BIG DATA HADOOP & SPARK TRAINING

ACADGILD

ASSIGNMENT 8.1

BY :-

SAHIL KHURANA

PROBLEM STATEMENT

Get a list of employees who receive a salary less than 100, compared to their immediate employee with higher salary in the same unit

List of all employees who draw higher salary than the average salary of that department.

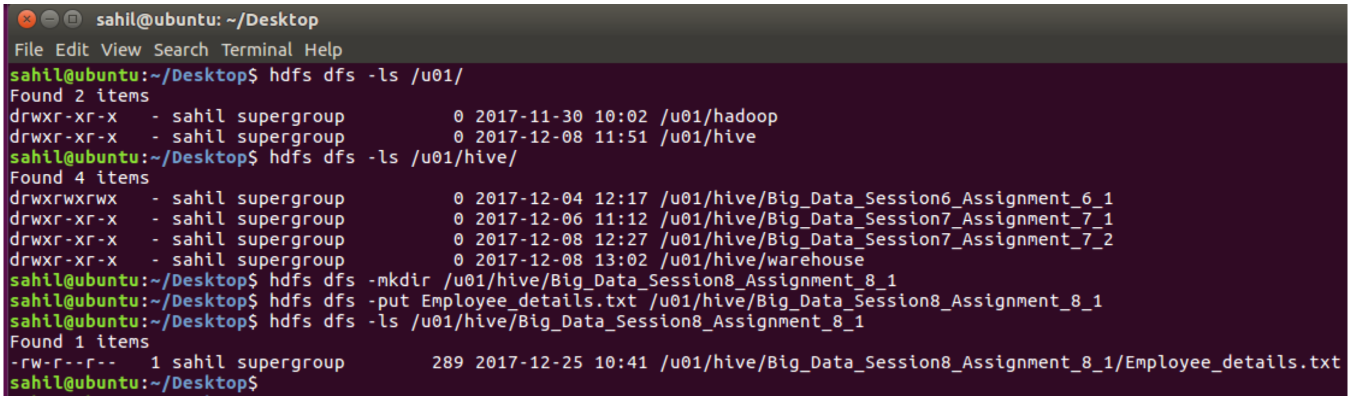
Associated Data Files

Employee_details.txt

101,Amitabh,200,1
102,Shahrukh,100,2
103,Akshay,110,3
104,Anubhav,500,4
105,Pawan,250,5
106,Aamir,250,1
107,Salman,175,2
108,Ranbir,140,3
109,Katrina,100,4
110,Priyanka,200,5
111,Tushar,500,1
112,Ajay,500,2
113,Jubeen,100,1
114,Madhuri,200,2
115,Sahil,100,2
116,Khurana,20,2

Note: - To solve the Assignment 6.1, I have created a VM with Ubuntu 16.04 OS and configured Hadoop 2.8.2 and hive-2.3.2 on the same.

