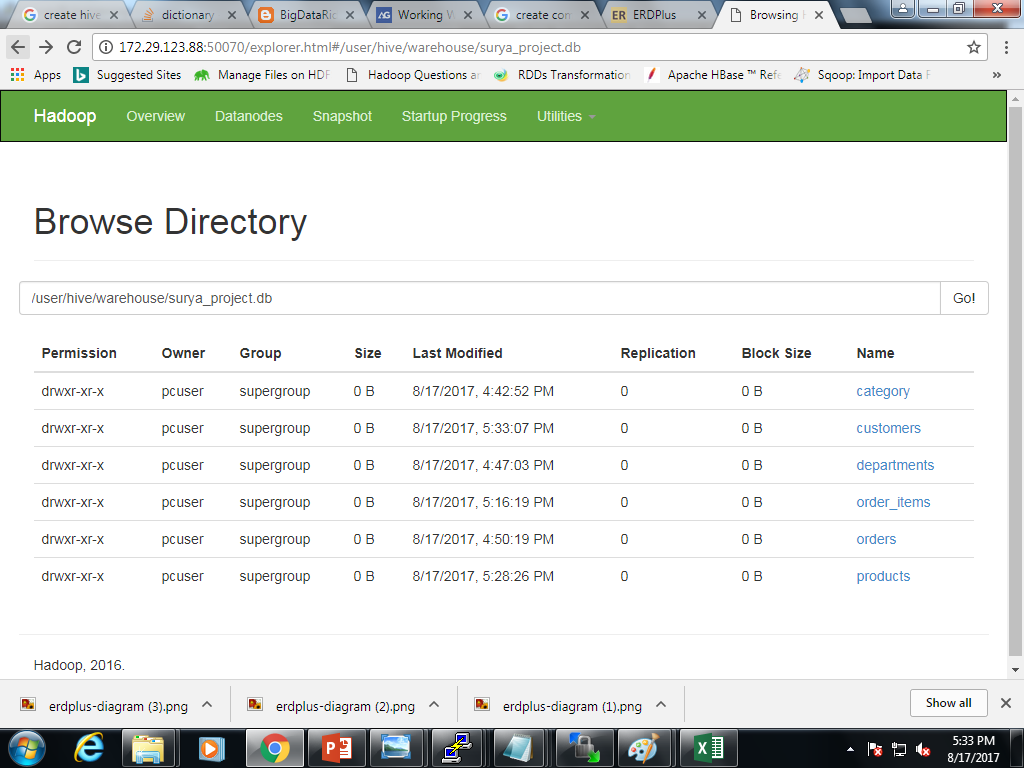
****

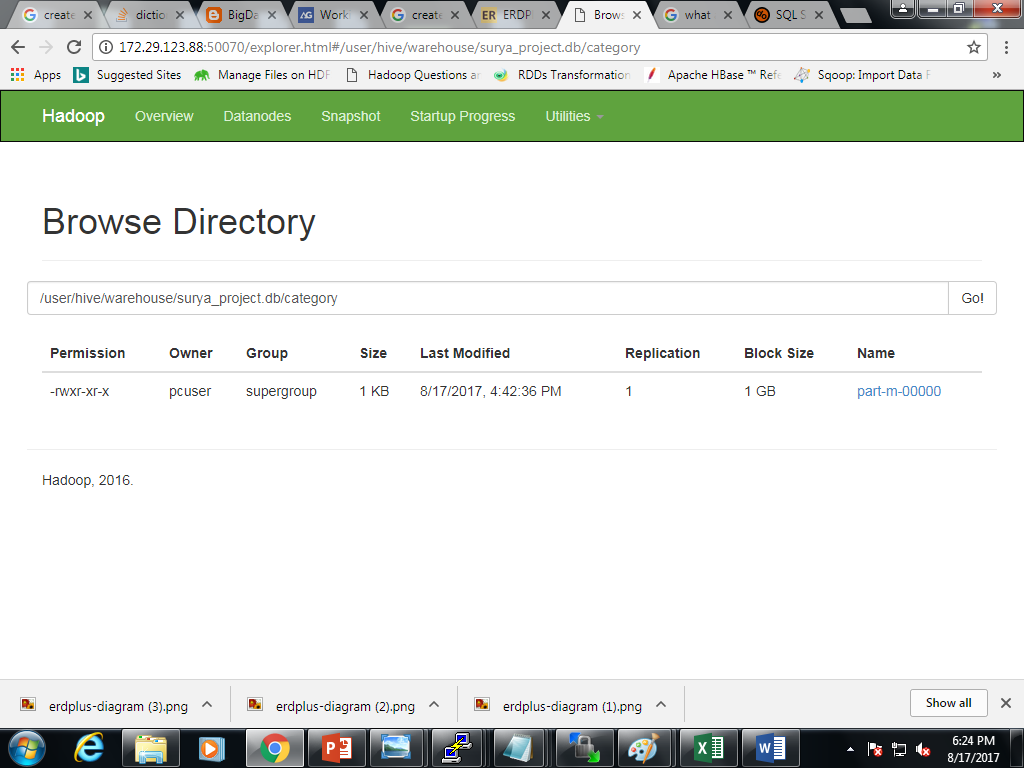
6)sqoop bin$>sqoop import --connect jdbc:mysql://localhost/surya\_project --username root --password root --table products --hive-import --hive-table surya\_project.products -m 1;

