Education

Task 1 : Total count of male/female based on education. :if it bachelor, how many male/female

PIG :

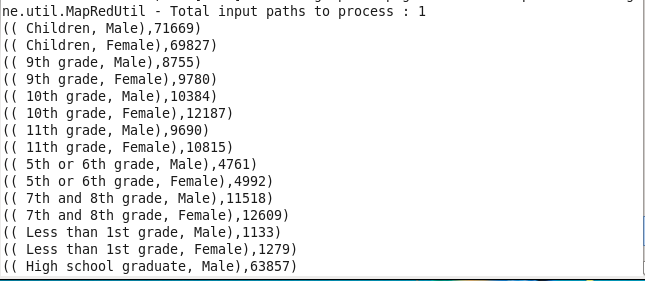
step1 = load '/user/cloudera/Census\_Records.json' using JsonLoader('Age:int,Education:chararray,MartialStatus:chararray,Gender:chararray,TaxFilerStatus:chararray,Income:float,Parents:chararray,CountryOfBirth:chararray,Citizenship:chararray,WeeksWorked:chararray');

step2 = foreach step1 generate $1 as Edu,$3 as Gen;

step3 = group step2 by ($0,$1);

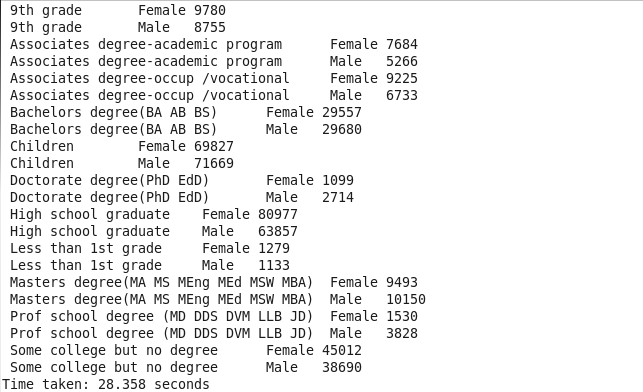
step4 = foreach step3 generate group,COUNT(step2.Gen);

dump step4;



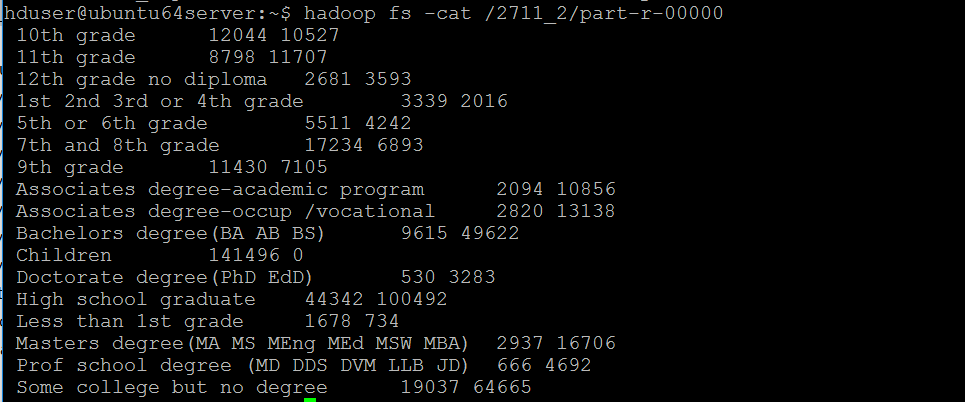
Hive :

select edu,gen, COUNT(\*) Total from final\_census1 group by edu,gen;



Task 2 .Total count of employed/unemployed based on education. : if week work is 0 then unemployed

Advanced Map Reduce:



PIG :

Employed Counts :

step1 = load '/user/cloudera/Census\_Records.json' using JsonLoader('Age:int,Education:chararray,MartialStatus:chararray,Gender:chararray,TaxFilerStatus:chararray,Income:float,Parents:chararray,CountryOfBirth:chararray,Citizenship:chararray,WeeksWorked:int');