Step 1:- Put the dataset in HDFS location

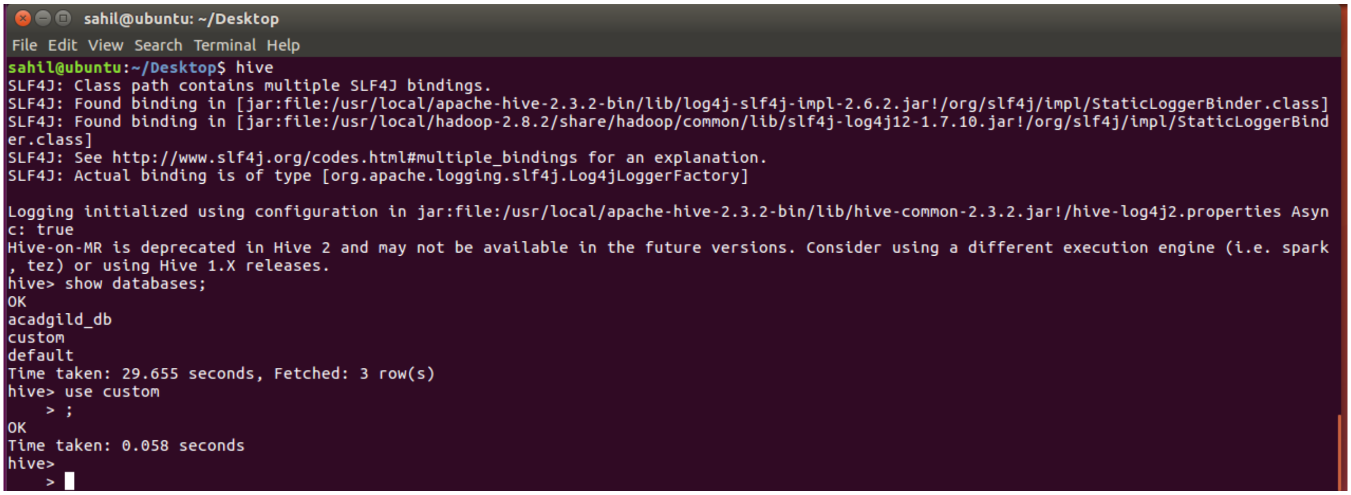


```
sahil@ubuntu:~/Desktop$ hdfs dfs -ls /u01/
Found 2 items
drwxr-xr-x   - sahil supergroup          0 2017-11-30 10:02 /u01/hadoop
drwxr-xr-x   - sahil supergroup          0 2017-12-08 11:51 /u01/hive
sahil@ubuntu:~/Desktop$ hdfs dfs -ls /u01/hive/
Found 4 items
drwxrwxrwx   - sahil supergroup          0 2017-12-04 12:17 /u01/hive/Big_Data_Session6_Assignment_6_1
drwxr-xr-x   - sahil supergroup          0 2017-12-06 11:12 /u01/hive/Big_Data_Session7_Assignment_7_1
drwxr-xr-x   - sahil supergroup          0 2017-12-08 12:27 /u01/hive/Big_Data_Session7_Assignment_7_2
drwxr-xr-x   - sahil supergroup          0 2017-12-08 13:02 /u01/hive/warehouse
sahil@ubuntu:~/Desktop$ hdfs dfs -mkdir /u01/hive/Big_Data_Session8_Assignment_8_1
sahil@ubuntu:~/Desktop$ hdfs dfs -put Employee_details.txt /u01/hive/Big_Data_Session8_Assignment_8_1
sahil@ubuntu:~/Desktop$ hdfs dfs -ls /u01/hive/Big_Data_Session8_Assignment_8_1
Found 1 items
-rw-r--r--   1 sahil supergroup        289 2017-12-25 10:41 /u01/hive/Big_Data_Session8_Assignment_8_1/Employee_details.txt
sahil@ubuntu:~/Desktop$
```

Step 2:- Open the Hive Shell and CREATE the DATABASE.

Commands used in Step 2

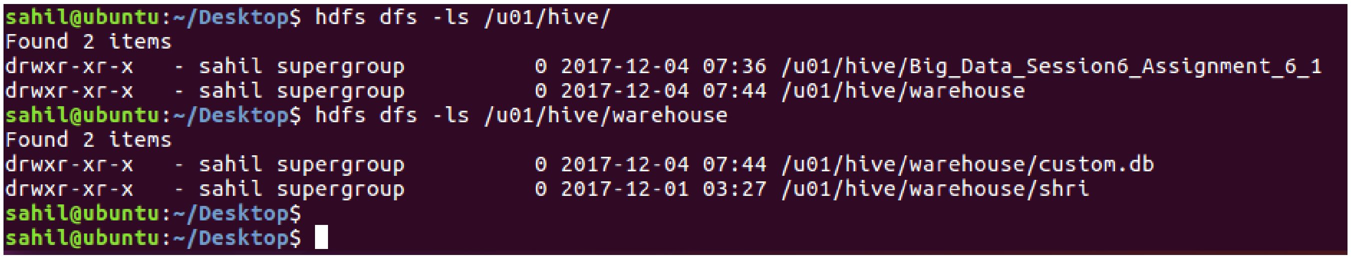
```
hive                                        -- open the hive shell
hive> create database if not exists custom; -- create database
hive> show databases;                       -- check whether database is created or not
hive> use custom;
```



```
sahil@ubuntu:~/Desktop$ hive
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/local/apache-hive-2.3.2-bin/lib/log4j-slf4j-impl-2.6.2.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/local/hadoop-2.8.2/share/hadoop/common/lib/slf4j-log4j12-1.7.10.jar!/org/slf4j/impl/StaticLoggerBind
er.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]

Logging initialized using configuration in jar:file:/usr/local/apache-hive-2.3.2-bin/lib/hive-common-2.3.2.jar!/hive-log4j2.properties Asyn
c: true
Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spark
, tez) or using Hive 1.X releases.
hive> show databases;
OK
acadgild_db
custom
default
Time taken: 29.655 seconds, Fetched: 3 row(s)
hive> use custom
    > ;
OK
Time taken: 0.058 seconds
hive>
    > 
```