****

****

**Hdfs OUTPUT:** **/user/hive/warehouse/surya\_project.db**

****



**Category output file.**

**PARTITIONING**

**//to set dynamic partitioning.**

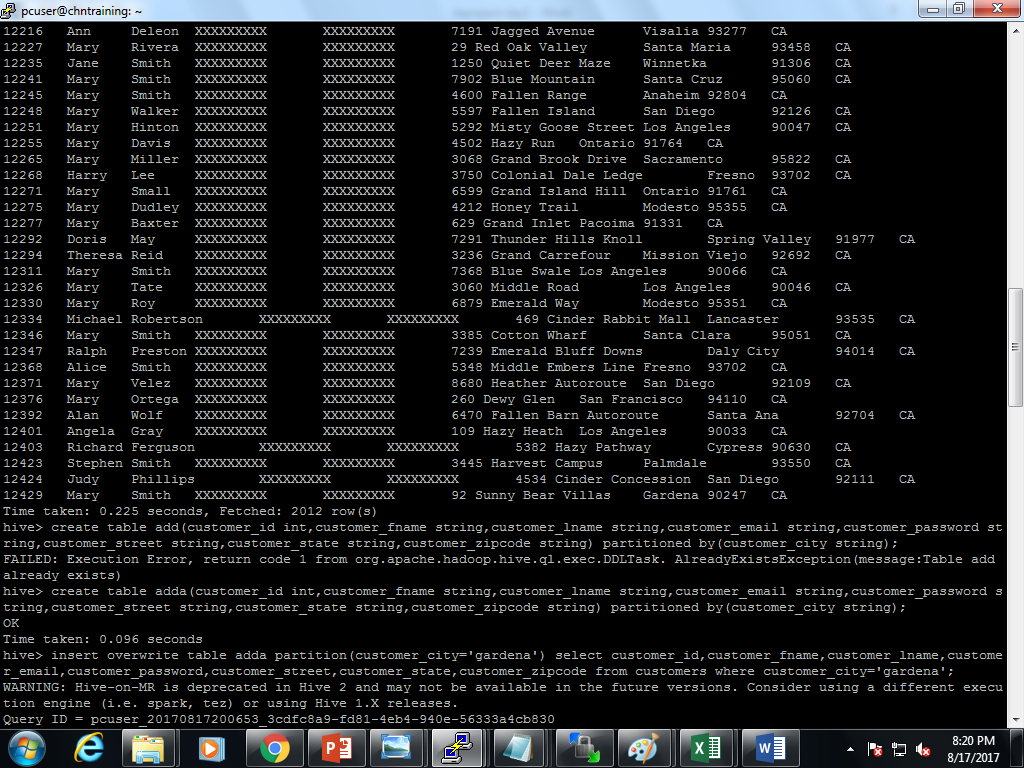
**//default:hive partitioning is static**

>SET hive.exec.dynamic.partition=true;

>SET hive.exec.dynamic.partition.mode = nonstrict;

**1)>**create table add (customer\_id int,customer\_name string,customer\_lname string,customer\_fname string,customer\_email string,customer\_password string,customer\_street string,customer\_city string,customer\_zipcode string) partitioned by (customer\_state string);

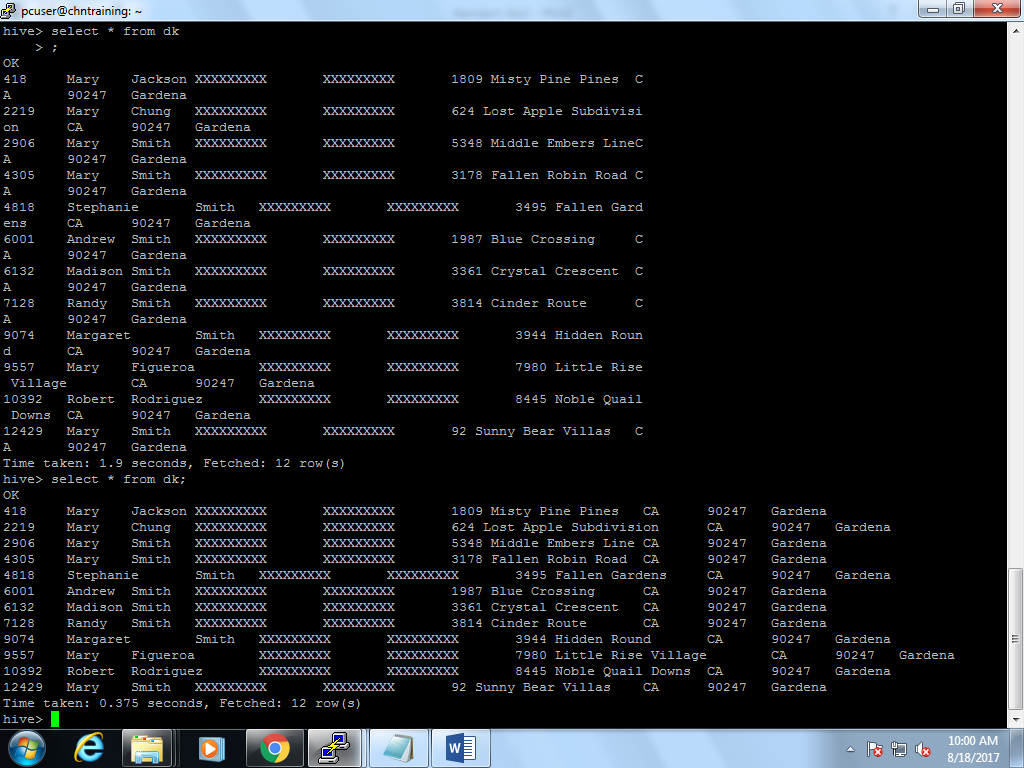
>insert overwrite table add partition(customer\_state='ca') select customer\_id,customer\_name,customer\_lname,customer\_fname,customer\_email,customer\_password,customer\_street,customer\_city,customer\_zipcode from customers where customer\_state = 'ca';