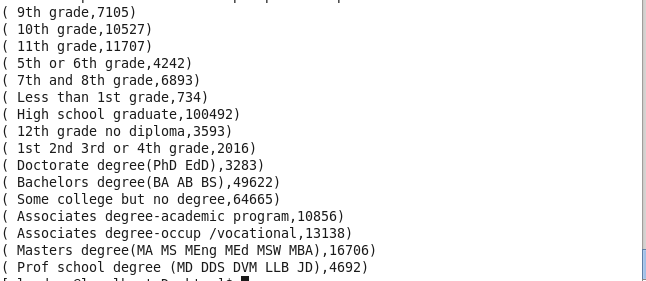
step2 = foreach step1 generate $1 as Edu,$9 as ww;

step3 = filter step2 by $1>0;

step4 = group step3 by $0;

step5 = foreach step4 generate group,COUNT($1);

dump step5;



UnEmployed:

step1 = load '/user/cloudera/Census\_Records.json' using JsonLoader('Age:int,Education:chararray,MartialStatus:chararray,Gender:chararray,TaxFilerStatus:chararray,Income:float,Parents:chararray,CountryOfBirth:chararray,Citizenship:chararray,WeeksWorked:int');

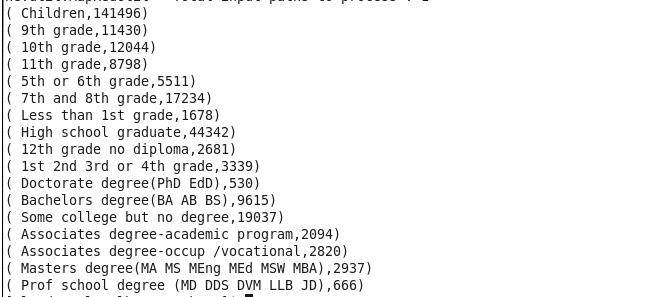
step2 = foreach step1 generate $1 as Edu,$9 as ww;

step3 = filter step2 by $1==0;

step4 = group step3 by $0;

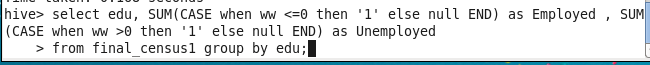
step5 = foreach step4 generate group,COUNT($1);

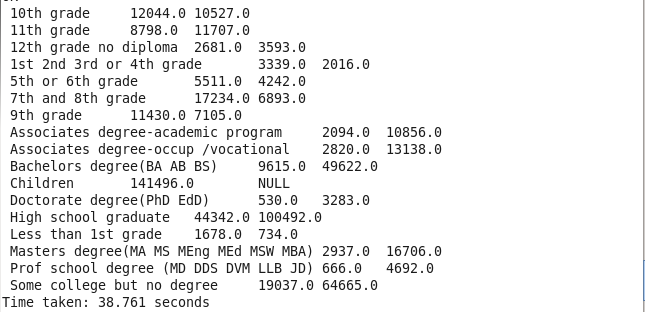
dump step5;



Hive :

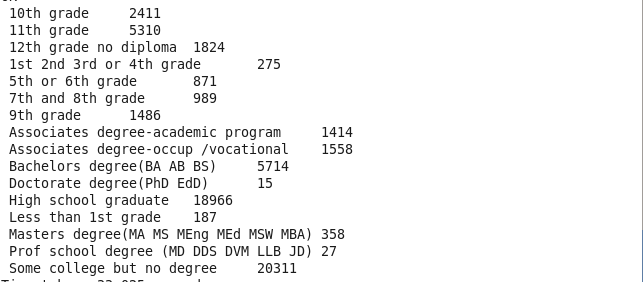
select edu, SUM(CASE when ww <=0 then '1' else null END) as Employed , SUM(CASE when ww >0 then '1' else null END) as Unemployed from final\_census1 group by edu;





