1.Simple Assignment(products\_dim)

CREATE TABLE retail\_demo.products\_dim

(

product\_id TEXT,

category\_id TEXT,

price TEXT,

product\_name TEXT

)

Loading the data for all the tables Using Pig:

product = load '/home/hduser/simplepigproject/sample-data/products\_dim.tsv' as (product\_id:int,categoryid:int,price:float,product\_name:chararray);

category = load '/home/hduser/simplepigproject/sample-data/categories\_dim.tsv' as (category\_id:int,category\_name:chararray);

orderitems = load '/home/hduser/simplepigproject/sample-data/order\_lineitems.tsv' as (orderid:int,order\_item\_id:int,productid:int,productname:chararray,customerid:int,storied:int,item\_shipment\_status\_code:chararray,orderdate:datetime,shipdate:datetime,itemreturndate:datetime,itemrefunddate:datetime,product\_category\_id:int,product\_category\_name:chararray,paymentmethod:chararray,tax\_amount:double,item\_quantity:int,item\_price:double,discount\_amount:double,couponcode:chararray,coupon\_amount:double,ship\_address\_line1:chararray,ship\_address\_line2:chararray,ship\_address\_line3:chararray,ship\_address\_city:chararray,ship\_address\_state:chararray,ship\_address\_postal\_code:chararray,ship\_address\_country:chararray,ship\_phone\_number:long,ship\_customer\_name:chararray,ship\_customer\_email\_address:chararray,ordersessionid:chararray,weburl:chararray);

orders = load '/home/hduser/simplepigproject/sample-data/orders.tsv' as (orderid:long,customerid:int,storeid:int,orderdate:chararray,ship\_completion\_date:chararray,returndate:chararray,refunddate:chararray,payment\_method:chararray,total\_tax\_amount:double,total\_paid\_amount:double,total\_item\_quantity:int,total\_discount\_amount:double,coupon\_code:chararray,coupon\_amount:double,order\_canceled\_flag:chararray,has\_returned\_items\_flag:chararray,has\_refunded\_items\_flag:chararray,fraud\_code:chararray,fraud\_resolution\_code:chararray,billing\_address\_line1:chararray,billing\_address\_line2:chararray,billing\_address\_line3:chararray,billing\_address\_city:chararray,billing\_address\_state:chararray,billing\_address\_postal\_code:chararray,billing\_address\_country:chararray,billing\_phone\_number:long,customername:chararray,customer\_email\_address:chararray,ordering\_session\_id:chararray,website\_url:chararray);

customer\_address = load '/home/hduser/simplepigproject/sample-data/customer\_addresses\_dim.tsv' as (customer\_addressid:int,customer\_id:int,valid\_from:datetime,valid\_to:datetime,housenumber:int,street\_name:chararray,apt\_suit\_no:chararray,

city:chararray,state\_code:chararray,zipcode:int,zipcode\_plus:chararray,country:chararray,phonenumber:long);

customers = load '/home/hduser/simplepigproject/sample-data/customers\_dim.tsv' as(customerid:int,firstname:chararray,lastname:chararray,gender:chararray);

1. Get the product between 2 price range(500-1000)

prodpricebtw5kto100k = filter product by price > 500 and price < 1000;

1. Find out all the categories that the stores sell.

fewdetails = foreach orderitems generate product\_category\_id as categoryid,product\_category\_name as categoryname;

distinctdetails = distinct fewdetails;

1. Find out all the categories and the total product on that categories.

categoryId,categoryName,totalProducts,products details

36,sports,20,{(),()}

result = join product by categoryid,category by category\_id;

newproduct = foreach result generate product::product\_id as productid,product::categoryid as categoryid,category::category\_name as categoryname,product::price as pirce,product::product\_name as productname

grpd = group newproduct by (categoryid,categoryname);

categorywiseproduct = foreach grpd generate group.categoryid as categoryid,group.categoryname as categoryname,COUNT(newproduct.productid) as totalProducts,newproduct.productname as productdetails;

1. Find out the product with maximum & minimum price on each categories

result = join product by categoryid,category by category\_id;

newproduct = foreach result generate product::product\_id as productid,product::categoryid as categoryid,category::category\_name as categoryname,product::price as price,product::product\_name as productname;

grpd = group newproduct by (categoryid,categoryname);

categorywisemax\_min = foreach grpd generate group.categoryid as categoryid,group.categoryname as categoryname,MAX(newproduct.price) as maxprice,MIN(newproduct.price) as minprice;

1. Find out all the product between 2 prices on a particular category.

newproduct = foreach result generate product::product\_id as productid,product::categoryid as categoryid,category::category\_name as categoryname,product::price as price,product::product\_name as productname;

grpd = group newproduct by (categoryid,categoryname);

fewproductdetails = foreach grpd {

productrange = filter newproduct by price > 15 and price < 17;

products = foreach productrange generate categoryid,categoryname , productid,price;

generate products;

};

1. Search a product based on a keyword

prdcts = filter product by product\_name matches '.\*music.\*';

1. Average price of each category products

newproduct = foreach result generate product::product\_id as productid,product::categoryid as categoryid,category::category\_name as categoryname,product::price as price,product::product\_name as productname;

grpd = group newproduct by (categoryid,categoryname);

avgprice = foreach grpd generate group.categoryid as categoryid,group.categoryname as categoryname,ROUND(AVG(newproduct.price)) as avg\_price;

2.Based on ORder table

CREATE TABLE retail\_demo.orders\_hawq

(

order\_id TEXT,

customer\_id TEXT,

store\_id TEXT,

order\_datetime TEXT,

ship\_completion\_datetime TEXT,

return\_datetime TEXT,

refund\_datetime TEXT,

payment\_method\_code TEXT,

total\_tax\_amount TEXT,

total\_paid\_amount TEXT,

total\_item\_quantity TEXT,

total\_discount\_amount TEXT,

coupon\_code TEXT,

coupon\_amount TEXT,

order\_canceled\_flag TEXT,

has\_returned\_items\_flag TEXT,

has\_refunded\_items\_flag TEXT,

fraud\_code TEXT,

fraud\_resolution\_code TEXT,

billing\_address\_line1 TEXT,

billing\_address\_line2 TEXT,

billing\_address\_line3 TEXT,

billing\_address\_city TEXT,

billing\_address\_state TEXT,

billing\_address\_postal\_code TEXT,

billing\_address\_country TEXT,

billing\_phone\_number TEXT,

customer\_name TEXT,

customer\_email\_address TEXT,

ordering\_session\_id TEXT,

website\_url TEXT

)

1. find out total orders by each customer.

grpd = group orders by customerid;

totalcount = foreach grpd generate group as customerid,COUNT(orders.orderid);

1. find out all order details by any customer in the given time range.

input customer id,time range

output: all order description

vim /home/hduser/simplepigproject/sample-data/Myparam.param

CUSTOMERID = 23

TIMERANGE = 2010

vim /home/hduser/simplepigproject/sample-data/Myscript.pig

orders = load '/home/hduser/simplepigproject/sample-data/orders.tsv' as (orderid:long,customerid:int,storeid:int,orderdate:chararray,ship\_completion\_date:chararray,returndate:chararray,refunddate:chararray,payment\_method:chararray,total\_tax\_amount:double,total\_paid\_amount:double,total\_item\_quantity:int,total\_discount\_amount:double,coupon\_code:chararray,coupon\_amount:double,order\_canceled\_flag:chararray,has\_returned\_items\_flag:chararray,has\_refunded\_items\_flag:chararray,fraud\_code:chararray,fraud\_resolution\_code:chararray,billing\_address\_line1:chararray,billing\_address\_line2:chararray,billing\_address\_line3:chararray,billing\_address\_city:chararray,billing\_address\_state:chararray,billing\_address\_postal\_code:chararray,billing\_address\_country:chararray,billing\_phone\_number:long,customername:chararray,customer\_email\_address:chararray,ordering\_session\_id:chararray,website\_url:chararray);

neworders = foreach orders generate $0,$1,$2,GetYear(ToDate($3,'yyyy-MM-dd HH:mm:ss')) as orderdate,$4 .. $30;

details = filter neworders by customerid == $CUSTOMERID and orderdate == $TIMERANGE;

orderdesc = foreach details generate orderid,customerid;

dump;

1. find all the total\_paid amount city\_wise in a particular month

neworders = foreach orders generate $0,$1,$2,GetMonth(ToDate($3,'yyyy-MM-dd HH:mm:ss')) as orderdate,$4 .. $30;

grpd = group neworders by (orderdate,billing\_address\_city);

details = foreach grpd generate group.orderdate as ordermonth,group.billing\_address\_city as city,SUM(neworders.total\_paid\_amount) as totalamount;

