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

>show databases; //lists the databases.

>use surya\_project; //uses the userdefined database.

----------------------------------------------------------

>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';

----------------------------------------------------

>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';

----------------------------------------------------

>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';

----------------------------------------------------

>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;

-----------------------------------------------------

>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';

-----------------------------------------------------

>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';

-------------------------------------------------------

HIVE

>hive

>show databases;

>use surya\_project;

//after importing tables from sqoop

>select \* from customers;

>select \* from category;

-------------------------------------------------------

SQOOP

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;

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;

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;

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;

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;

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;

-------------------BUCKETING----------------------------

SET hive.exec.dynamic.partition=true;

SET hive.exec.dynamic.partition.mode=non-strict;

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

>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';

--------------------------BUCKETING------------------------

create table bucket\_cus(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) clustered by (customers\_city) into 4 buckets;

insert overwrite table bucket\_cus select \* from customer;

create table customers1(id int,name string,email\_preferences struct<email\_format:string,frequency:string,categories:struct<promos:boolean,surveys:boolean>>4 addresses map<string,struct<street\_1:string,street\_2:string,city:string,state:string,zip\_code:string>> 5 orders array<struct<order\_id:string,order\_date:string,items:array<struct<product\_id:int,sku:string,name:string,price:double,qty:int>>>>);

--------------bucketing-------------------------------------

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;

load data local inpath '/home/pcuser/surya/project/orders.csv' into table bucket\_orders1;

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;

insert overwrite table bucket\_orders partition(order\_status) select \* from bucket\_orders1;

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

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

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

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

------------------------json---------------------------------

CREATE TABLE r1 (jdoc JSON);

create table r1(id int,name string,email\_preferences struct<email\_format:string,frequency:string,categories:struct<promos:boolean,surveys:boolean>>4 addresses map<string,struct<street\_1:string,street\_2:string,city:string,state:string,zip\_code:string>> 5 orders array<struct<order\_id:string,order\_date:string,items:array<struct<product\_id:int,sku:string,name:string,price:double,qty:int>>>>);

--------------------------------------------------------------

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;