3. Total count for people in age range of 18-25 based on education. :





PIG :

a = load '/user/cloudera/Census\_Records.json' using JsonLoader('age:int,edu:chararray,mar:chararray,gen:chararray,tax:chararray,income:chararray,parent:chararray,country:chararray,citizen:chararray,ww:int');

b = foreach a generate age,edu;

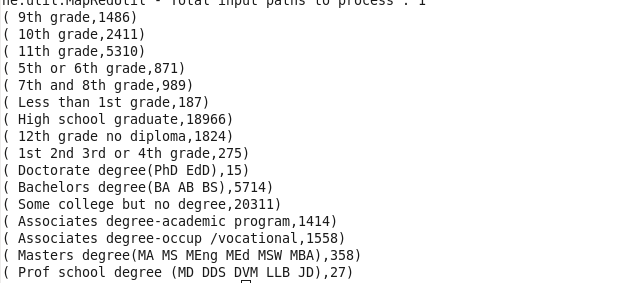
c = filter b by age>17 and age<26;

j = group c by edu;

d = foreach j generate group,COUNT(c.age);

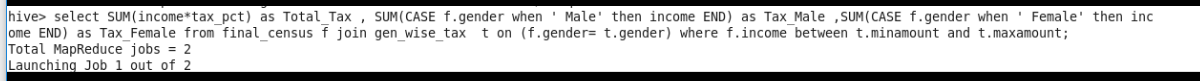
dump d;

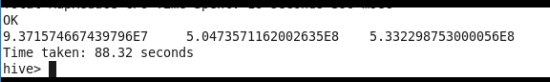
Output:



Finance:

Task 1 : Tax analysis total and gender wise

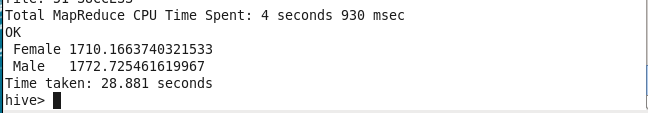




Task 2: Per Capita Income(PCI) analysis consolidated,gender wise and category wise

Genderwise:





PIG:

a = load '/user/cloudera/Census\_Records.json' using JsonLoader('age:int,edu:chararray,mar:chararray,gen:chararray,tax:chararray,income:float,parent:chararray,country:chararray,citizen:chararray,ww:int');

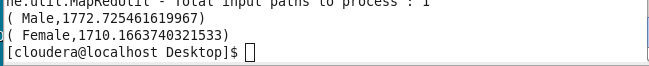
b = foreach a generate gen,income;

c = group b by gen;

d = foreach c generate group,SUM(b.income)/COUNT(b.gen);

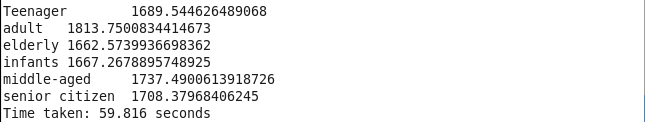
dump d;

Output:



Category wise:





PIG:

a = load '/user/cloudera/Census\_Records.json' using JsonLoader('age:int,edu:chararray,mar:chararray,gen:chararray,tax:chararray,income:float,parent:chararray,country:chararray,citizen:chararray,ww:int');

b = load '/user/cloudera/agegroup1.dat' using PigStorage('\t') as (age:int,cat:chararray);

c = join a by age,b by age;

d = group c by a.age;

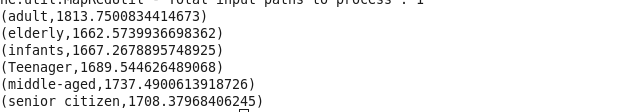
e = foreach c generate $5 as income,$11 as cat;

f = group e by cat;

g = foreach f generate group,SUM(e.income)/COUNT(e.cat);