1. find out the maximum sales happend in any city in last 1 year

details = join orders by customerid ,customer\_address by customer\_id;

fewdetails = foreach details generate orders::orderid as orderid,orders::customerid as customerid,orders::orderdate as orderdate,customer\_address::city as city;

recentdetails = foreach fewdetails generate orderid,customerid, ToDate(orderdate,'yyyy-MM-dd HH:mm:ss') as orderdate,city;

recentdetails1 = foreach recentdetails generate $0,$1,GetYear($2) as year,$3;

grpd = group recentdetails1 by (city,year);

city\_wise\_sales = foreach grpd generate group.city as city,group.year as year ,COUNT(recentdetails1.orderid) as sales\_count;

flt = filter city\_wise\_sales by year >=2009 and year <=2010;

max\_sales = group flt ALL;

max\_sales\_09\_10 = foreach max\_sales generate,MAX(flt.sales\_count) as salescount;

recent = join flt by sales\_count,max\_sales\_09\_10 by salescount;

recent1 = foreach recent generate flt::city as city,flt::year as year,flt::sales\_count as sales;

Hyd which month has highest sales

Banglore which month has highest sales

1. Find out total business done and total tax to be paid till date

grpd = group orders ALL;

details = foreach grpd generate COUNT(orders.orderid) as Nooforders,SUM(orders.total\_paid\_amount) as total,SUM(orders.total\_tax\_amount) as tax;

1. Find out top 20 product in good demand & top 20 product in less demand.

grpd = group orderitems by (productid,productname);

details = foreach grpd generate group.productid as productid,group.productname as productname,COUNT(orderitems.item\_quantity) as quantity;

fewdetails = order details by quantity desc;

lmt = limit fewdetails 20;

fewdetails = order details by quantity asc;

lmt = limit fewdetails 20;

1. Find out all customer with frequent order cancellation

output

c001 8(cancelled) 10(total) order by percentage

c005 2(cancelled) 10(total)

grpd = group orders by customerid;

details = foreach grpd generate group as customerid,COUNT(orders.orderid) as totalorders;

cancelorders = filter orders by order\_canceled\_flag == ‘Y’;

recentdetails = group cancelorders by customerid;

canceldetails = foreach recentdetails generate group as customerid,COUNT(cancelorders.orderid) as cancelcount;

newdetails = join details by customerid,canceldetails by customerid;

ordercancelation = foreach newdetails generate details::customerid as customerid,details::totalorders as total, canceldetails::cancelcount as cancelled,(canceldetails::cancelcount\*100/details::totalorders) as order\_percentage;

1. Find out all the products returned after using order by the total number of returns.

1.Store all the data in pig relation

A = load ' '

1.Find out top 3 customers with highest no of Orders.

grpd = group orders by customerid;

details = foreach grpd generate group as customerid,COUNT(orders.orderid) as nooforders;

orderdetails = order details by nooforders desc;

lmt = limit orderdetails 3;

dump;

2.Find out top 5 customers with highest amount of order.

details = order orders by total\_paid\_amount desc;

lmt = limit details 5;

dump;

3.How many Female staying at "Some Pin Code location " have purchased "Jewelry" Items.

newdetails = join orderitems by customerid,customers by customerid;

details = foreach newdetails generate orderitems::customerid as customerid,customers::gender as gender,orderitems::product\_category\_name as category,orderitems::ship\_address\_postal\_code as zipcode;

fewdetails = filter details by gender == 'F' and category == 'Jewelry';

grpd = group fewdetails by zipcode;

recent = foreach grpd generate group as zipcode,COUNT(fewdetails.customerid) as noofpeople;

4.How many Male customers have purchased "Jewelry" Item.

newdetails = join orderitems by customerid,customers by customerid;

details = foreach newdetails generate orderitems::customerid as customerid,customers::gender as gender,orderitems::product\_category\_name as category,orderitems::ship\_address\_postal\_code as zipcode;

fewdetails = filter details by gender == 'M' and category == 'Jewelry';

grpd = group fewdetails ALL;

recent = foreach grpd generate COUNT(fewdetails.customerid) as noofemp;

5.How many people from zipcode= "80744" has purchased "Video Games".

newdetails = filter orderitems by ship\_address\_postal\_code == '80744' and product\_category\_name == 'Video Games';

grpd = group newdetails ALL;

noofpeople = foreach grpd generate COUNT(newdetails.customerid) as people\_count;

dump;