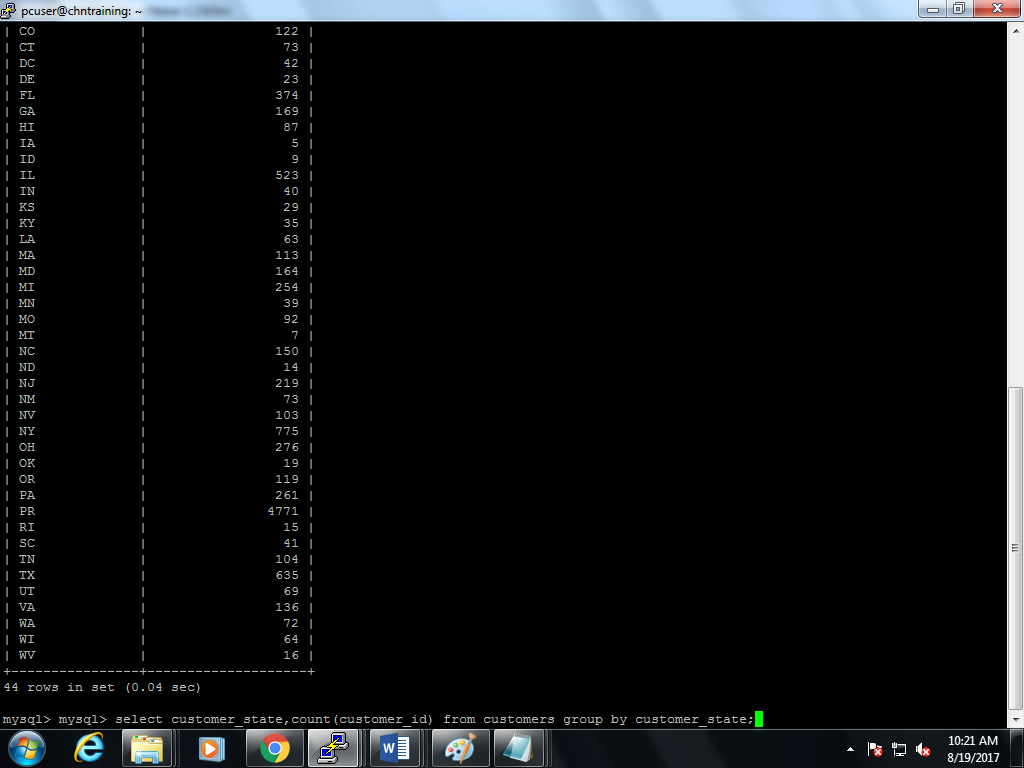
**2)>**create table dk (customer\_id int,customer\_name string,customer\_lname string,customer\_fname string,customer\_email string,customer\_password string,customer\_street string,customer\_state string,customer\_zipcode string) partitioned by (customer\_city string);

>insert overwrite table dk partition (customer\_state='Gardena') select customer\_id,customer\_name,customer\_lname,customer\_fname,customer\_email,customer\_password,customer\_street,customer\_city,customer\_zipcode from customers where customer\_city = 'Gardena';

Select \* from dk;

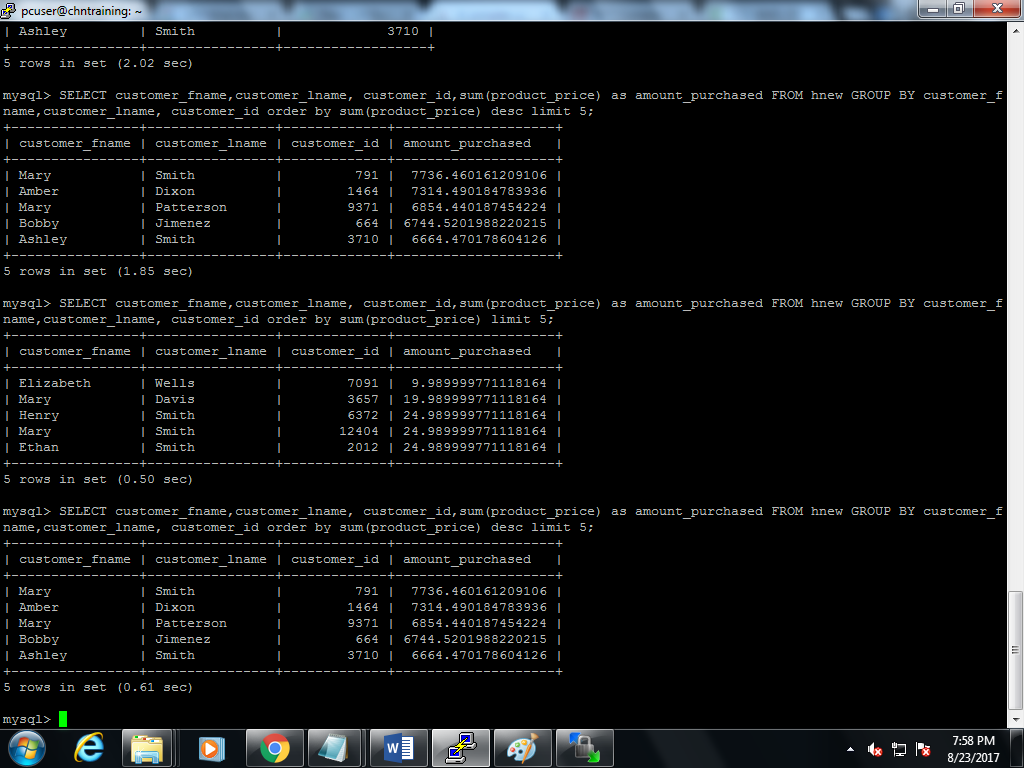


**GROUP BY**

select customer\_state,count(customer\_id) from customers group by customer\_state;

Top five customer:

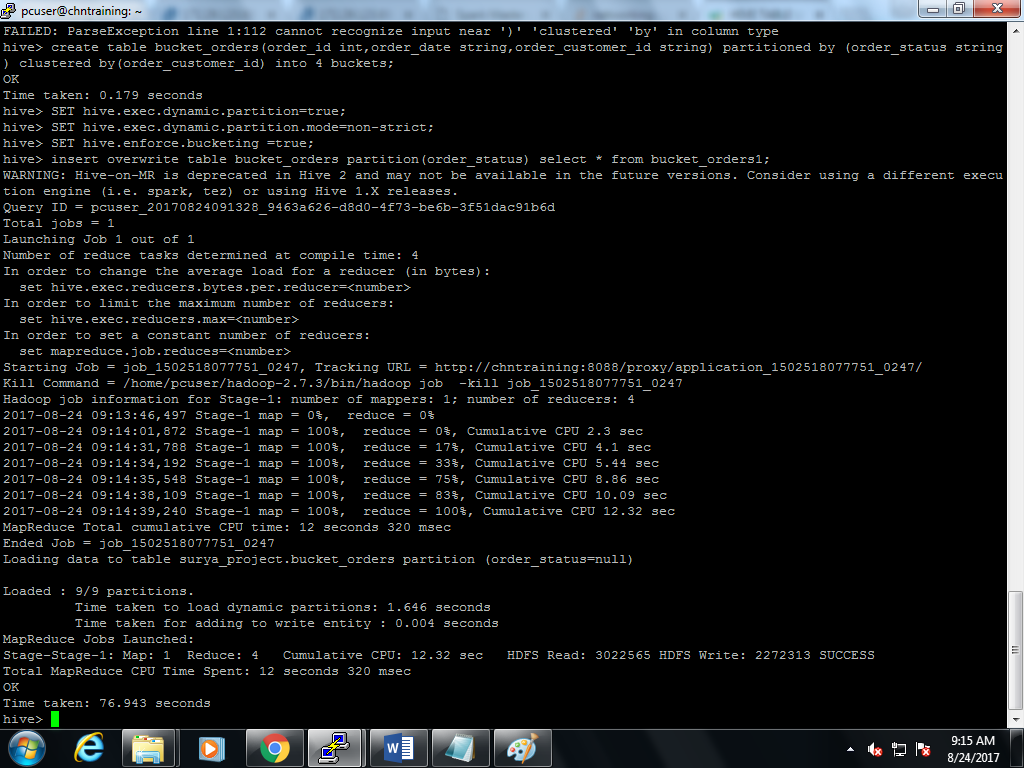
SELECT customer\_fname,customer\_lname, customer\_id,sum(product\_price) as amount\_purchased FROM hnew GROUP BY customer\_fname,customer\_lname, customer\_id order by sum(product\_price) desc limit 5;

