PROJECT ON H1-B VISA

HASIT SYAL

05 MAY 2017

BIG DATA HADOOP

H-1B VISA ANALSYIS

Objective : In this case study, we will be performing analysis on the H1B visa applicants between the years 2011-2015. After analyzing the data, we can derive the following facts.

Abstract: The H1B is an employment-based, non-immigrant visa category for temporary foreign workers in the United States. This is the most common visa status applied for and held by international students once they complete college/ higher education (Masters, Ph.D.) and work in a full-time position.

The dataset has nearly 3 million records !!!

The dataset description is as follow :

CASE_STATUS: Status associated with the last significant event or decision. Valid values include
“Certified”
“Certified-Withdrawn”
“Denied”
“Withdrawn”

EMPLOYER_NAME: Name of employer submitting labor condition application.

SOC_NAME: the Occupational name associated with the SOC_CODE. SOC_CODE is the occupational code ociated with the job being requested for temporary labour condition, as classified by the Standard Occupational Classification (SOC) System.

JOB_TITLE: Title of the job
FULL_TIME_POSITION: Y = Full Time Position; N = Part Time Position

PREVAILING_WAGE: Prevailing Wage for the job being requested for temporary labour condition. The wage is listed at annual scale in USD. The prevailing wage for a job position is defined as the average wage paid to similarly employed workers in the requested occupation in the area of intended employment. The prevailing wage is based on the employer’s minimum requirements for the position.

YEAR: Year in which the H1B visa petition was filed

WORKSITE: City and State information of the foreign worker’s intended area of employment

lon: longitude of the Worksite

lat: latitude of the Worksite

DATA SCRUBBING OR DATA CLEANSING

In the data, few columns are enclosed by double quotes and also we have comma's in a single column and the column is enclosed by double quotes. So we have used hive csv serve to load the data. In the quote Char, we have given **"(**double quote**).** So this will take the column value in between the double quotes

We have create a table to load the h1b applicant's data as shown below :

```
CREATE TABLE h1b_applications( s_no int ,case_status string, employer_name string , soc_name  string ,
job_title string , full_time_position string , prevailing_wage  int ,
year string , worksite string , longitute double ,  latitute double )
ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.OpenCSVSerde'
WITH SERDEPROPERTIES (
"separatorChar" = ",",
"quoteChar" = "\""
)STORED AS TEXTFILE;
```

After loading the file in HIVE table we were still not able to parse the data correctly for Map-Reduce and PIG programs due to the comma’s as a separator , so we created a new table again to replace the separator with “\t”

```
CREATE TABLE h1b_app(s_no int,case_status string ,employer_name  string ,soc_name string ,
job_title string  , full_time_position string  , prevailing_wage  Int , year string ,worksite string ,
longitute  double ,latitute  double)
row format delimited
fields terminated by '\t'
stored as textfile;

INSERT OVERWRITE TABLE h1b_app SELECT regexp_replace(s_no, "\t", ""),
regexp_replace(case_status, "\t", ""), regexp_replace(employer_name,
"\t", ""), regexp_replace(soc_name, "\t", ""),regexp_replace(job_title, "\t", ""),
regexp_replace(full_time_position, "\t", ""),regexp_replace(prevailing_wage, "\t", ""),
 regexp_replace(year, "\t",""), regexp_replace(worksite, "\t", ""), regexp_replace(longitute,
"\t", ""), regexp_replace(latitute, "\t", "") FROM h1b_applicationswhere case_status != "NA";
```

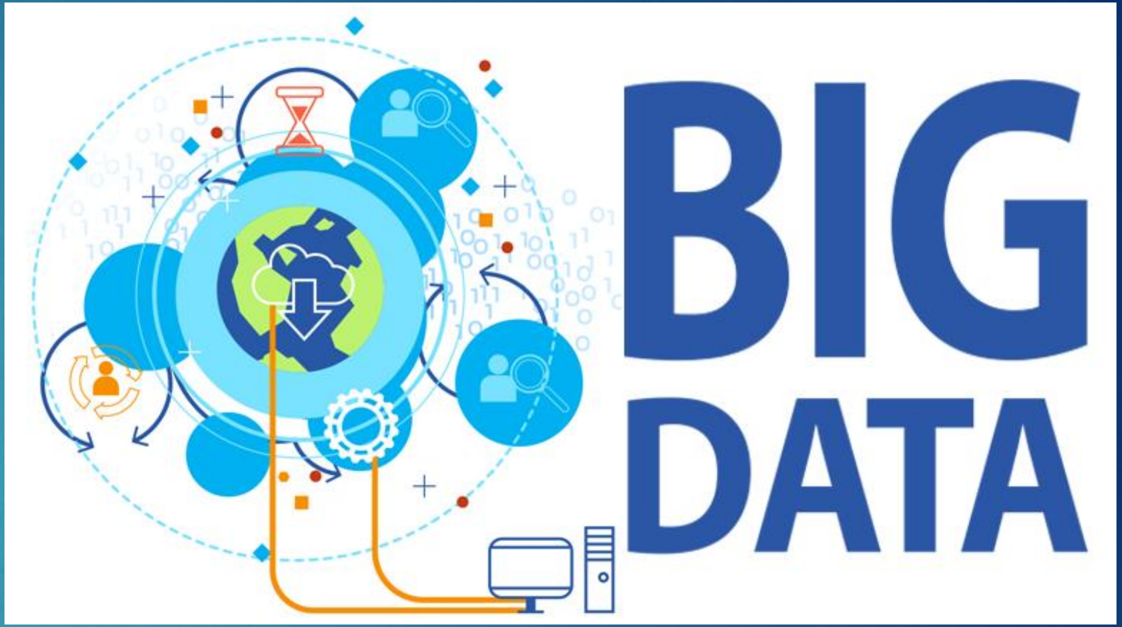
While Executing the use cases we came across an issue again within the h1b_app table ,while running the Query on CASE_STATUS column which consist of four values that includes “CERTIFIED” , “CERTIFIED-WITHDRAWN” , “WITHDRAWN” and “DENIED” along with this we got four more cases that should come under “DENIED” CASE_STATUS which was impacting the over-all result .

So we created a new table , merging up all cases excluding the cases coming under “DENIED” CASE_STATUS.

```
CREATE TABLE h1b_final( s_no int,case_status string, employer_name
string, soc_name string, job_title string, full_time_position
string,prevailing_wage int,year string, worksite string, longitute
double, latitute double )
row format delimited
fields terminated by '\t'
STORED AS TEXTFILE;

INSERT OVERWRITE TABLE h1b_final SELECT s_no,
case when trim(case_status) = "PENDING QUALITY AND COMPLIANCE REVIEW - UNASSIGNED" then "DENIED"
else case_status end,employer_name,
soc_name, job_title,full_time_position,prevailing_wage,
year, worksite, longitute, latitute FROM h1b_final;
```

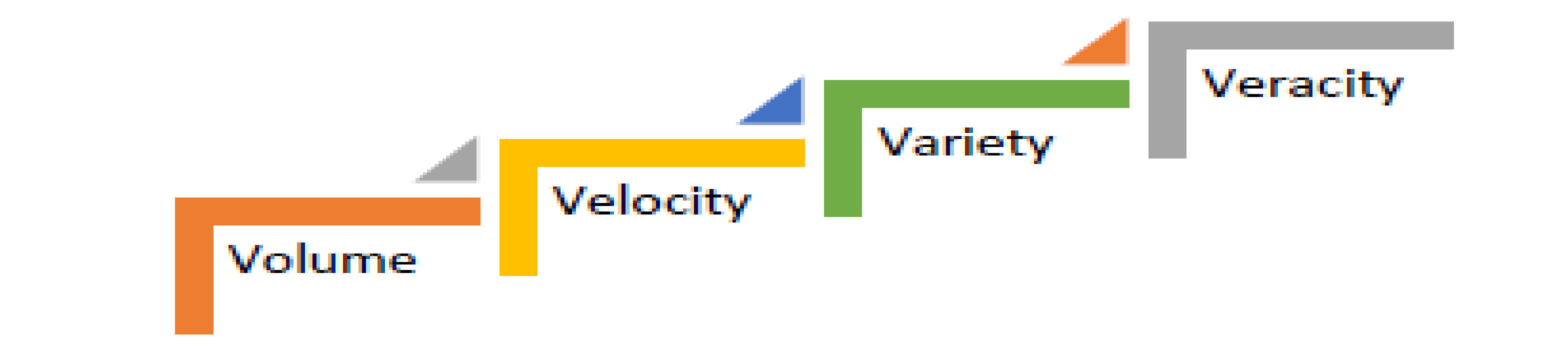
A Brief Introduction about Various Technologies used in our Project

❖ BIG DATA



Big data is a term for data sets that are so large or complex that traditional data processing applications are inadequate to deal with them.

- **4 V's of Big Data :**



HADOOP

- **Apache Hadoop** is an open-source software framework used for distributed storage and processing of very large data sets.

- It consists of computer clusters built from commodity hardware.

- All the modules in Hadoop are designed with a fundamental assumption that hardware failures are a common occurrence and should be automatically handled by the framework.