Custom database created in default directory in hive



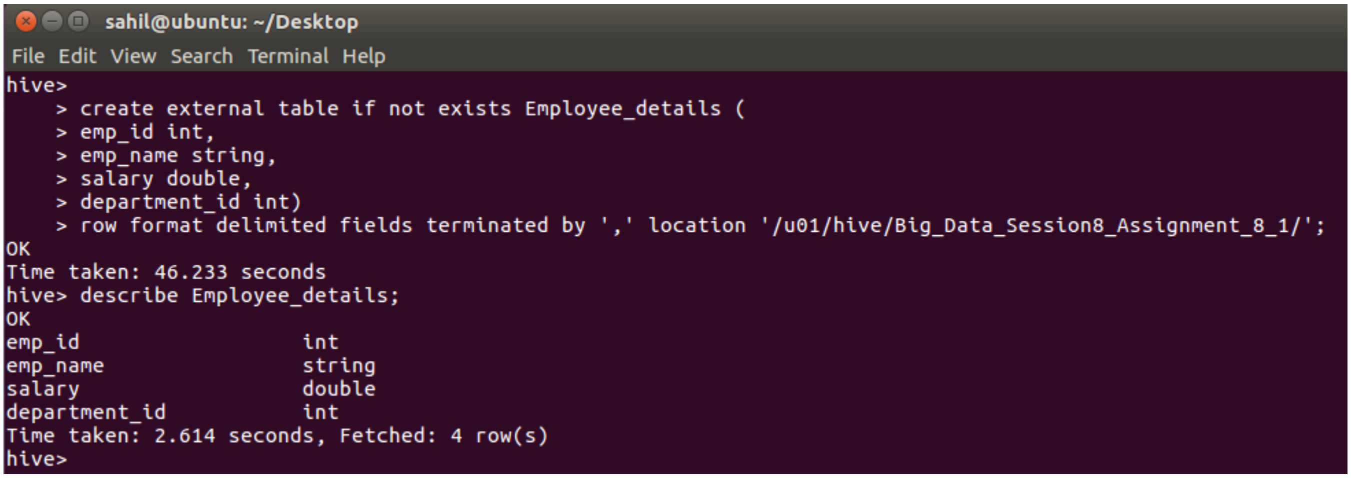
```
sahil@ubuntu:~/Desktop$ hdfs dfs -ls /u01/hive/
Found 2 items
drwxr-xr-x   - sahil supergroup          0 2017-12-04 07:36 /u01/hive/Big_Data_Session6_Assignment_6_1
drwxr-xr-x   - sahil supergroup          0 2017-12-04 07:44 /u01/hive/warehouse
sahil@ubuntu:~/Desktop$ hdfs dfs -ls /u01/hive/warehouse
Found 2 items
drwxr-xr-x   - sahil supergroup          0 2017-12-04 07:44 /u01/hive/warehouse/custom.db
drwxr-xr-x   - sahil supergroup          0 2017-12-01 03:27 /u01/hive/warehouse/shri
sahil@ubuntu:~/Desktop$
sahil@ubuntu:~/Desktop$ 
```