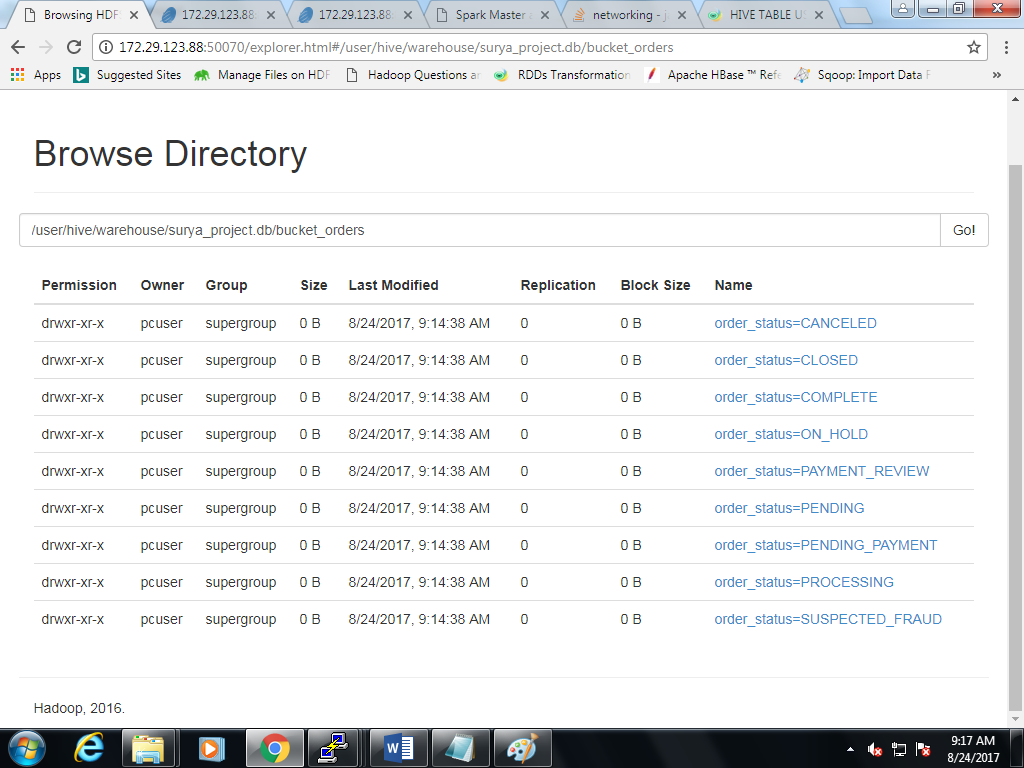
****

**BUCKETING**

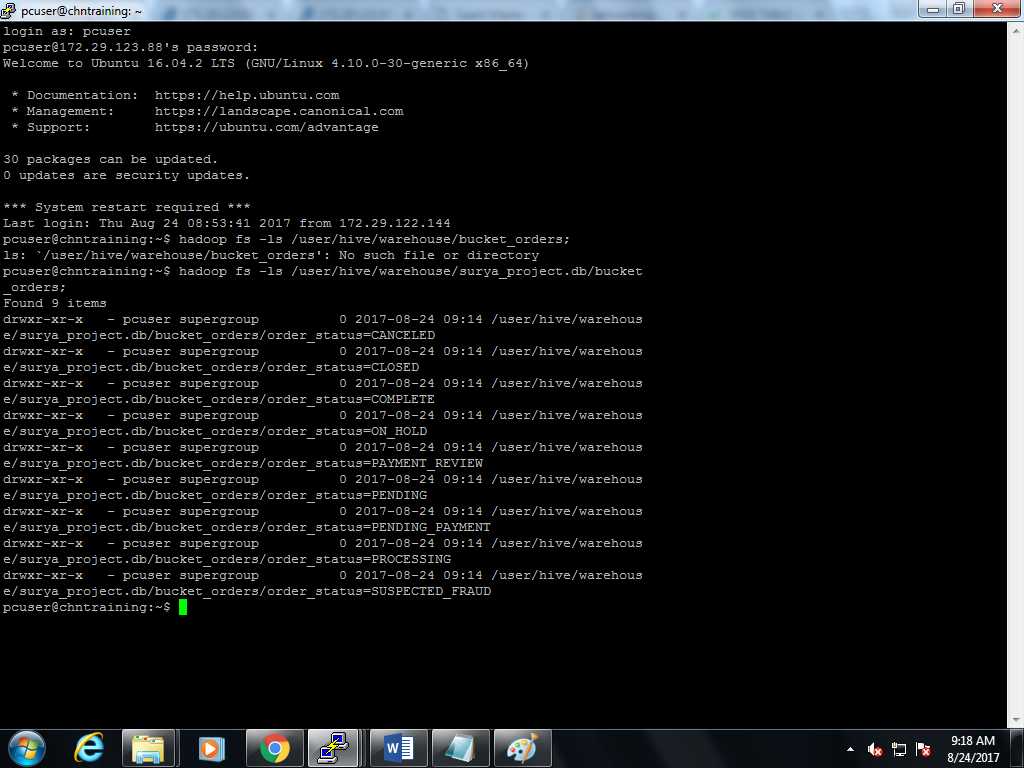
set hive.enforce.bucketing =true;

1. create table bucket\_orders1(order\_id int,order\_date string, order\_customer\_id string,order\_status string) row format delimited fields terminated by ',' stored as textfile;
2. load data local inpath '/home/pcuser/surya/project/orders.csv' into table bucket\_orders1;
3. create table bucket\_orders(order\_id int,order\_date string, order\_customer\_id string) partitioned by (order\_status string) clustered by (order\_customer\_id) into 4 buckets;
4. insert overwrite table bucket\_orders partition(order\_status) select \* from bucket\_orders1;

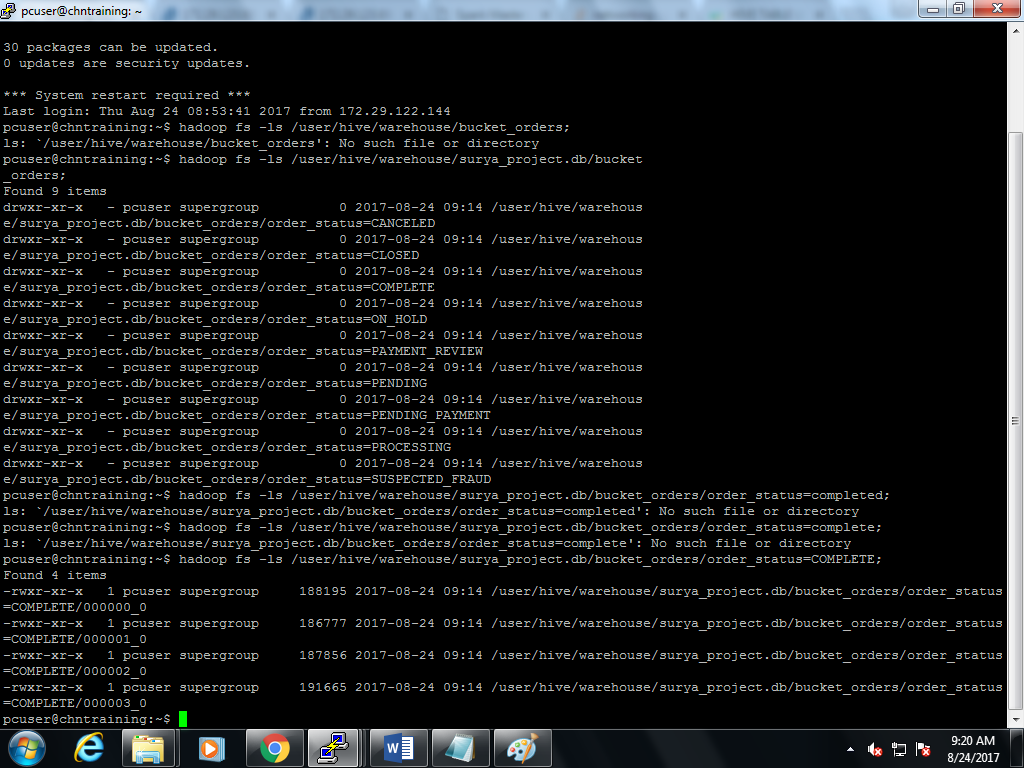
****

****

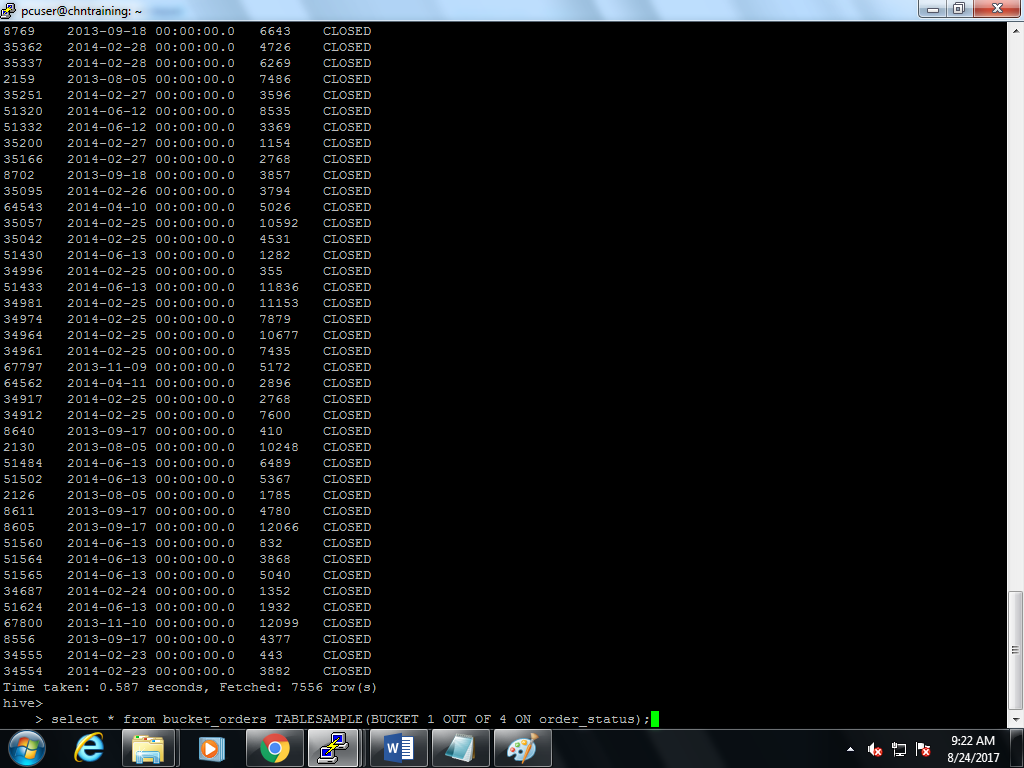
1. hadoop fs -ls /user/hive/warehouse/surya\_project.db/bucket\_orders;