❖ *HIVE*

- **Apache Hive** is a data warehouse infrastructure built on top of Hadoop for providing data summarization, query, and analysis

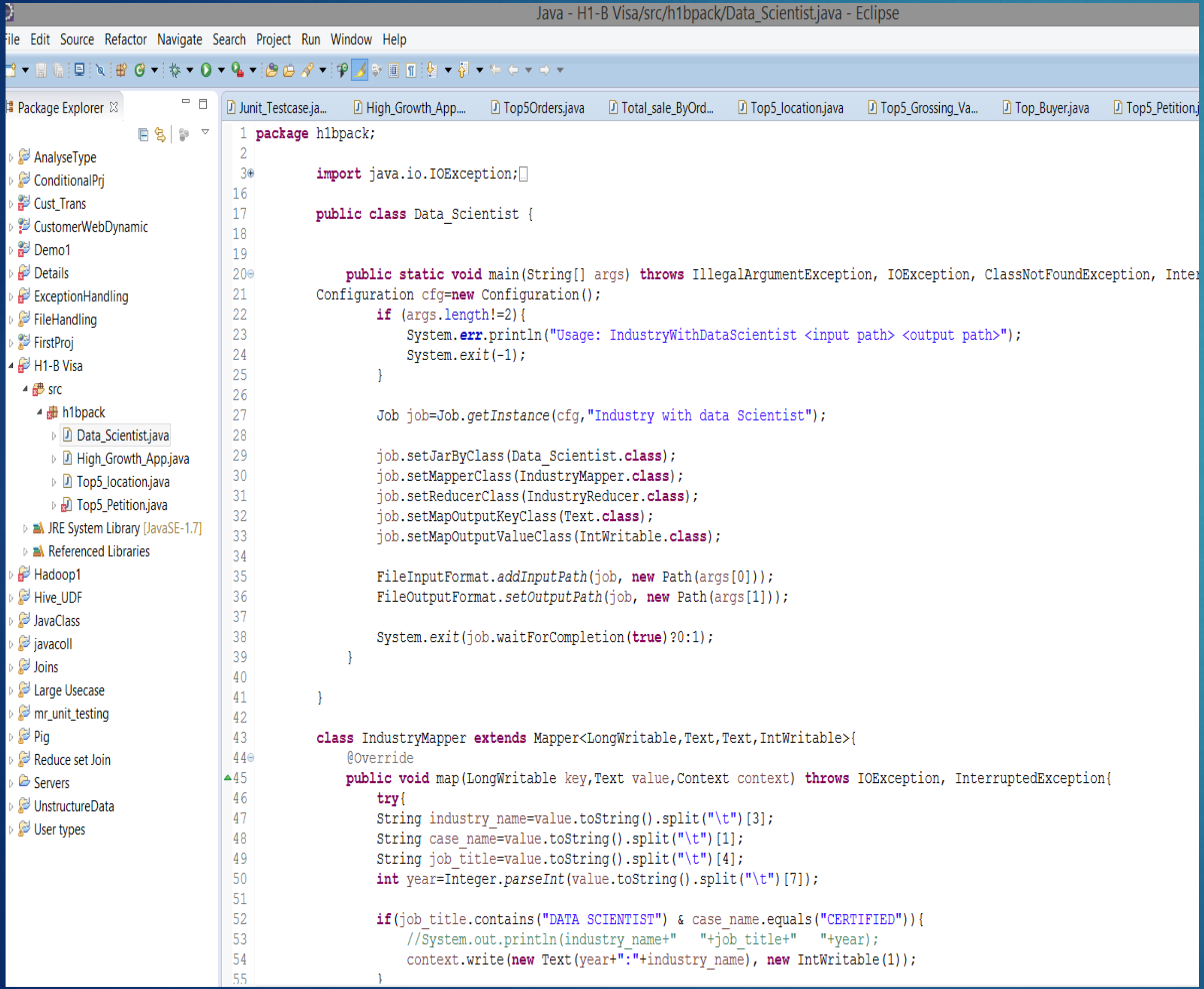
- Hive gives an SQL-like interface to query data stored in various databases and file systems that integrate with Hadoop.

❖ *PIG*

- **Apache Pig** is a high-level platform for creating programs that run on Apache Hadoop. The language for this platform is called **Pig Latin**.
- Pig can execute its Hadoop jobs in Map Reduce, Apache Tez, or Apache Spark. Pig Latin abstracts the programming from the Java Map Reduce idiom into a notation which makes Map Reduce programming high level, similar to that of SQL for RDBMSs
- Pig Latin can be extended using User Defined Functions (UDFs) which the user can write in Java, Python, JavaScript, Ruby or Groovy[2] and then call directly from the language.

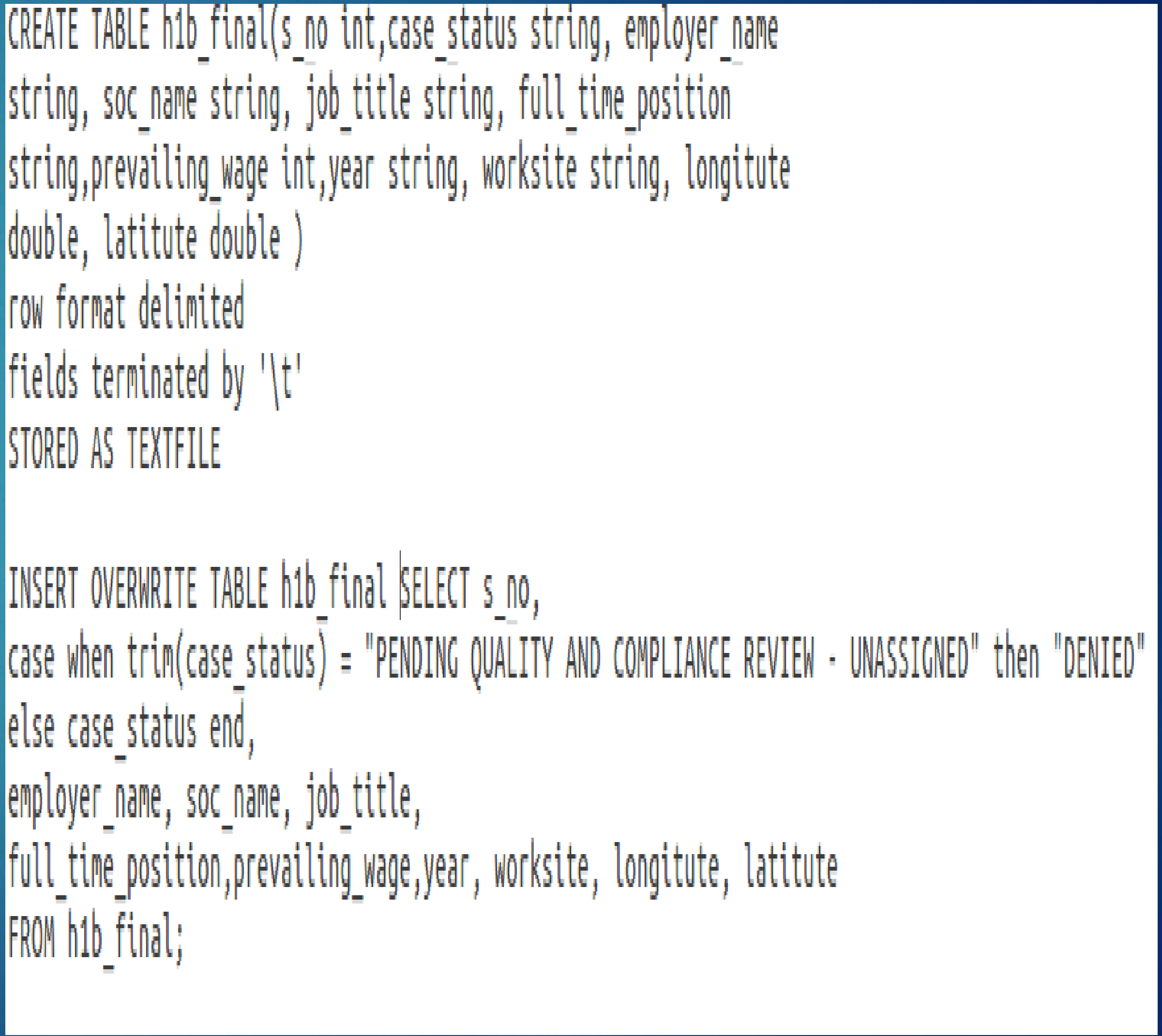
COMMAND LINE INTERFACES

MAP REDUCE



```java
package h1bpack;

        import java.io.IOException;

        public class Data_Scientist {


            public static void main(String[] args) throws IllegalArgumentException, IOException, ClassNotFoundException, Inter
        Configuration cfg=new Configuration();
                if (args.length!=2){
                    System.err.println("Usage: IndustryWithDataScientist <input path> <output path>");
                    System.exit(-1);
                }

                Job job=Job.getInstance(cfg,"Industry with data Scientist");

                job.setJarByClass(Data_Scientist.class);
                job.setMapperClass(IndustryMapper.class);
                job.setReducerClass(IndustryReducer.class);
                job.setMapOutputKeyClass(Text.class);
                job.setMapOutputValueClass(IntWritable.class);

                FileInputFormat.addInputPath(job, new Path(args[0]));
                FileOutputFormat.setOutputPath(job, new Path(args[1]));

                System.exit(job.waitForCompletion(true)?0:1);
            }

        }

        class IndustryMapper extends Mapper<LongWritable,Text,Text,IntWritable>{
            @Override
            public void map(LongWritable key,Text value,Context context) throws IOException, InterruptedException{
                try{
                String industry_name=value.toString().split("\t")[3];
                String case_name=value.toString().split("\t")[1];
                String job_title=value.toString().split("\t")[4];
                int year=Integer.parseInt(value.toString().split("\t")[7]);

                if(job_title.contains("DATA SCIENTIST") & case_name.equals("CERTIFIED")){
                    //System.out.println(industry_name+"   "+job_title+"   "+year);
                    context.write(new Text(year+":"+industry_name), new IntWritable(1));
                }
```