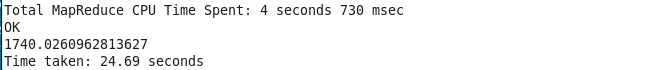
dump g;

Output :



Total CPI:

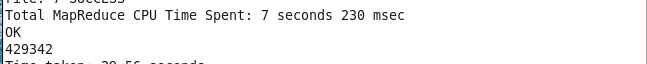




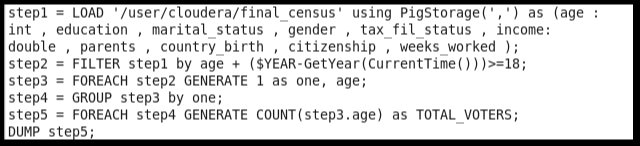
Planning:

1. Voter(s) count in x year(s)





PIG:

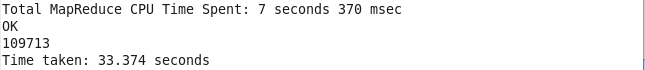




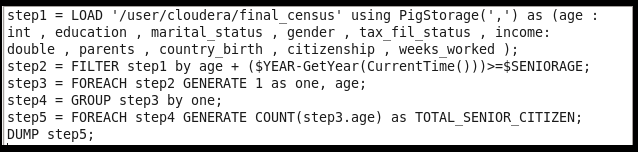


2. Senior Citizen(s) count in x year(s)





PIG:







3. Total number of Male/Female





PIG :

a = load '/user/cloudera/Census\_Records.json' using JsonLoader('age:int,edu:chararray,mar:chararray,gen:chararray,tax:chararray,income:chararray,parent:chararray,country:chararray,citizen:chararray,ww:int');

b = foreach a generate gen;

c = group b by gen;

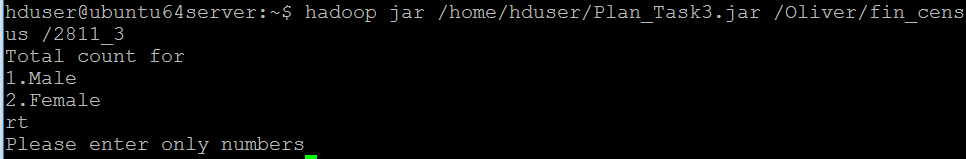
d = foreach c generate group,COUNT(b.gen);

dump d;

Output:



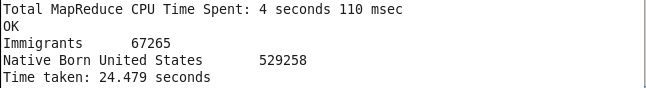
Map Reduce :



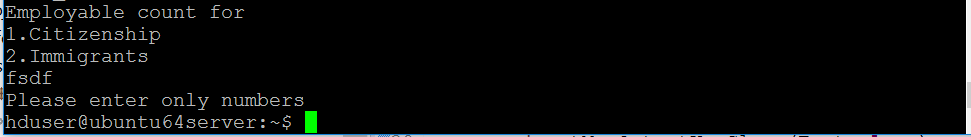


4. Citizens and immigrants count for employed lot

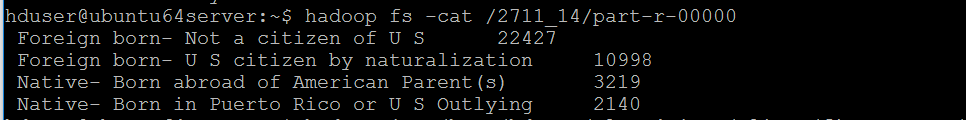




Advanced Map Reduce:



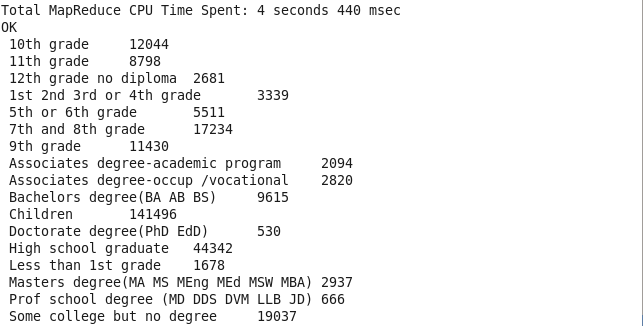




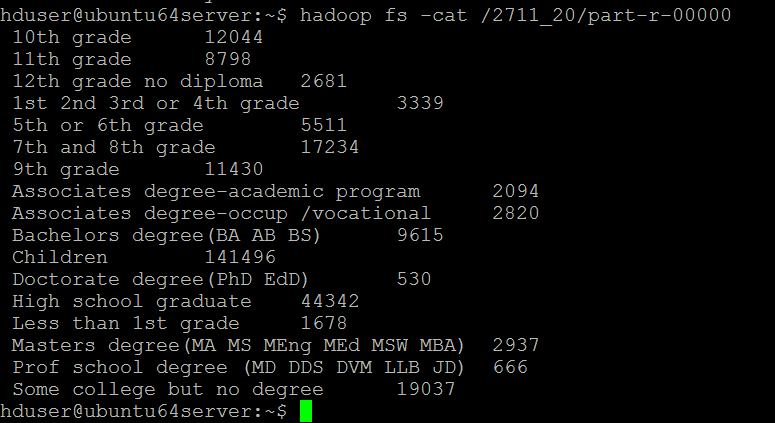
Miscellaneous :

1.Degree wise count for employability





Map Reduce :



PIG :

a = load '/user/cloudera/Census\_Records.json' using JsonLoader('age:int,edu:chararray,mar:chararray,gen:chararray,tax:chararray,income:float,parent:chararray,country:chararray,citizen:chararray,ww:int');

b = foreach a generate $1,$9;

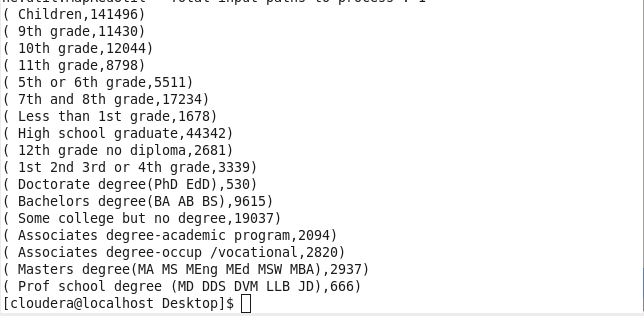
c = filter b by ww==0;

d = group c by $0;

e = foreach d generate group,COUNT(c.$0);

dump e;

OUTPUT:



2. Customer base analysis :

T1.txt :

a = load '/user/cloudera/Census\_Records.json' using JsonLoader('age:int,edu:chararray,mar:chararray,gen:chararray,tax:chararray,income:long,parent:chararray,country:chararray,citizen:chararray,ww:int');

b = foreach a generate age,gen,income,mar;

d = filter b by ((gen==' Female' and mar==' Divorced') and (age>45 and age<60)) ;

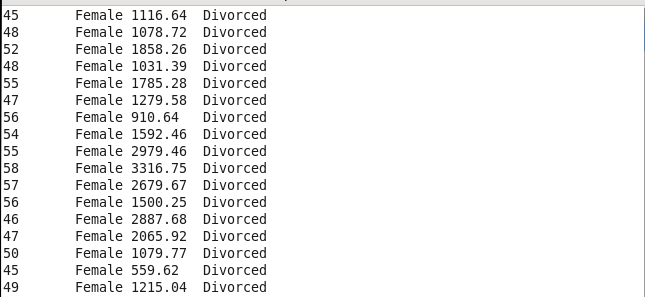
dump d;

Output:



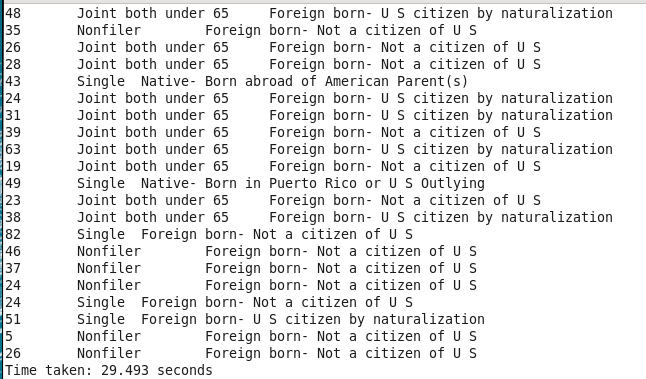
Hive:

select age,gen,income,mar from final\_census1 where gen=' Female' and age between 45 and 60 and mar=' Divorced';

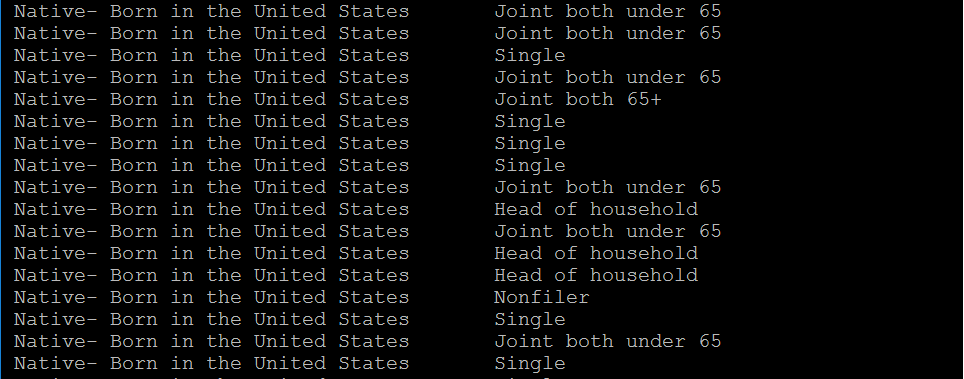


3.Non-US citizen(s) tax filer status





Advanced Map Reduce:



PIG :

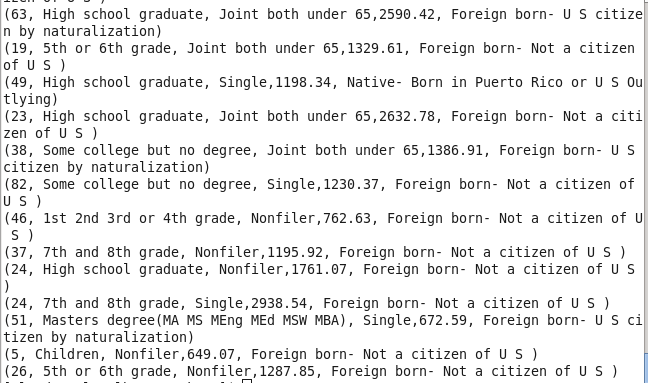
a = load '/user/cloudera/Census\_Records.json' using JsonLoader('age:int,edu:chararray,mar:chararray,gen:chararray,tax:chararray,income:float,parent:chararray,country:chararray,citizen:chararray,ww:int');

b = foreach a generate $0,$1,$4,$5,$8 as citizen;

c = filter b by citizen!=' Native- Born in the United States';

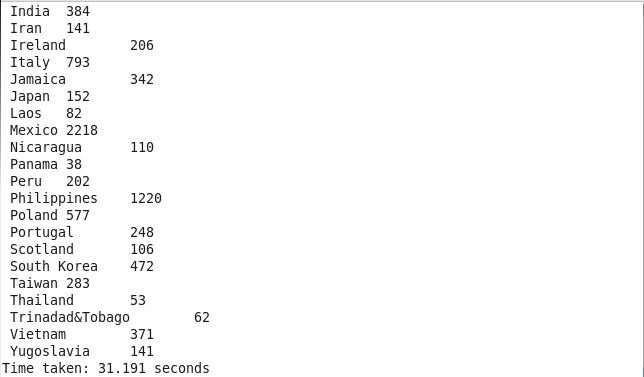
dump c;