Step 3:- CREATE EXTERNAL TABLE

Commands used in Step 3

```
hive>
    > create external table if not exists Employee_details (
    > emp_id int,
    > emp_name string,
    > salary double,
    > department_id int)
    > row format delimited fields terminated by ',' location
'/u01/hive/Big_Data_Session8_Assignment_8_1/';

hive> describe Employee_details;
```

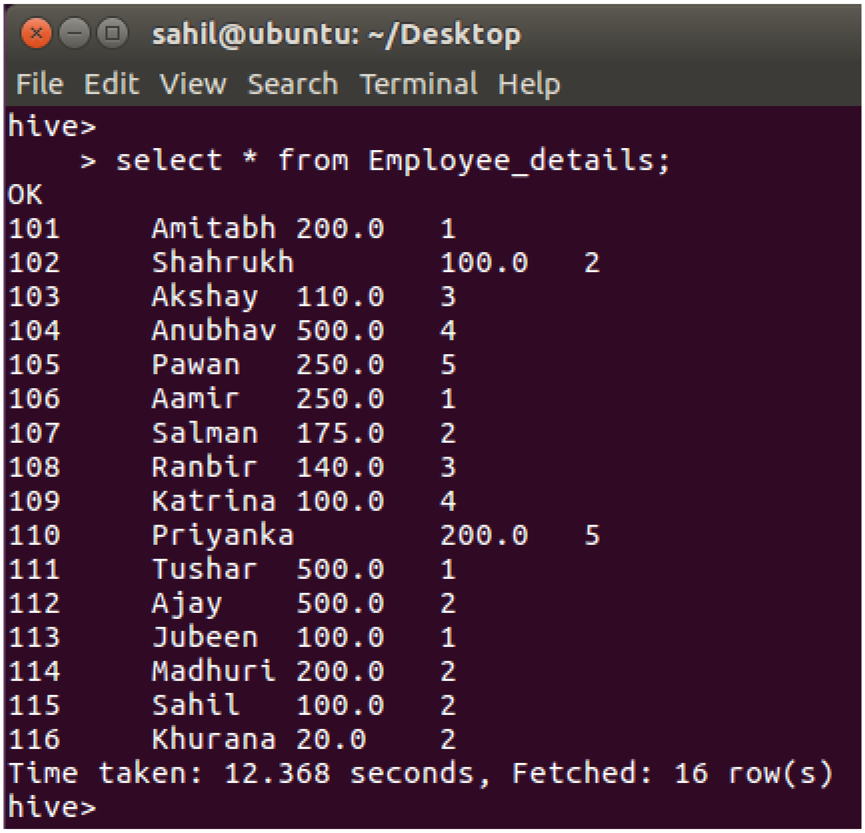


```
hive>
    > create external table if not exists Employee_details (
    > emp_id int,
    > emp_name string,
    > salary double,
    > department_id int)
    > row format delimited fields terminated by ',' location '/u01/hive/Big_Data_Session8_Assignment_8_1/';
OK
Time taken: 46.233 seconds
hive> describe Employee_details;
OK
emp_id                  int
emp_name                string
salary                  double
department_id           int
Time taken: 2.614 seconds, Fetched: 4 row(s)
hive>
```

Step 4 :- Check whether the dataset is imported in the hive table or not.

Commands used in Step 4

```
hive>
    > select * from Employee_details;
```



```
hive>
    > select * from Employee_details;
OK
101     Amitabh 200.0   1
102     Shahrukh        100.0   2
103     Akshay  110.0   3
104     Anubhav 500.0   4
105     Pawan   250.0   5
106     Aamir   250.0   1
107     Salman  175.0   2
108     Ranbir  140.0   3
109     Katrina 100.0   4
110     Priyanka        200.0   5
111     Tushar  500.0   1
112     Ajay    500.0   2
113     Jubeen  100.0   1
114     Madhuri 200.0   2
115     Sahil   100.0   2
116     Khurana 20.0    2
Time taken: 12.368 seconds, Fetched: 16 row(s)
hive>
```