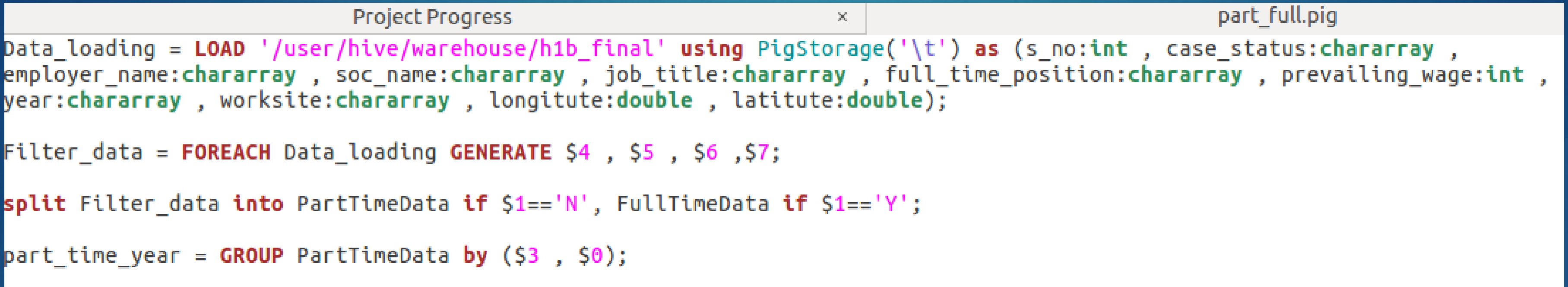
HIVE



```sql
CREATE TABLE h1b_final(s_no int,case_status string, employer_name
string, soc_name string, job_title string, full_time_position
string,prevailing_wage int,year string, worksite string, longitute
double, latitute double )
row format delimited
fields terminated by '\t'
STORED AS TEXTFILE


INSERT OVERWRITE TABLE h1b_final SELECT s_no,
case when trim(case_status) = "PENDING QUALITY AND COMPLIANCE REVIEW - UNASSIGNED" then "DENIED"
else case_status end,
employer_name, soc_name, job_title,
full_time_position,prevailing_wage,year, worksite, longitute, latitute
FROM h1b_final;
```

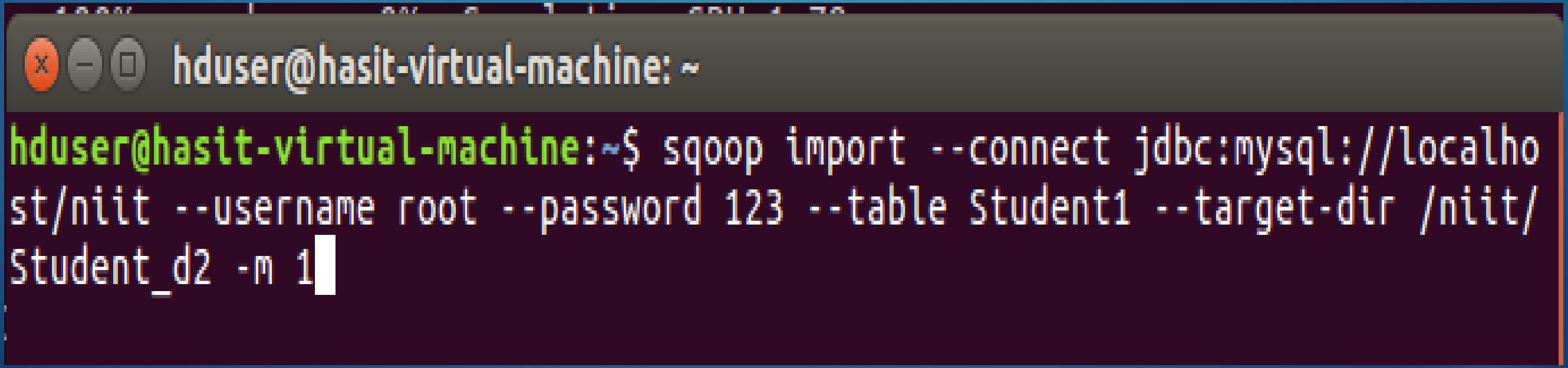
COMMAND LINE INTERFACES

PIG



```
Data_loading = LOAD '/user/hive/warehouse/h1b_final' using PigStorage('\t') as (s_no:int , case_status:chararray ,
employer_name:chararray , soc_name:chararray , job_title:chararray , full_time_position:chararray , prevailing_wage:int ,
year:chararray , worksite:chararray , longitute:double , latitute:double);

Filter_data = FOREACH Data_loading GENERATE $4 , $5 , $6 ,$7;

split Filter_data into PartTimeData if $1=='N', FullTimeData if $1=='Y';

part_time_year = GROUP PartTimeData by ($3 , $0);
```

SQOOP



```
hduser@hasit-virtual-machine:~$ sqoop import --connect jdbc:mysql://localho
st/niit --username root --password 123 --table Student1 --target-dir /niit/
Student_d2 -m 1
```

USE - CASES GENERATION

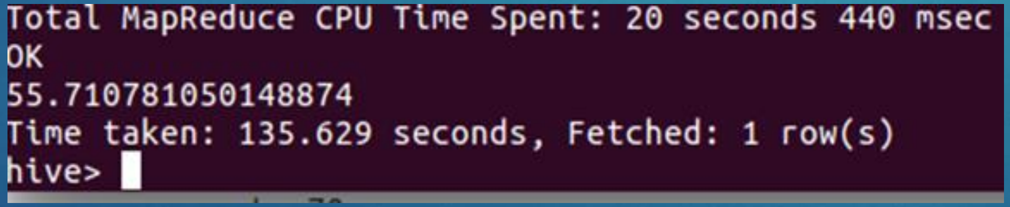
❑ Use Case 1:

Is the number of petitions with Data Engineer job title increasing over time

- Table Used: h1b_final Data
- Description: we will find the number of applicants applied for the various kind of data engineer positions.
- Tool Used: HIVE Technique
- HIVE Query :

```
select year , lag(count(*)) over(order by year),count(*) from h1b_final where job_title like'DATA ENGINEER%' group by year;

select (curr_year-prev_year)/prev_year*100 from (select year , lag(count(*)) over(order by year) prev_year , count(*) curr_year from
h1b_final where job_title like 'DATA ENGINEER%' group by year) p;

select avg ((curr_year-prev_year)/prev_year*100) growth from (select year,coalesce(lag(count(*)) over(order by year),0)
prev_year,count(*) curr_year from h1b_final where job_title like'DATA ENGINEER%' group by year) p;
```

- Screenshot:

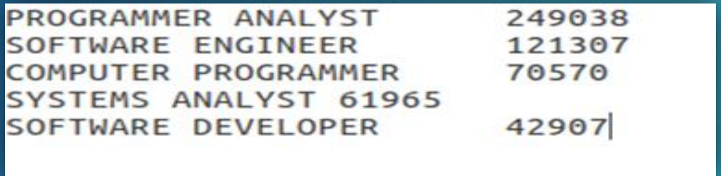


```
Total MapReduce CPU Time Spent: 20 seconds 440 msec
OK
55.710781050148874
Time taken: 135.629 seconds, Fetched: 1 row(s)
hive>
```

❑ USE CASE 2

Find top 5 job titles who are having highest growth in applications.

- Input File – h1b_final Data
- Description : To obtain the Highest Growth in applications for job titles
- Tool used : Map-Reduce
- Key: job title   Value: 1
- Output Path : `hadoop jar  h1b_jars/Job_Titles.jar /user/hive/warehouse/h1b_final /h1b_project/5Job_Titles ;`
- Screenshot :



```
PROGRAMMER ANALYST          249038
SOFTWARE ENGINEER           121307
COMPUTER PROGRAMMER         70570
SYSTEMS ANALYST 61965
SOFTWARE DEVELOPER          42907
```

- Why this report : To find the max growth for 5 job titles

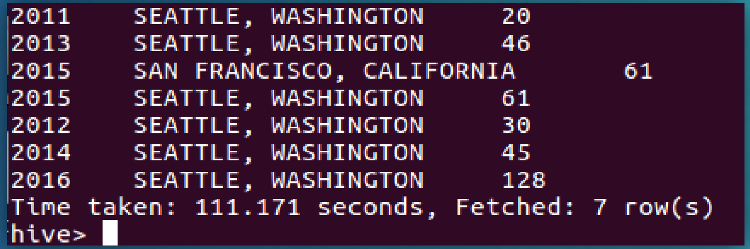
❑ USE CASE 3

Which part of the US has the most Data Engineer jobs for each year

- Table Used : h1b_final Data
- Description: we will get the count of the number of data engineer petitions as per the state.
- Tool used : HIVE
- HIVE Query:

```
create table yearsite as select year , worksite,count(*) year_wise_count  from h1b_final where job_title like '%DATA
ENGINEER%'group by year,worksite;

select * from yearsite a where year_wise_count in (select max(year_wise_count)   from yearsite b where b.year=a.year);
```

- Screenshot :



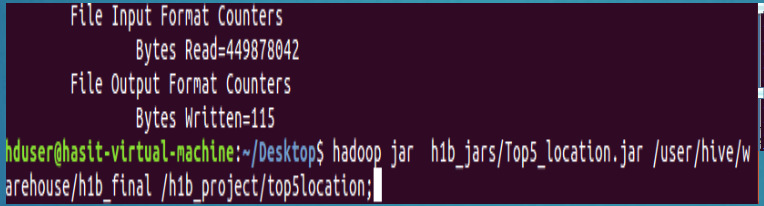
```
2011	SEATTLE, WASHINGTON	20
2013	SEATTLE, WASHINGTON	46
2015	SAN FRANCISCO, CALIFORNIA	61
2015	SEATTLE, WASHINGTON	61
2012	SEATTLE, WASHINGTON	30
2014	SEATTLE, WASHINGTON	45
2016	SEATTLE, WASHINGTON	128
Time taken: 111.171 seconds, Fetched: 7 row(s)
hive>
```

- Why this Report : For an effective analyses

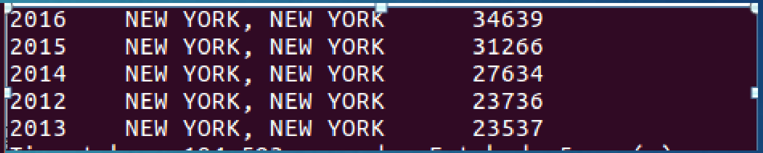
❑ USE CASE 4

Find top 5 locations in the US who have got certified visa for each year

- Input File – h1b_final Data
- Description : To find out Top 5 location received Certified Visa for each year
- Tool used : Map-Reduce
- Key: Year , Location Value: Count
- Output path : `hadoop jar h1b_jars/Top5_location.jar /user/hive/warehouse/h1b_final /h1b_project/top5location;`
- Hive : `select year , worksite ,COUNT(*)as location from h1b_app where case_status ='CERTIFIED' group by year, worksite order by location desc limit 5;`
- Screenshot MR :