****

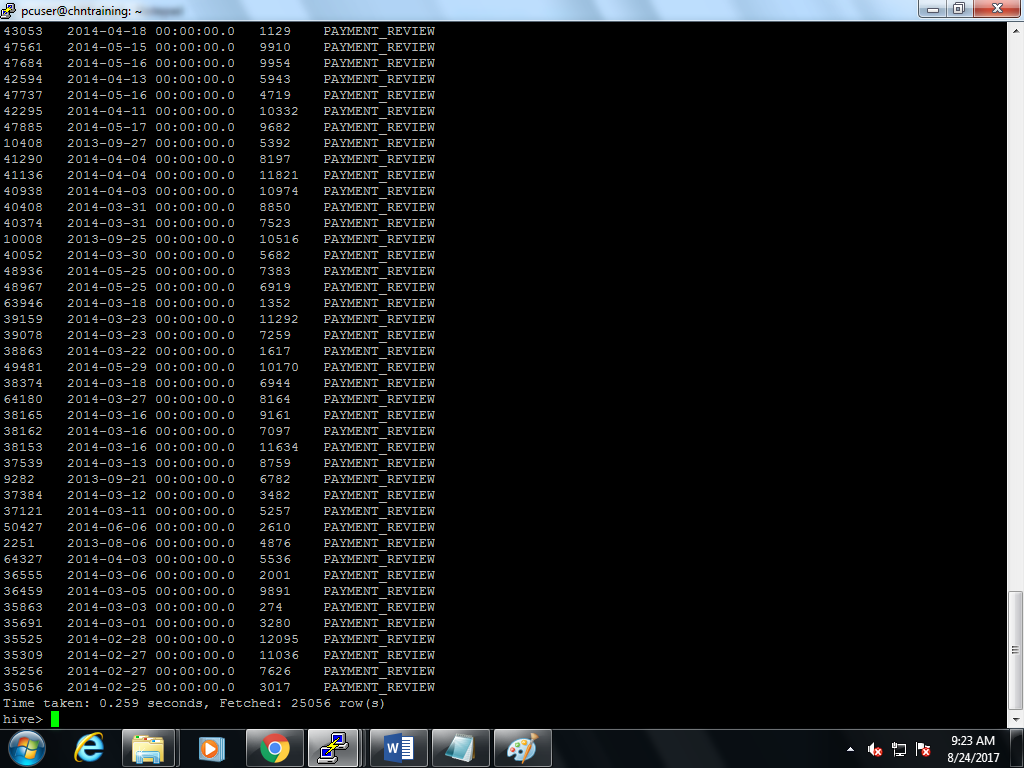
1. hadoop fs –ls /user/hive/warehouse/surya\_project.db/bucket\_orders/order\_status=completed;

****

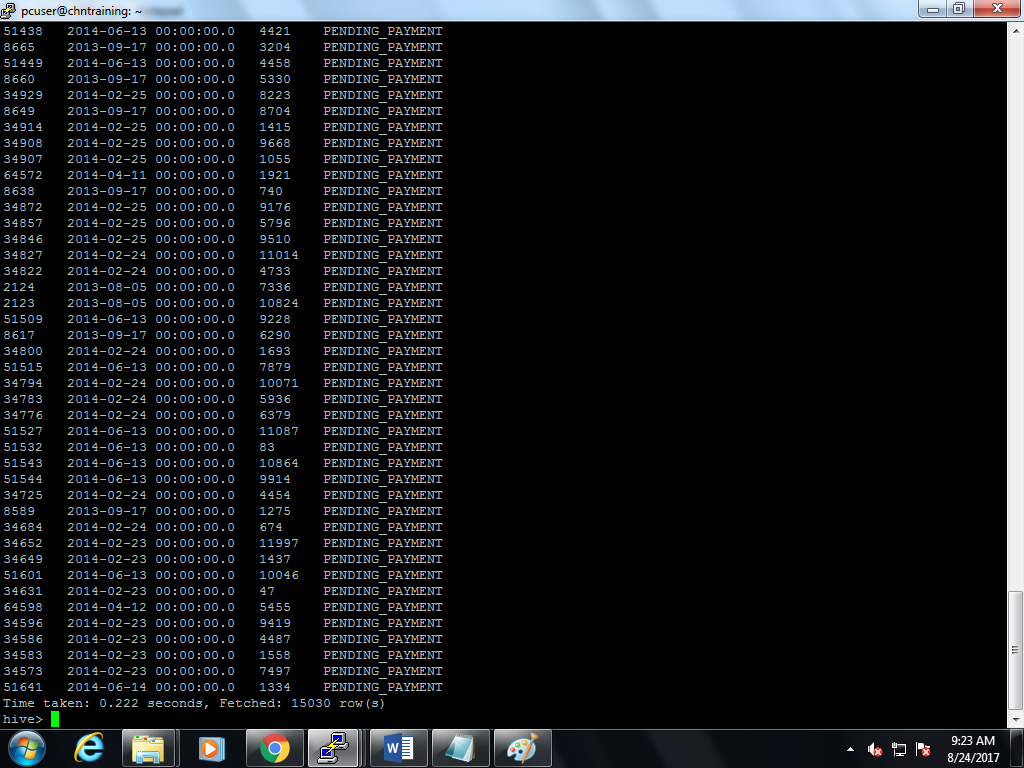
1. select \* from bucket\_orders TABLESAMPLE(BUCKET 1 OUT OF 4 ON order\_status);