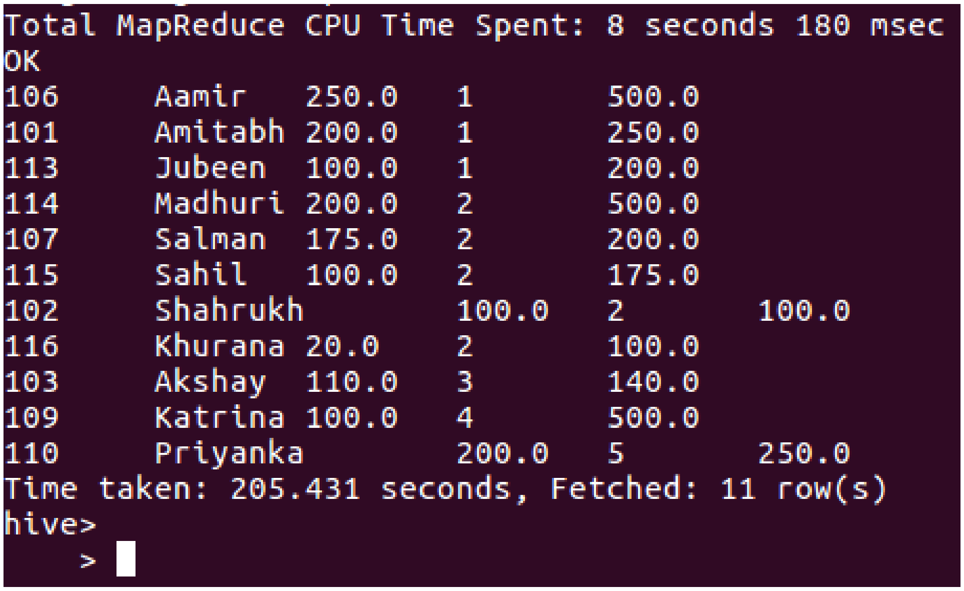
Task 1:-

Get a list of employees who receive a salary less than 100, compared to their immediate employee with higher salary in the same unit

Commands used in Task 1

```
hive> select * from
    > (select *,lag(salary,1,0) over
    > (partition by department_id order by salary desc) as sal1
    > from Employee_details) as sal2
    > where (salary - sal1)< 100;
```

Result:-



```
Total MapReduce CPU Time Spent: 8 seconds 180 msec
OK
106     Aamir   250.0   1       500.0
101     Amitabh 200.0   1       250.0
113     Jubeen  100.0   1       200.0
114     Madhuri 200.0   2       500.0
107     Salman  175.0   2       200.0
115     Sahil   100.0   2       175.0
102     Shahrukh        100.0   2       100.0
116     Khurana 20.0    2       100.0
103     Akshay  110.0   3       140.0
109     Katrina 100.0   4       500.0
110     Priyanka        200.0   5       250.0
Time taken: 205.431 seconds, Fetched: 11 row(s)
hive>
    >
```

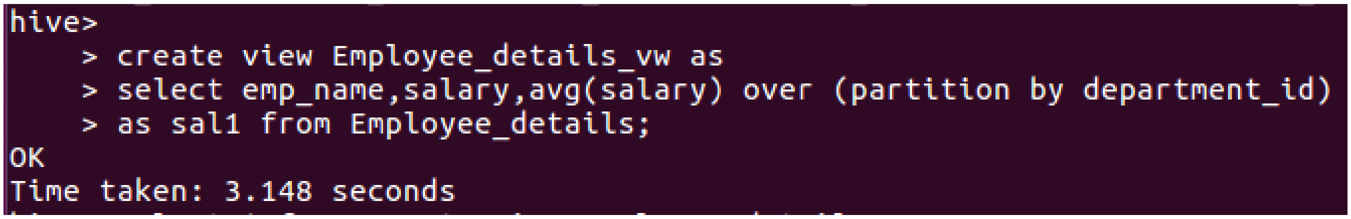
Task 2:-

List of all employees who draw higher salary than the average salary of that department.