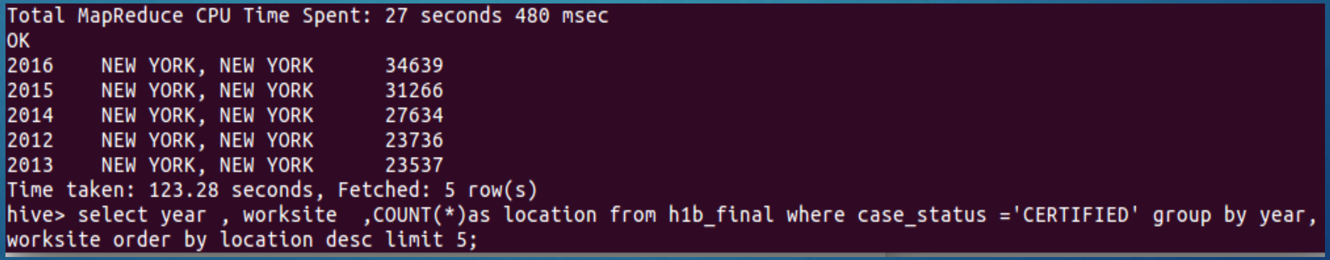


```
        File Input Format Counters
                Bytes Read=449878042
        File Output Format Counters
                Bytes Written=115
hduser@hasit-virtual-machine:~/Desktop$ hadoop jar  h1b_jars/Top5_location.jar /user/hive/w
arehouse/h1b_final /h1b_project/top5location;
```



```
2016    NEW YORK, NEW YORK      34639
2015    NEW YORK, NEW YORK      31266
2014    NEW YORK, NEW YORK      27634
2012    NEW YORK, NEW YORK      23736
2013    NEW YORK, NEW YORK      23537
```

- Screenshot HIVE :



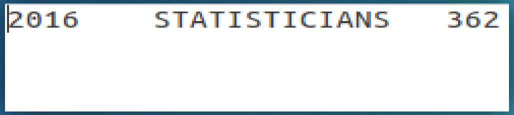
```
Total MapReduce CPU Time Spent: 27 seconds 480 msec
OK
2016    NEW YORK, NEW YORK      34639
2015    NEW YORK, NEW YORK      31266
2014    NEW YORK, NEW YORK      27634
2012    NEW YORK, NEW YORK      23736
2013    NEW YORK, NEW YORK      23537
Time taken: 123.28 seconds, Fetched: 5 row(s)
hive> select year , worksite  ,COUNT(*)as location from h1b_final where case_status ='CERTIFIED' group by year,
worksite order by location desc limit 5;
```

- Why this Report : Analyze the Worksite column for Certified visa

USE CASE 5

Which industry has the most number of Data Scientist positions

- Input File – h1b_final Data
- Description : Number of petitions each industry received for data scientist position ordered by the count in descending order.
- Tool used : Map-Reduce
- Key: Year , Soc_name   Value: 1
- Output path : `hadoop jar  h1b_jars/Industry_data_scientist.jar /user/hive/warehouse/h1b_final /h1b_project/data_scientist`
- Screenshot :



```
2016      STATISTICIANS    362
```

- Why this Report : To obtain a summarized view for Data scientist position

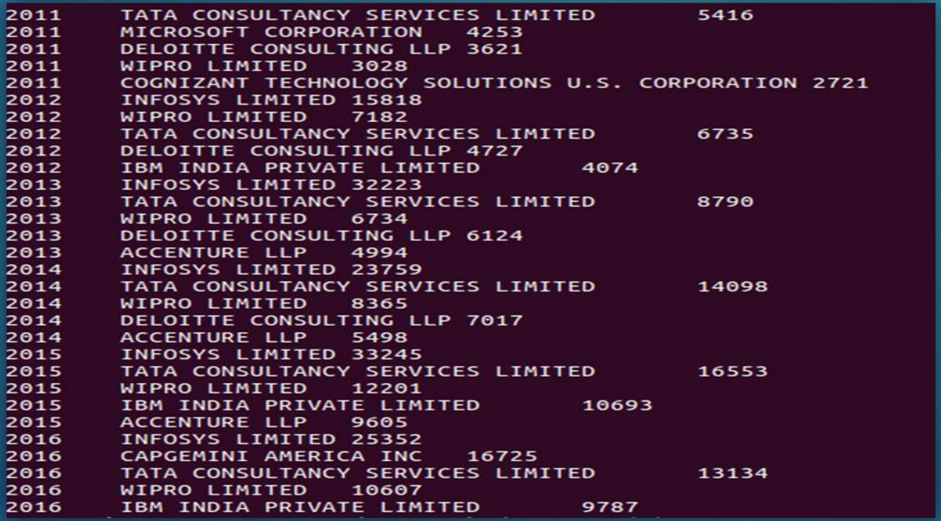
❑ USE CASE 6

Which top 5 employers file the most petitions each year

- Table Used : emp_top
- Description: This case select the employer_name and the number of petitions it has received each year.
- Tool used : HIVE
- HIVE Query :

```
create table emp_top as select COUNT(*) year , year_count ,  employer_name from emp_top group  by year_count;

select year,employer_name , year_count from (select rank() over(partition by year order by year_count desc) nrow,* from emp_top) where nrow<6;
```

- Screenshot :



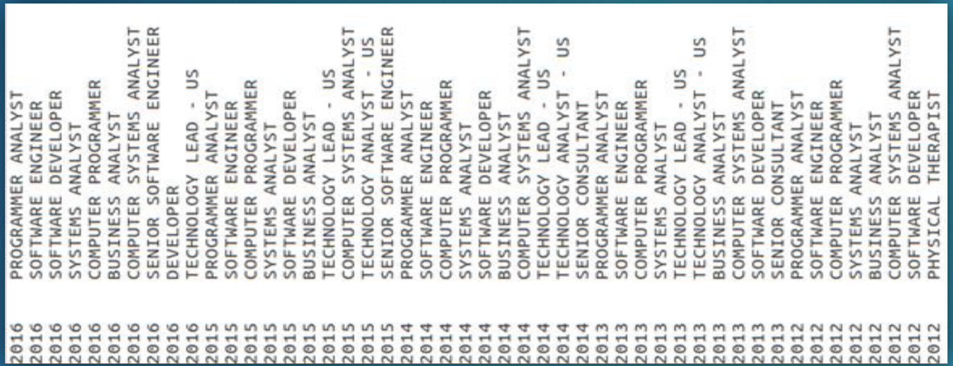
```
2011	TATA CONSULTANCY SERVICES LIMITED	5416
2011	MICROSOFT CORPORATION	4253
2011	DELOITTE CONSULTING LLP	3621
2011	WIPRO LIMITED	3028
2011	COGNIZANT TECHNOLOGY SOLUTIONS U.S. CORPORATION	2721
2012	INFOSYS LIMITED	15818
2012	WIPRO LIMITED	7182
2012	TATA CONSULTANCY SERVICES LIMITED	6735
2012	DELOITTE CONSULTING LLP	4727
2012	IBM INDIA PRIVATE LIMITED	4074
2013	INFOSYS LIMITED	32223
2013	TATA CONSULTANCY SERVICES LIMITED	8790
2013	WIPRO LIMITED	6734
2013	DELOITTE CONSULTING LLP	6124
2013	ACCENTURE LLP	4994
2014	INFOSYS LIMITED	23759
2014	TATA CONSULTANCY SERVICES LIMITED	14098
2014	WIPRO LIMITED	8365
2014	DELOITTE CONSULTING LLP	7017
2014	ACCENTURE LLP	5498
2015	INFOSYS LIMITED	33245
2015	TATA CONSULTANCY SERVICES LIMITED	16553
2015	WIPRO LIMITED	12201
2015	IBM INDIA PRIVATE LIMITED	10693
2015	ACCENTURE LLP	9605
2016	INFOSYS LIMITED	25352
2016	CAPGEMINI AMERICA INC	16725
2016	TATA CONSULTANCY SERVICES LIMITED	13134
2016	WIPRO LIMITED	10607
2016	IBM INDIA PRIVATE LIMITED	9787
```

- Why this Report : Range of Result as per Year

❑ USE CASE 7

Find the most popular top 10 job positions for H1B visa applications for each year

- Input File – h1b_final Data
- Description : Here we find the number of applicants applied forH1B visa applications in each year
- Tool used : Map-Reduce
- Key: Year   Value: Job title
- Output path : `hadoop jar  h1b_jars/Top10_job.jar /user/hive/warehouse/h1b_final /h1b_project/top10_jobs;`
- Screenshot :



```
2016    PROGRAMMER ANALYST
2016    SOFTWARE ENGINEER
2016    SOFTWARE DEVELOPER
2016    SYSTEMS ANALYST
2016    COMPUTER PROGRAMMER
2016    BUSINESS ANALYST
2016    COMPUTER SYSTEMS ANALYST
2016    SENIOR SOFTWARE ENGINEER
2016    DEVELOPER
2016    TECHNOLOGY LEAD - US
2015    PROGRAMMER ANALYST
2015    SOFTWARE ENGINEER
2015    COMPUTER PROGRAMMER
2015    SYSTEMS ANALYST
2015    SOFTWARE DEVELOPER
2015    BUSINESS ANALYST
2015    TECHNOLOGY LEAD - US
2015    COMPUTER SYSTEMS ANALYST
2015    TECHNOLOGY ANALYST - US
2015    SENIOR SOFTWARE ENGINEER
2014    PROGRAMMER ANALYST
2014    SOFTWARE ENGINEER
2014    COMPUTER PROGRAMMER
2014    SYSTEMS ANALYST
2014    SOFTWARE DEVELOPER
2014    BUSINESS ANALYST
2014    COMPUTER SYSTEMS ANALYST
2014    TECHNOLOGY LEAD - US
2014    TECHNOLOGY ANALYST - US
2014    SENIOR CONSULTANT
2013    PROGRAMMER ANALYST
2013    SOFTWARE ENGINEER
2013    COMPUTER PROGRAMMER
2013    SYSTEMS ANALYST
2013    TECHNOLOGY LEAD - US
2013    TECHNOLOGY ANALYST - US
2013    BUSINESS ANALYST
2013    COMPUTER SYSTEMS ANALYST
2013    SOFTWARE DEVELOPER
2013    SENIOR CONSULTANT
2012    PROGRAMMER ANALYST
2012    SOFTWARE ENGINEER
2012    COMPUTER PROGRAMMER
2012    SYSTEMS ANALYST
2012    BUSINESS ANALYST
2012    COMPUTER SYSTEMS ANALYST
2012    SOFTWARE DEVELOPER
2012    PHYSICAL THERAPIST
```