****

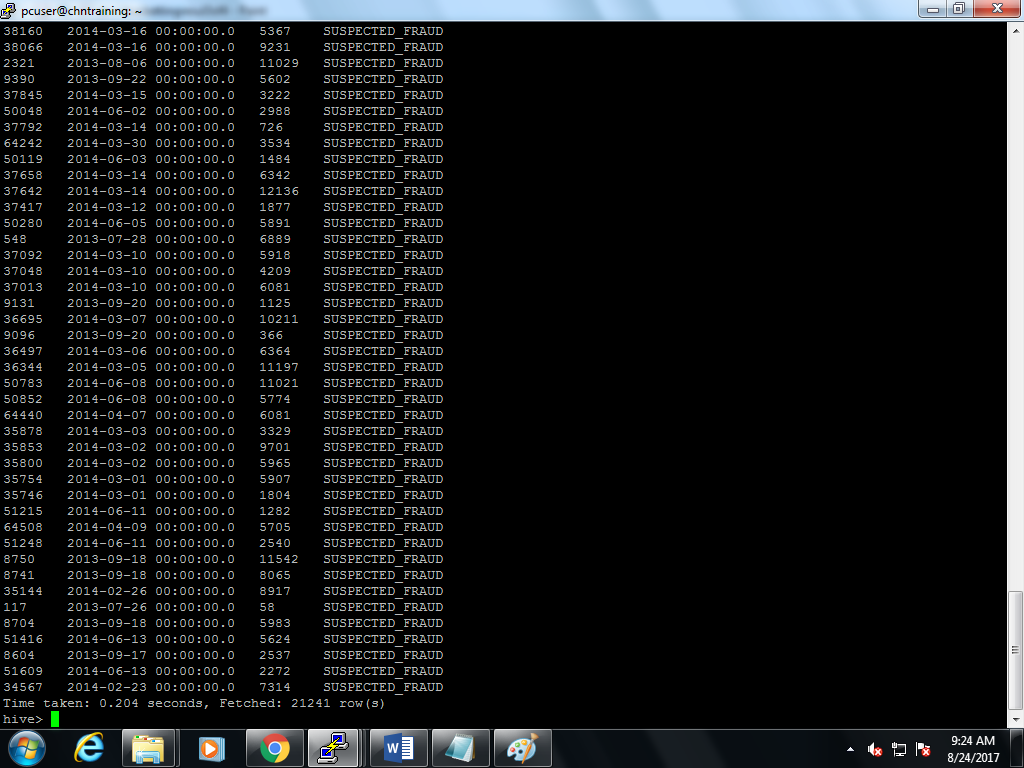
1. select \* from bucket\_orders TABLESAMPLE(BUCKET 2 OUT OF 4 ON order\_status);

****

1. select \* from bucket\_orders TABLESAMPLE(BUCKET 3 OUT OF 4 ON order\_status);

****

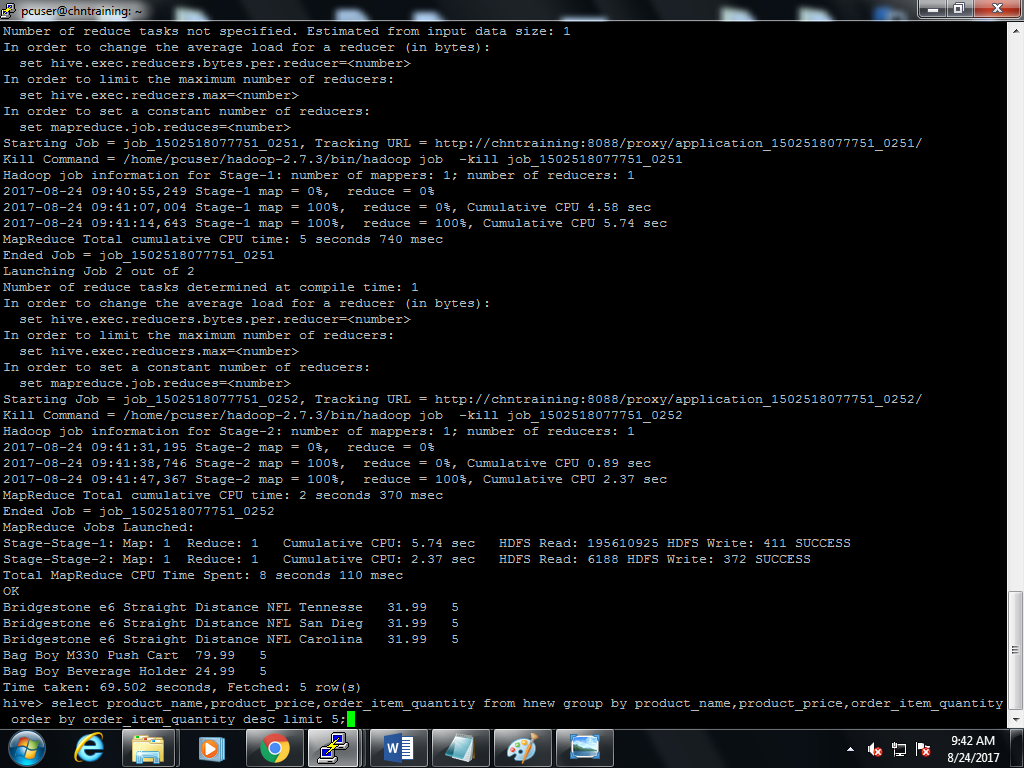
1. select \* from bucket\_orders TABLESAMPLE(BUCKET 4 OUT OF 4 ON order\_status);