Commands used in Task 2

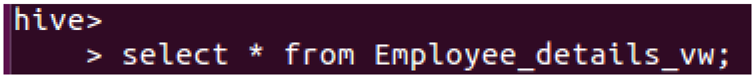
hive>

> create view Employee_details_vw as select name,salary,avg(salary) over (partition by department_id) as sal1 from Employee_details;

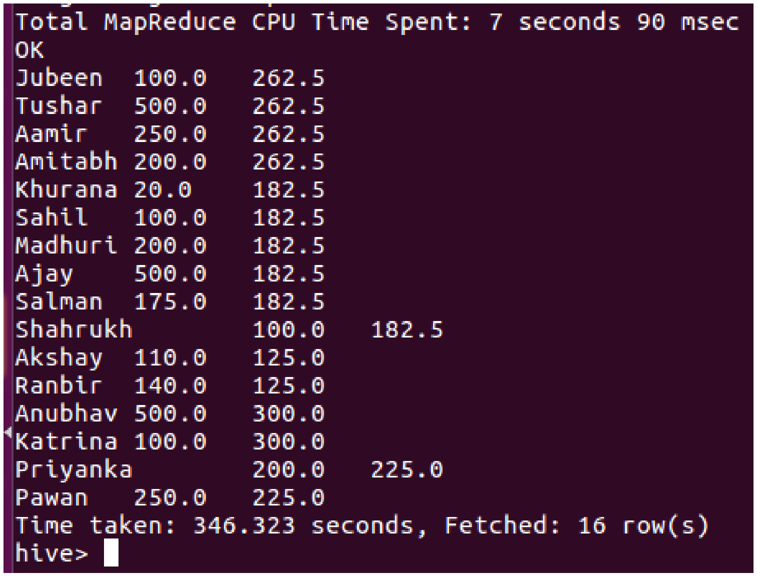


```
hive>
    > create view Employee_details_vw as
    > select emp_name,salary,avg(salary) over (partition by department_id)
    > as sal1 from Employee_details;
OK
Time taken: 3.148 seconds
```

hive> select * from Employee_details_vw;

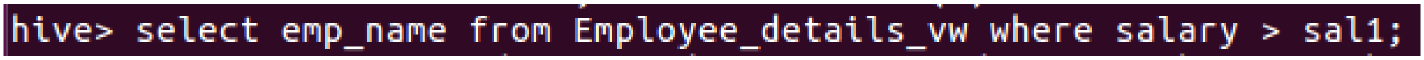


```
hive>
    > select * from Employee_details_vw;
```