- Why this Report : To identify the best job for H1b visa applicant

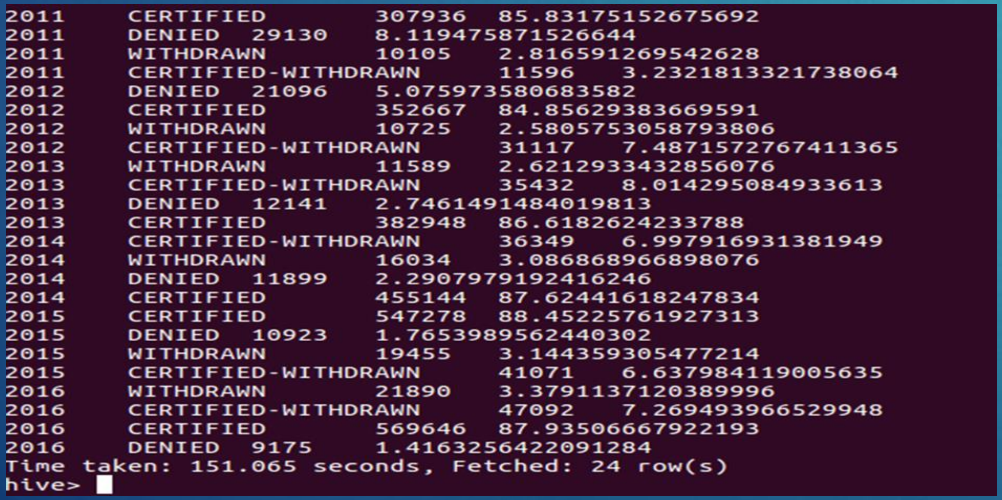
❑ USE CASE 8

Find the percentage and the count of each case status on total applications for each year. Create a graph depicting the pattern of All the cases over the period of time

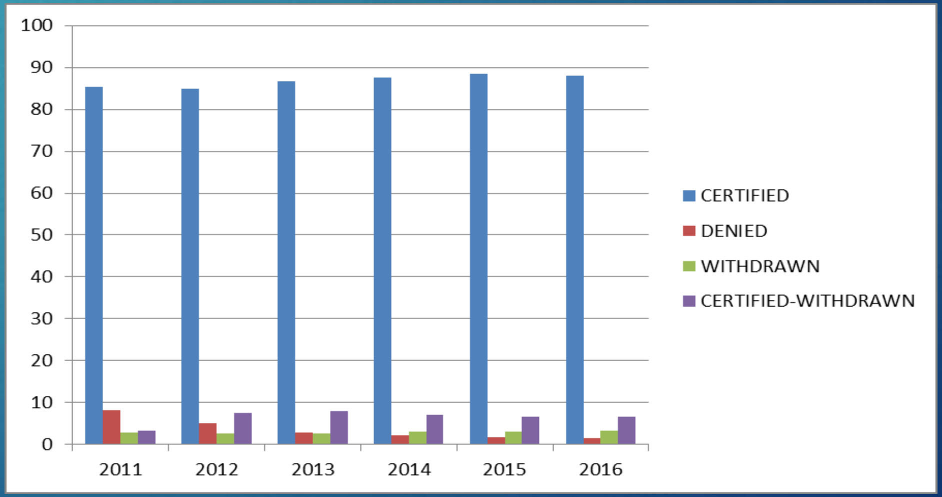
- Table Used : h1b_final Data
- Description: This case count the case status on all application received for each year.
- Tool used : HIVE & PIG
- HIVE Query :

```
select a.year ,case_status,case_count, case_count/year_case*100 case_percent from (select year,case_status,count(*) case_count
from h1b_final2  group by year,case_status) a join year_wise_case b  on a.year=b.year  order by year;
```

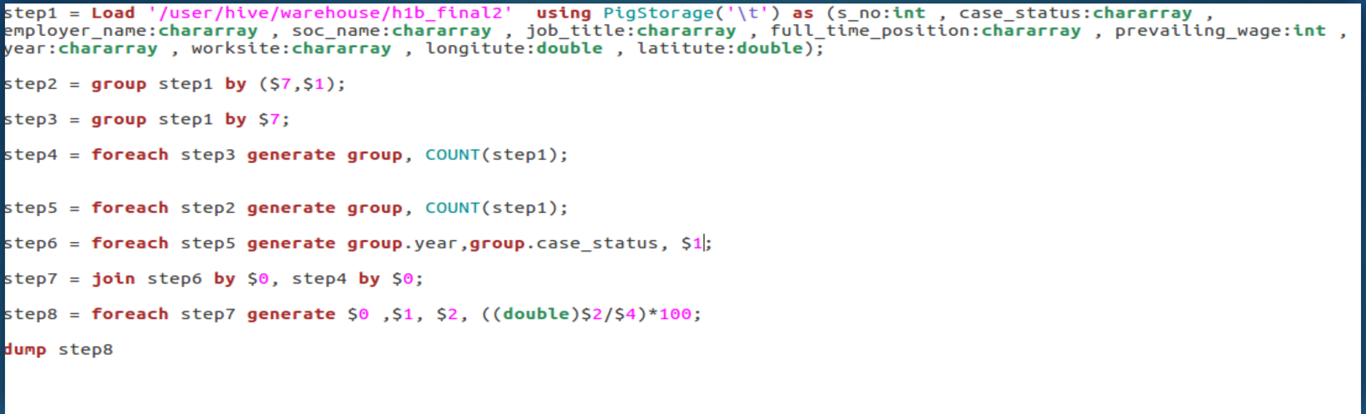
- Screenshot & Graph Representation :



```
2011    CERTIFIED       307936  85.83175152675692
2011    DENIED  29130   8.119475871526644
2011    WITHDRAWN       10105   2.816591269542628
2011    CERTIFIED-WITHDRAWN     11596   3.2321813321738064
2012    DENIED  21096   5.075973580683582
2012    CERTIFIED       352667  84.85629383669591
2012    WITHDRAWN       10725   2.5805753058793806
2012    CERTIFIED-WITHDRAWN     31117   7.4871572767411365
2013    WITHDRAWN       11589   2.6212933432856076
2013    CERTIFIED-WITHDRAWN     35432   8.014295084933613
2013    DENIED  12141   2.7461491484019813
2013    CERTIFIED       382948  86.6182624233788
2014    CERTIFIED-WITHDRAWN     36349   6.997916931381949
2014    WITHDRAWN       16034   3.086868966898076
2014    DENIED  11899   2.2907979192416246
2014    CERTIFIED       455144  87.62441618247834
2015    CERTIFIED       547278  88.45225761927313
2015    DENIED  10923   1.7653989562440302
2015    WITHDRAWN       19455   3.144359305477214
2015    CERTIFIED-WITHDRAWN     41071   6.637984119005635
2016    WITHDRAWN       21890   3.3791137120389996
2016    CERTIFIED-WITHDRAWN     47092   7.269493966529948
2016    CERTIFIED       569646  87.93506667922193
2016    DENIED  9175    1.4163256422091284
Time taken: 151.065 seconds, Fetched: 24 row(s)
hive>
```



- PIG Output : `pig -f pigproject/Graph_percen6.pig`
- Screenshot :



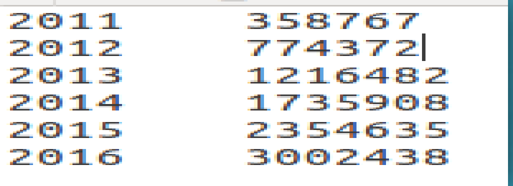
```
step1 = Load '/user/hive/warehouse/h1b_final2'  using PigStorage('\t') as (s_no:int , case_status:chararray ,
employer_name:chararray , soc_name:chararray , job_title:chararray , full_time_position:chararray , prevailing_wage:int ,
year:chararray , worksite:chararray , longitute:double , latitute:double);

step2 = group step1 by ($7,$1);

step3 = group step1 by $7;

step4 = foreach step3 generate group, COUNT(step1);

step5 = foreach step2 generate group, COUNT(step1);

step6 = foreach step5 generate group.year,group.case_status, $1;

step7 = join step6 by $0, step4 by $0;

step8 = foreach step7 generate $0 ,$1, $2, ((double)$2/$4)*100;

dump step8
```

- Why this Report : With this graph, we analyses the case status with higher percentage on applications

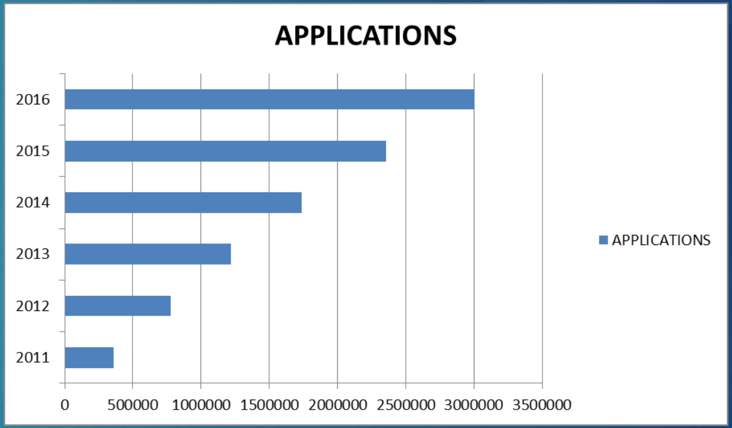
USE CASE 9

Create a bar graph to depict the number of applications for each year

- Input File – h1b_final Data
- Description : To Obtain a Distributed output application for each year
- Tool used : Map-Reduce
- Key: Year   Value: Count
- Output path : hadoop jar h1b_jars/application_each_year.jar /user/hive/warehouse/h1b_final /h1b_project/application_each_year;
- Screenshot & Bar graph :



```
2011          358767
2012          774372
2013          1216482
2014          1735908
2015          2354635
2016          3002438
```