****

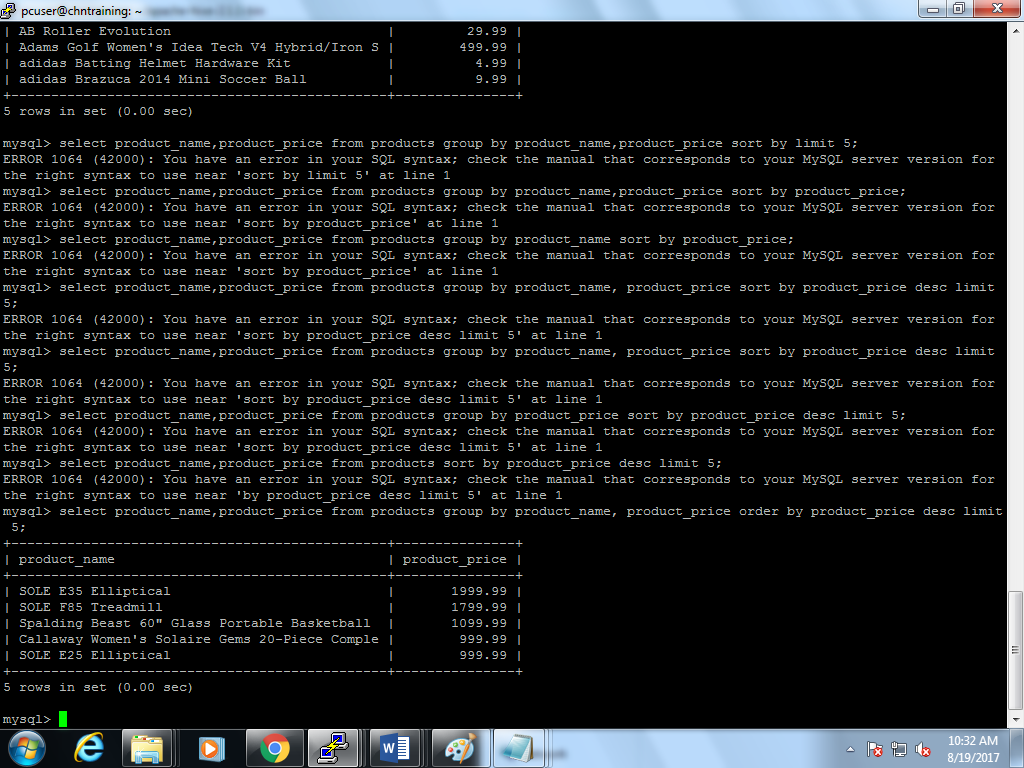
**Top 5 products:**

Select product\_name,product\_price,order\_item\_quantity from hnew group by prouct\_name,product\_price,order\_item\_quantity order by order\_item\_quantity desc limit 5;

**Based on quantity:**

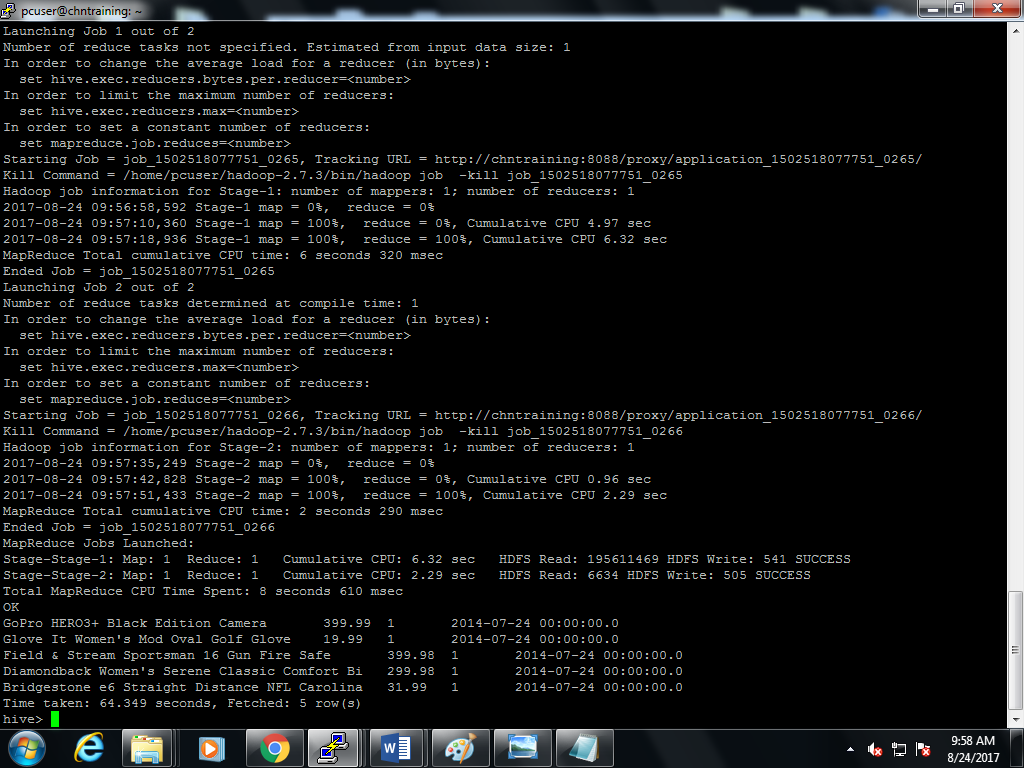
****

**Based on price:**

****

**Top 5 orders based on date:**

**Select product\_name,product\_price,order\_item\_quantity,order\_date from hnew group by product\_name,product\_price,order\_item\_quantity,order\_date order by order\_date desc limit 5;**

****