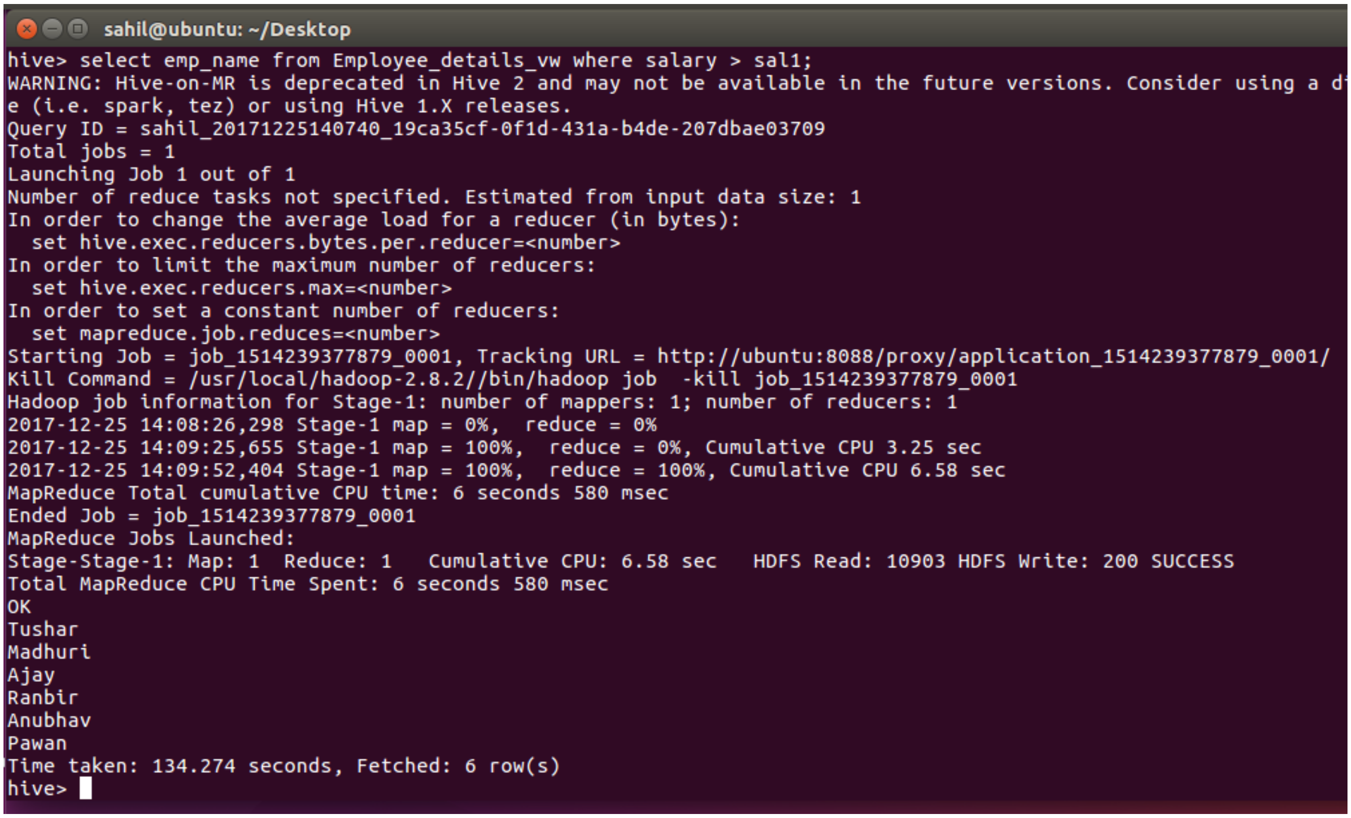
```
Total MapReduce CPU Time Spent: 7 seconds 90 msec
OK
Jubeen  100.0   262.5
Tushar  500.0   262.5
Aamir   250.0   262.5
Amitabh 200.0   262.5
Khurana 20.0    182.5
Sahil   100.0   182.5
Madhuri 200.0   182.5
Ajay    500.0   182.5
Salman  175.0   182.5
Shahrukh        100.0   182.5
Akshay  110.0   125.0
Ranbir  140.0   125.0
Anubhav 500.0   300.0
Katrina 100.0   300.0
Priyanka        200.0   225.0
Pawan   250.0   225.0
Time taken: 346.323 seconds, Fetched: 16 row(s)
hive>
```

hive> select emp_name from Employee_details_vw where salary > sal1;



```
hive> select emp_name from Employee_details_vw where salary > sal1;
```

Final Result:-



```
hive> select emp_name from Employee_details_vw where salary > sal1;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a d
e (i.e. spark, tez) or using Hive 1.X releases.
Query ID = sahil_20171225140740_19ca35cf-0f1d-431a-b4de-207dbae03709
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 1
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_1514239377879_0001, Tracking URL = http://ubuntu:8088/proxy/application_1514239377879_0001/
Kill Command = /usr/local/hadoop-2.8.2//bin/hadoop job  -kill job_1514239377879_0001
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2017-12-25 14:08:26,298 Stage-1 map = 0%,  reduce = 0%
2017-12-25 14:09:25,655 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 3.25 sec
2017-12-25 14:09:52,404 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 6.58 sec
MapReduce Total cumulative CPU time: 6 seconds 580 msec
Ended Job = job_1514239377879_0001
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 6.58 sec   HDFS Read: 10903 HDFS Write: 200 SUCCESS
Total MapReduce CPU Time Spent: 6 seconds 580 msec
OK
Tushar
Madhuri
Ajay
Ranbir
Anubhav
Pawan
Time taken: 134.274 seconds, Fetched: 6 row(s)
hive> 
```