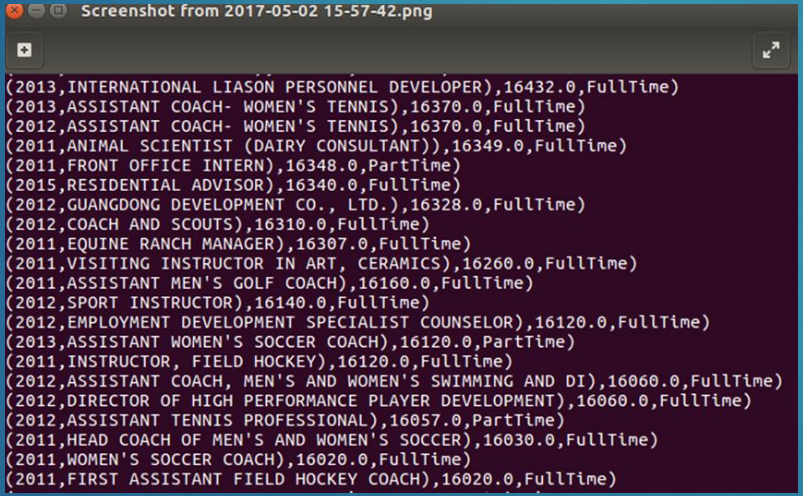


- Why this Report : To obtain a summarized data for application in each year

❑ USE CASE 10

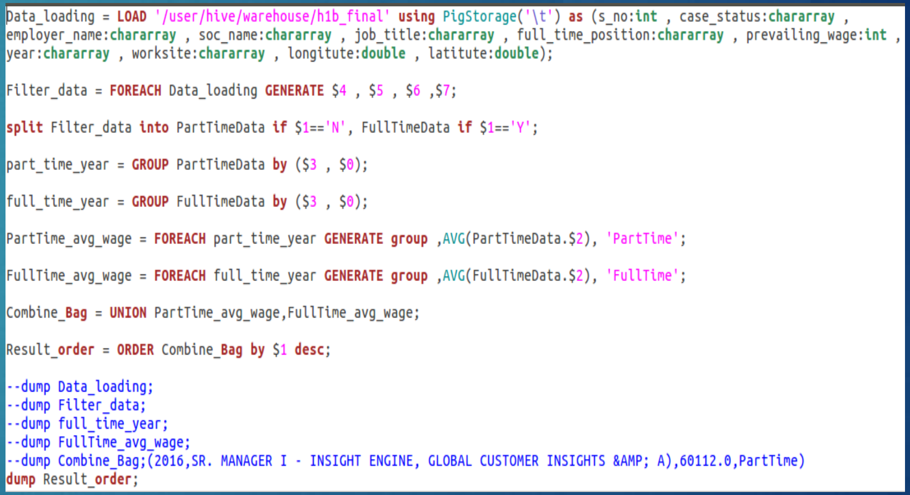
Find the average Prevailing Wage for each Job for each Year (take part time and full time separate). Arrange the output in descending order

- Table Used : h1b_final Data
- Description: This case find the average prevailing wage for all jobs in each year
- Tool used : PIG
- PIG Query :`pig -f pigproject/part_full.pig`
- Screenshot :



Screenshot from 2017-05-02 15-57-42.png

```
(2013,INTERNATIONAL LIASON PERSONNEL DEVELOPER),16432.0,FullTime)
(2013,ASSISTANT COACH- WOMEN'S TENNIS),16370.0,FullTime)
(2012,ASSISTANT COACH- WOMEN'S TENNIS),16370.0,FullTime)
(2011,ANIMAL SCIENTIST (DAIRY CONSULTANT)),16349.0,FullTime)
(2011,FRONT OFFICE INTERN),16348.0,PartTime)
(2015,RESIDENTIAL ADVISOR),16340.0,FullTime)
(2012,GUANGDONG DEVELOPMENT CO., LTD.),16328.0,FullTime)
(2012,COACH AND SCOUTS),16310.0,FullTime)
(2011,EQUINE RANCH MANAGER),16307.0,FullTime)
(2011,VISITING INSTRUCTOR IN ART, CERAMICS),16260.0,FullTime)
(2011,ASSISTANT MEN'S GOLF COACH),16160.0,FullTime)
(2012,SPORT INSTRUCTOR),16140.0,FullTime)
(2012,EMPLOYMENT DEVELOPMENT SPECIALIST COUNSELOR),16120.0,FullTime)
(2013,ASSISTANT WOMEN'S SOCCER COACH),16120.0,PartTime)
(2011,INSTRUCTOR, FIELD HOCKEY),16120.0,FullTime)
(2012,ASSISTANT COACH, MEN'S AND WOMEN'S SWIMMING AND DI),16060.0,FullTime)
(2012,DIRECTOR OF HIGH PERFORMANCE PLAYER DEVELOPMENT),16060.0,FullTime)
(2012,ASSISTANT TENNIS PROFESSIONAL),16057.0,PartTime)
(2011,HEAD COACH OF MEN'S AND WOMEN'S SOCCER),16030.0,FullTime)
(2011,WOMEN'S SOCCER COACH),16020.0,FullTime)
(2011,FIRST ASSISTANT FIELD HOCKEY COACH),16020.0,FullTime)
```



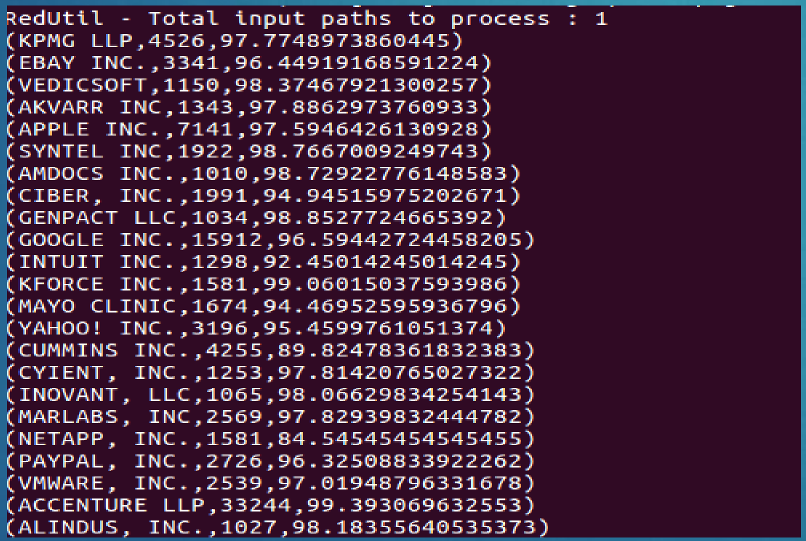
```
Data_loading = LOAD '/user/hive/warehouse/h1b_final' using PigStorage('\t') as (s_no:int , case_status:chararray ,
employer_name:chararray , soc_name:chararray , job_title:chararray , full_time_position:chararray , prevailing_wage:int ,
year:chararray , worksite:chararray , longitute:double , latitute:double);

Filter_data = FOREACH Data_loading GENERATE $4 , $5 , $6 ,$7;

split Filter_data into PartTimeData if $1=='N', FullTimeData if $1=='Y';

part_time_year = GROUP PartTimeData by ($3 , $0);

full_time_year = GROUP FullTimeData by ($3 , $0);

PartTime_avg_wage = FOREACH part_time_year GENERATE group ,AVG(PartTimeData.$2), 'PartTime';

FullTime_avg_wage = FOREACH full_time_year GENERATE group ,AVG(FullTimeData.$2), 'FullTime';

Combine_Bag = UNION PartTime_avg_wage,FullTime_avg_wage;

Result_order = ORDER Combine_Bag by $1 desc;

--dump Data_loading;
--dump Filter_data;
--dump full_time_year;
--dump FullTime_avg_wage;
--dump Combine_Bag;(2016,SR. MANAGER I - INSIGHT ENGINE, GLOBAL CUSTOMER INSIGHTS &AMP; A),60112.0,PartTime)
dump Result_order;
```

- Why this Report : For an effective analysis

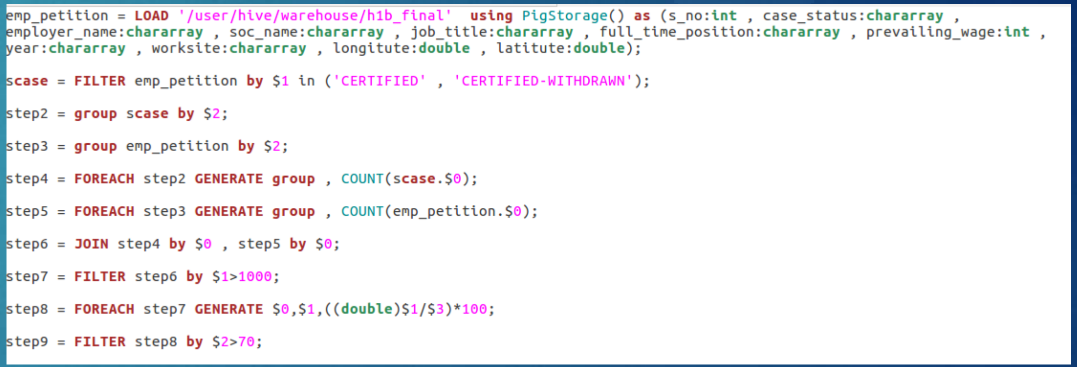
❑ USE CASE 11

Which are employers along with the number of petitions who have the success rate more than 70% in petitions and total petitions filed more than 1000