OUTPUT:



4. Country of birth wise count for US citizenship by naturalisation





PIG :

a = load '/user/cloudera/Census\_Records.json' using JsonLoader('age:int,edu:chararray,mar:chararray,gen:chararray,tax:chararray,income:float,parent:chararray,country:chararray,citizen:chararray,ww:int');

b = foreach a generate $7,$8;

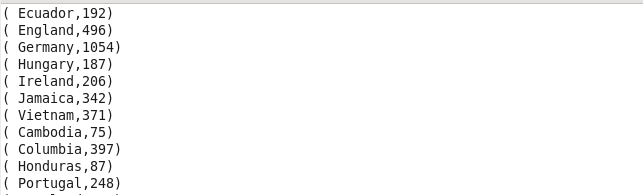
c = filter b by citizen==' Foreign born- U S citizen by naturalization';

d = group c by $0;

e = foreach d generate group,COUNT(c.$0);

dump e;

OUTPUT:



Social:

1.Total amount dispensed on pension in x year(s)





2. Total amount dispensed on scholarship in current year:

PIG:

PIG File: t1.txt

a = load '/user/cloudera/Census\_Records.json' using JsonLoader('Age:int,Education:chararray,MartialStatus:chararray,Gender:chararray,TaxFilerStatus:chararray,Income:float,Parents:chararray,CountryOfBirth:chararray,Citizenship:chararray,WeeksWorked:chararray');

b = load '/user/cloudera/scholar1' using PigStorage(',') as (status:chararray,schamt:int);

c = join a by Parents,b by status;

d = foreach c generate $6 as parent,$11 as Schamt;

e = group d by $0;

f = foreach e generate group,SUM(d.Schamt);

dump f;

Secondary table: scholar1:

Father only present, 2000

Mother only present, 4000

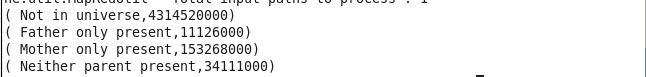
Neither parent present, 7000

Not in universe, 10000

Run:

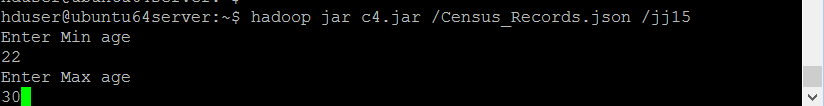


Output:

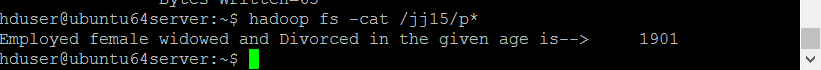


3. For given age range employable female widowed and divorced count – done

Map Reduce



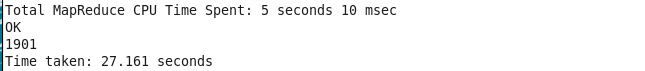
Output :



Hive :

select count(age) from final\_census1 where age between 22 and 30 and gen=' Female' and mar in(' Divorced',' Widowed') and ww>0;

Output:



PIG:

a = load '/user/cloudera/Census\_Records.json' using JsonLoader('age:int,edu:chararray,mar:chararray,gen:chararray,tax:chararray,income:float,parent:chararray,country:chararray,citizen:chararray,ww:int');

b = foreach a generate $0,$2,$3,$9;

c = filter b by ($0>=22 and $0<=30) and ($2==' Female') and ($3>0) and ($1==' Divorced' OR $1==' Widowed');

d = group c by $2;

e = foreach d generate group,COUNT(c.$0);

dump e;

OUTPUT:

