Project :

Camera:

1.

grunt> camera = LOAD '/home/hduser/Camera.csv' using org.apache.pig.piggybank.storage.CSVExcelStorage(';') AS (model:chararray, release\_date:double, max\_resolution:double, low\_resolution :double, eff\_pixel :double, zoom\_wide :double, zoom\_tele :double, normal\_focus\_range:double, macro\_focus\_range:double, storage\_incl :double, weight\_battery :double, dimensions:double, price:double);

yearWiseCamera = GROUP camera BY release\_date;

cameraCountYearWise = FOREACH yearWiseCamera GENERATE group as year, COUNT (camera) as TotalCameras;

(1994.0,1)

(1995.0,1)

(1996.0,4)

(1997.0,11)

(1998.0,32)

(1999.0,53)

(2000.0,61)

(2001.0,85)

(2002.0,89)

(2003.0,101)

(2004.0,141)

(2005.0,143)

(2006.0,153)

(2007.0,163)

2.

fiveOutput = FILTER camera BY max\_resolution>1600 AND max\_resolution<3000 AND low\_resolution>1600 AND low\_resolution<3000 AND price>1000 ;

six = group fiveOutput all;

seven = FOREACH six GENERATE 'Total\_cameras' as key ,COUNT(fiveOutput);

dump seven;

('Total\_cameras’ ,17)

3.

camera0507\_ALL = FILTER camera BY (price>800.0 AND price<1500.0) AND ((release\_date==2005.0) OR (release\_date==2006.0) OR (release\_date ==2007.0));

D = group camera0507\_ALL all;

E = FOREACH D {

sort = ORDER camera0507\_ALL BY eff\_pixel DESC, max\_resolution DESC, dimensions DESC;

latest =LIMIT sort 1;

GENERATE FLATTEN(latest);

}

dump E;

(Panasonic Lumix DMC-FX100,2007.0,4000.0,3264.0,12.0,28.0,100.0,50.0,5.0,27.0,150.0,54.0,1499.0)

Cars:

1.

cars = LOAD '/home/hduser/cars.csv' using org.apache.pig.piggybank.storage.CSVExcelStorage(';') AS (car:chararray, MPG:double, cylinders:INT,displacement:double, horsepower :double, weight :double, acceleration :double, model :INT, origin:chararray);

cargroup = group cars ALL;

carGroupTotalCount = foreach cargroup generate 'Tot\_count' as key, COUNT(cars);

horsePowerGreater200 = FILTER cars BY (horsepower>200.0);

horsePowerGreaterGroup =group horsePowerGreater200 ALL;

horsePowerCount = foreach horsePowerGreaterGroup generate 'hp\_count' as key, COUNT(horsePowerGreater200);

result = CROSS carGroupTotalCount, horsePowerCount;

count = foreach result GENERATE 'frequency' as key, (DOUBLE) $3/$1;

(frequency,0.024630541871921183)

2.

model75US = FILTER cars BY model==75 AND origin=='US';

model75Japan = FILTER cars BY model==75 AND origin=='Japan';

model75Europe = FILTER cars BY model==75 AND origin=='Europe';

group75US = group model75US ALL;

group75Japan = group model75Japan ALL;

group75Europe = group model75Europe ALL;

group75USCount = foreach group75US generate 'US' as key, COUNT(model75US);

group75JapanCount = foreach group75Japan generate 'Japan' as key, COUNT(model75Japan);

group75EuropeCount = foreach group75Europe generate 'Europe' as key, COUNT(model75Europe);

result = CROSS group75USCount, group75EuropeCount, group75JapanCount;

count = foreach result GENERATE ‘Max’ as key, MAX($1,$3,$5);

3.

Carless2500Cy6 = FILTER cars BY cylinders==6 AND weight<2500.00;

Carless2500Cy4 = FILTER cars BY cylinders==4 AND weight<2500.00;

Carless2500Cy3 = FILTER cars BY cylinders==3 AND weight<2500.00;

groupBy2500Cy6 = GROUP Carless2500Cy6 ALL;

groupBy2500Cy4 = GROUP Carless2500Cy6 ALL

groupBy2500Cy3 = GROUP Carless2500Cy6 ALL

groupBy2500Cy6Count = foreach groupBy2500Cy6 generate $0, COUNT(Carless2500Cy6);

groupBy2500Cy4Count = foreach groupBy2500Cy4 generate $0, COUNT(Carless2500Cy4);

groupBy2500Cy3Count = foreach groupBy2500Cy3 generate $0, COUNT(Carless2500Cy3);

dump groupBy2500Cy6Count;

(all,1)

dump groupBy2500Cy4Count;

(all,143)

dump groupBy2500Cy3Count;

(all,3)