- Table Used : h1b_final Data
- Description:
- Tool used : PIG
- PIG Query : `pig -f pigproject/emp_petition.pig`
- Screenshot :



```
RedUtil - Total input paths to process : 1
(KPMG LLP,4526,97.7748973860445)
(EBAY INC.,3341,96.44919168591224)
(VEDICSOFT,1150,98.37467921300257)
(AKVARR INC,1343,97.8862973760933)
(APPLE INC.,7141,97.5946426130928)
(SYNTEL INC,1922,98.7667009249743)
(AMDOCS INC.,1010,98.72922776148583)
(CIBER, INC.,1991,94.94515975202671)
(GENPACT LLC,1034,98.8527724665392)
(GOOGLE INC.,15912,96.59442724458205)
(INTUIT INC.,1298,92.45014245014245)
(KFORCE INC.,1581,99.06015037593986)
(MAYO CLINIC,1674,94.46952595936796)
(YAHOO! INC.,3196,95.4599761051374)
(CUMMINS INC.,4255,89.82478361832383)
(CYIENT, INC.,1253,97.81420765027322)
(INOVANT, LLC,1065,98.06629834254143)
(MARLABS, INC,2569,97.82939832444782)
(NETAPP, INC.,1581,84.54545454545455)
(PAYPAL, INC.,2726,96.32508833922262)
(VMWARE, INC.,2539,97.01948796331678)
(ACCENTURE LLP,33244,99.393069632553)
(ALINDUS, INC.,1027,98.18355640535373)
```



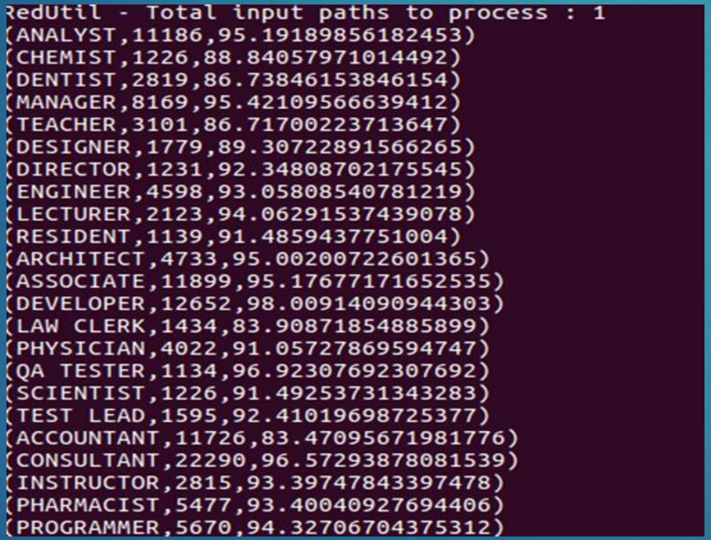
```
emp_petition = LOAD '/user/hive/warehouse/h1b_final'  using PigStorage() as (s_no:int , case_status:chararray ,
employer_name:chararray , soc_name:chararray , job_title:chararray , full_time_position:chararray , prevailing_wage:int ,
year:chararray , worksite:chararray , longitute:double , latitute:double);

scase = FILTER emp_petition by $1 in ('CERTIFIED' , 'CERTIFIED-WITHDRAWN');

step2 = group scase by $2;

step3 = group emp_petition by $2;

step4 = FOREACH step2 GENERATE group , COUNT(scase.$0);

step5 = FOREACH step3 GENERATE group , COUNT(emp_petition.$0);

step6 = JOIN step4 by $0 , step5 by $0;

step7 = FILTER step6 by $1>1000;

step8 = FOREACH step7 GENERATE $0,$1,((double)$1/$3)*100;

step9 = FILTER step8 by $2>70;
```

- Why this Report : For an effective analsyis

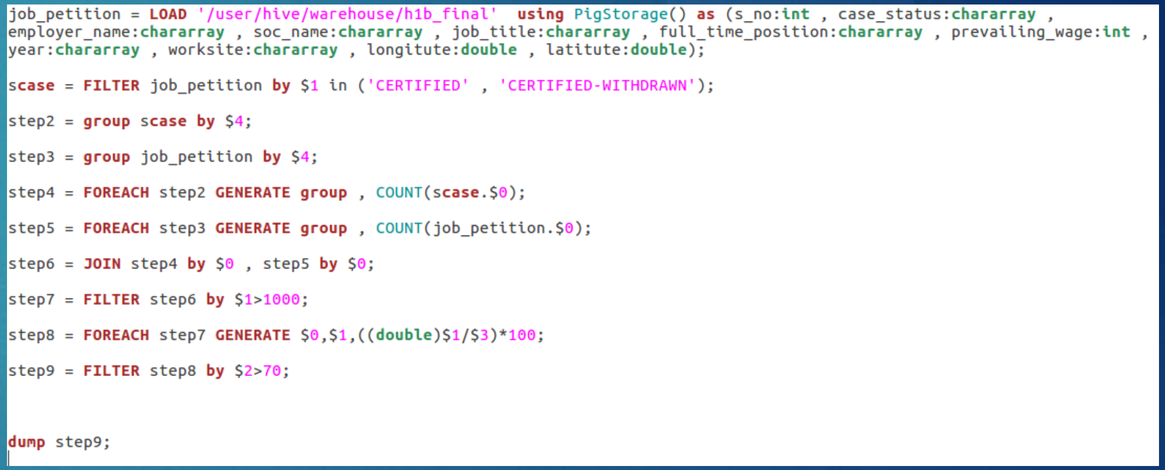
❑ USE CASE 12

Which are the job positions along with the number of petitions which have the success rate more than 70% in petitions and total petitions filed more than 1000

- Table Used : h1b_final Data
- Description: This case find the average prevailing wage for all jobs in each year
- Tool used : PIG
- PIG Query :`pig -f pigproject/job_petition.pig`
- **Screenshot :**



```
RedUtil - Total input paths to process : 1
(ANALYST,11186,95.19189856182453)
(CHEMIST,1226,88.84057971014492)
(DENTIST,2819,86.73846153846154)
(MANAGER,8169,95.42109566639412)
(TEACHER,3101,86.71700223713647)
(DESIGNER,1779,89.30722891566265)
(DIRECTOR,1231,92.34808702175545)
(ENGINEER,4598,93.05808540781219)
(LECTURER,2123,94.06291537439078)
(RESIDENT,1139,91.4859437751004)
(ARCHITECT,4733,95.00200722601365)
(ASSOCIATE,11899,95.17677171652535)
(DEVELOPER,12652,98.00914090944303)
(LAW CLERK,1434,83.90871854885899)
(PHYSICIAN,4022,91.05727869594747)
(QA TESTER,1134,96.92307692307692)
(SCIENTIST,1226,91.49253731343283)
(TEST LEAD,1595,92.41019698725377)
(ACCOUNTANT,11726,83.47095671981776)
(CONSULTANT,22290,96.57293878081539)
(INSTRUCTOR,2815,93.39747843397478)
(PHARMACIST,5477,93.40040927694406)
(PROGRAMMER,5670,94.32706704375312)
```



```
job_petition = LOAD '/user/hive/warehouse/h1b_final'  using PigStorage() as (s_no:int , case_status:chararray ,
employer_name:chararray , soc_name:chararray , job_title:chararray , full_time_position:chararray , prevailing_wage:int ,
year:chararray , worksite:chararray , longitute:double , latitute:double);

scase = FILTER job_petition by $1 in ('CERTIFIED' , 'CERTIFIED-WITHDRAWN');

step2 = group scase by $4;

step3 = group job_petition by $4;

step4 = FOREACH step2 GENERATE group , COUNT(scase.$0);

step5 = FOREACH step3 GENERATE group , COUNT(job_petition.$0);

step6 = JOIN step4 by $0 , step5 by $0;

step7 = FILTER step6 by $1>1000;

step8 = FOREACH step7 GENERATE $0,$1,((double)$1/$3)*100;

step9 = FILTER step8 by $2>70;


dump step9;
```

- Why this Report : Analyzing on job status who having good growth in over all cycle

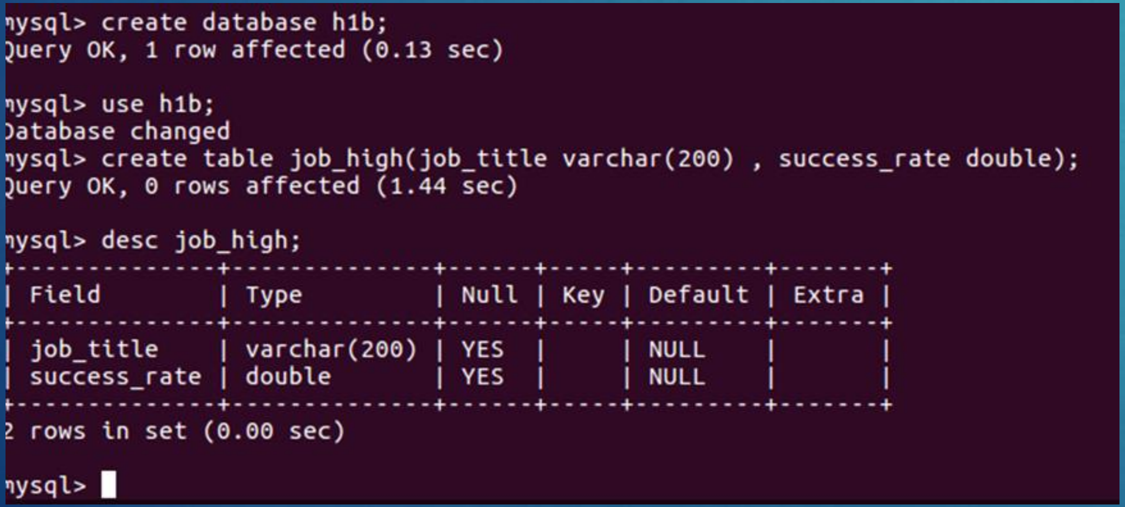
❑ USE CASE 13

Export result for question no 10 to MySQL database

- Input file : h1b_final Data
- Description:
- Tool Used: MySQL , HIVE , SQOOP
- Steps Follows :

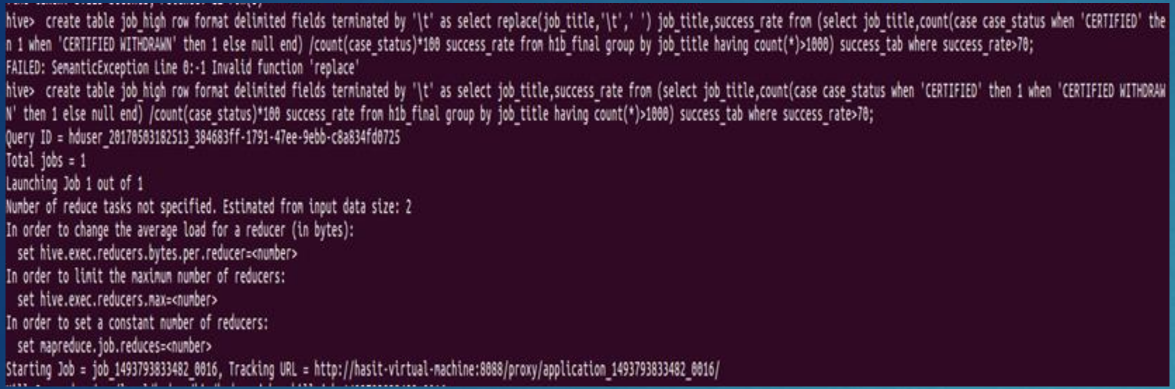
STEP 1 - Creating SQL data base with Table !!

Create table job_high(job_title varchar(200) , success_rate double);



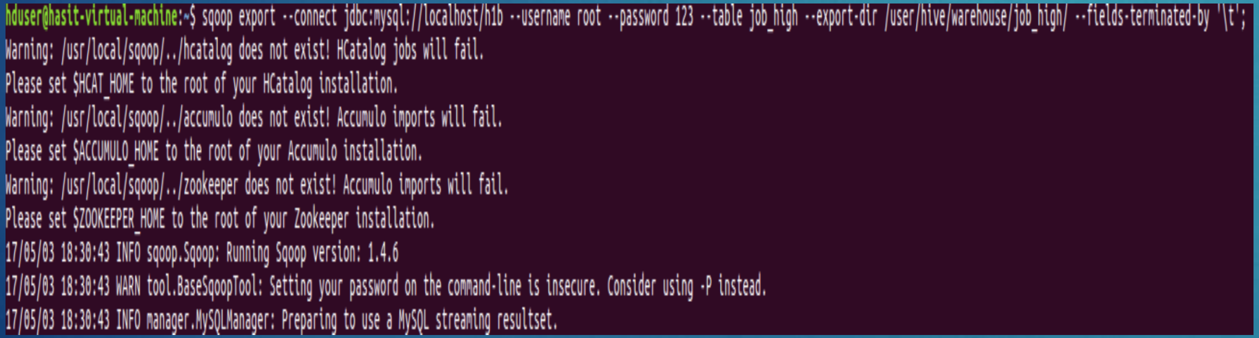
```
mysql> create database h1b;
Query OK, 1 row affected (0.13 sec)

mysql> use h1b;
Database changed
mysql> create table job_high(job_title varchar(200) , success_rate double);
Query OK, 0 rows affected (1.44 sec)

mysql> desc job_high;
+--------------+--------------+------+-----+---------+-------+
| Field        | Type         | Null | Key | Default | Extra |
+--------------+--------------+------+-----+---------+-------+
| job_title    | varchar(200) | YES  |     | NULL    |       |
| success_rate | double       | YES  |     | NULL    |       |
+--------------+--------------+------+-----+---------+-------+
2 rows in set (0.00 sec)

mysql> █
```

Step 2:Creating Hive table with query !!!



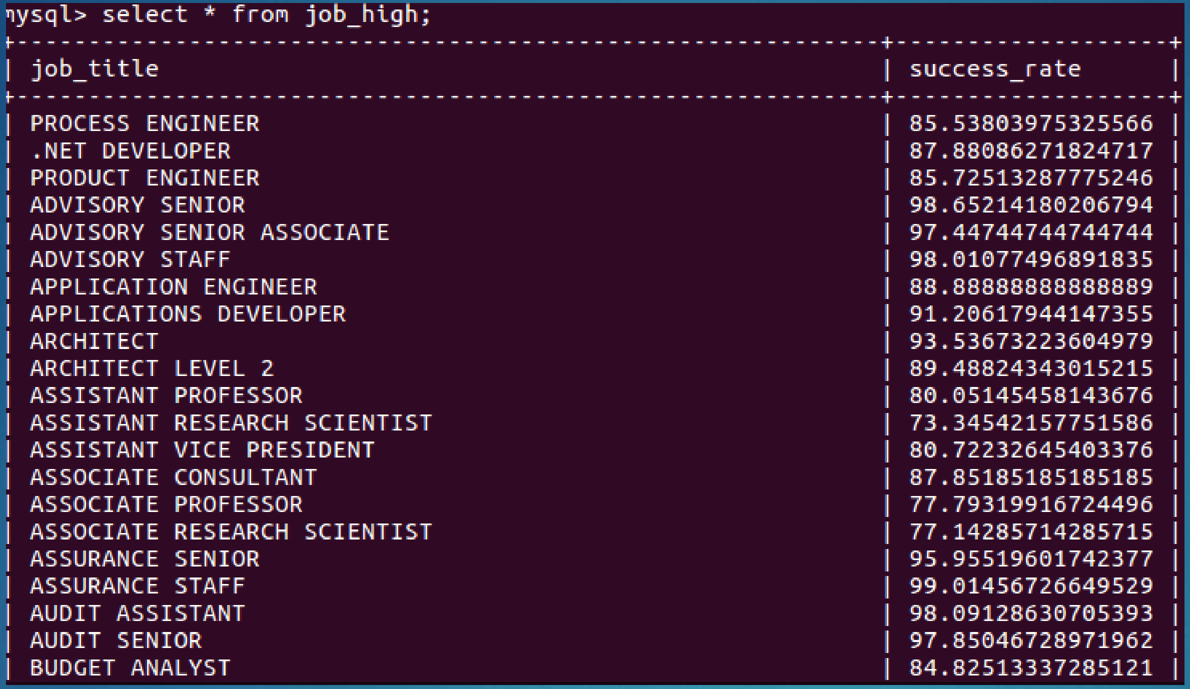
```
hive>  create table job_high row format delimited fields terminated by '\t' as select replace(job_title,'\t',' ') job_title,success_rate from (select job_title,count(case case_status when 'CERTIFIED' the
n 1 when 'CERTIFIED WITHDRAWN' then 1 else null end) /count(case_status)*100 success_rate from h1b_final group by job_title having count(*)>1000) success_tab where success_rate>70;
FAILED: SemanticException Line 0:-1 Invalid function 'replace'
hive>  create table job_high row format delimited fields terminated by '\t' as select job_title,success_rate from (select job_title,count(case case_status when 'CERTIFIED' then 1 when 'CERTIFIED WITHDRAW
N' then 1 else null end) /count(case_status)*100 success_rate from h1b_final group by job_title having count(*)>1000) success_tab where success_rate>70;
Query ID = hduser_20170503182513_384683ff-1791-47ee-9ebb-c8a834fd0725
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 2
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1493793833482_0016, Tracking URL = http://hasit-virtual-machine:8088/proxy/application_1493793833482_0016/
```

STEP 3: Exporting data with Sqoop to MySQL table !!!!



```
hduser@hasit-virtual-machine:~$ sqoop export --connect jdbc:mysql://localhost/h1b --username root --password 123 --table job_high --export-dir /user/hive/warehouse/job_high/ --fields-terminated-by '\t';
Warning: /usr/local/sqoop/../hcatalog does not exist! HCatalog jobs will fail.
Please set $HCAT_HOME to the root of your HCatalog installation.
Warning: /usr/local/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
Warning: /usr/local/sqoop/../zookeeper does not exist! Accumulo imports will fail.
Please set $ZOOKEEPER_HOME to the root of your Zookeeper installation.
17/05/03 18:30:43 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6
17/05/03 18:30:43 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
17/05/03 18:30:43 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
```

STEP 4 :Data has been fetched from SQOOP and has been loaded in MySQL



```
nysql> select * from job_high;
+------------------------------------------------------------+-------------------+
| job_title                                                  | success_rate      |
+------------------------------------------------------------+-------------------+
| PROCESS ENGINEER                                           | 85.53803975325566 |
| .NET DEVELOPER                                             | 87.88086271824717 |
| PRODUCT ENGINEER                                           | 85.72513287775246 |
| ADVISORY SENIOR                                            | 98.65214180206794 |
| ADVISORY SENIOR ASSOCIATE                                  | 97.44744744744744 |
| ADVISORY STAFF                                             | 98.01077496891835 |
| APPLICATION ENGINEER                                       | 88.88888888888889 |
| APPLICATIONS DEVELOPER                                     | 91.20617944147355 |
| ARCHITECT                                                  | 93.53673223604979 |
| ARCHITECT LEVEL 2                                          | 89.48824343015215 |
| ASSISTANT PROFESSOR                                        | 80.05145458143676 |
| ASSISTANT RESEARCH SCIENTIST                               | 73.34542157751586 |
| ASSISTANT VICE PRESIDENT                                   | 80.72232645403376 |
| ASSOCIATE CONSULTANT                                       | 87.85185185185185 |
| ASSOCIATE PROFESSOR                                        | 77.79319916724496 |
| ASSOCIATE RESEARCH SCIENTIST                               | 77.14285714285715 |
| ASSURANCE SENIOR                                           | 95.95519601742377 |
| ASSURANCE STAFF                                            | 99.01456726649529 |
| AUDIT ASSISTANT                                            | 98.09128630705393 |
| AUDIT SENIOR                                               | 97.85046728971962 |
| BUDGET ANALYST                                             | 84.82513337285121 |
```

. Why this Report : Exporting data to MySql from SQOOP

Thank You

Thank you

Thank you