# ASSIGNMENT 8.1

## Hive queries:

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

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

## Solution:

### Table creation query:

CREATE TABLE  emp_info

(emp_id INT,

emp_name STRING,

emp_salary INT,

emp_unit STRING)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY '\t';

```
hive> CREATE TABLE emp_info
    > (emp_id INT,
    > emp_name STRING,
    > emp_salary INT,
    > emp_unit STRING)
    > ROW FORMAT DELIMITED
    > FIELDS TERMINATED BY '\t';
OK
Time taken: 0.389 seconds
```

### Data load query:

LOAD DATA LOCAL INPATH '/home/acadgild/Emp_Sal.txt' INTO TABLE emp_info;

```
hive> LOAD DATA LOCAL INPATH '/home/acadgild/Emp_Sal.txt' INTO TABLE emp_info;
Loading data to table custom.emp_info
OK
Time taken: 1.827 seconds
hive> SELECT * FROM emp_info;
OK
1       Amit    105     Data Minning
2       Pankaj  85      Data Engineer
3       Kiran   110     Data Scientist
4       Arpitha 95      Data Engineer
5       Viraj   105     Data Mining
6       Smitha  80      Data Analyst
7       Supriya 90      Data Engineer
8       Vihan   120     Data Scientist
9       Emma    100     Data Engineer
10      Siddharath      100     Data Engineer
Time taken: 0.61 seconds, Fetched: 10 row(s)
```

**Problem 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.

**Query:**

```
SELECT emp_id, emp_name
FROM emp_info
WHERE emp_salary < 100;
```

**Comments:**

In the above query, I am trying to fetch employee 's id and name who get salary less than 100 taking all employees into account in same unit. We can see the output below:

```
hive> SELECT emp_id, emp_name
    > FROM emp_info
    > WHERE emp_salary < 100;
OK
2       Pankaj
4       Arpitha
6       Smitha
7       Supriya
Time taken: 15.632 seconds, Fetched: 4 row(s)
hive>
```

**Problem 2:** List of all employees who draw higher salary than the average salary of that department.

**Query:**

```
SELECT e.emp_id, e.emp_name
FROM emp_info e
WHERE e.emp_salary > (SELECT AVG(emp_salary)
                      FROM emp_info
                     WHERE emp_unit = e.emp_unit
                      GROUP BY emp_unit);
```

**Comments:**

I am using the concept of sub-queries for this problem. Firstly, the inner query will get executed and the resulting salary value (average salary for each unit) will be compared with salary of each employee. If the employee salary is greater than average salary of his/her unit, those employee's id and name will be returned as result. This is achieved by executing the outer query.

```
hive> SELECT e.emp_id, e.emp_name
    > FROM emp_info e
    > WHERE e.emp_salary >
    > (
    >   SELECT AVG(emp_salary)
    >   FROM emp_info
    >   WHERE emp_unit = e.emp_unit
    >   GROUP BY emp_unit
    > );
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execu

tion engine (i.e. spark, tez) or using Hive 1.X releases.
Query ID = acadgild_20171217183402_bd3d1e03-1979-4e45-b255-f0ac2d842180
Total jobs = 4
Launching Job 1 out of 4
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_1513491304174_0011, Tracking URL = http://localhost:8088/proxy/application_1513491304174_0011/
Kill Command = /home/acadgild/hadoop-2.6.0/bin/hadoop job  -kill job_1513491304174_0011
Hadoop job information for Stage-2: number of mappers: 1; number of reducers: 1
2017-12-17 18:34:33,494 Stage-2 map = 0%,  reduce = 0%
2017-12-17 18:35:11,022 Stage-2 map = 100%,  reduce = 0%, Cumulative CPU 7.17 sec
2017-12-17 18:35:36,923 Stage-2 map = 100%,  reduce = 67%, Cumulative CPU 10.87 sec
2017-12-17 18:35:42,981 Stage-2 map = 100%,  reduce = 100%, Cumulative CPU 13.87 sec
MapReduce Total cumulative CPU time: 13 seconds 870 msec
Ended Job = job_1513491304174_0011
Launching Job 2 out of 4
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:



2017-12-17 18:36:42,384 Stage-3 map = 100%,  reduce = 0%, Cumulative CPU 7.25 sec
2017-12-17 18:37:05,146 Stage-3 map = 100%,  reduce = 100%, Cumulative CPU 11.08 sec
MapReduce Total cumulative CPU time: 11 seconds 250 msec
Ended Job = job_1513491304174_0012
Stage-6 is selected by condition resolver.
Stage-1 is filtered out by condition resolver.
FAILED: Execution Error, return code 1 from org.apache.hadoop.hive.ql.exec.mr.MapredLocalTask
ATTEMPT: Execute BackupTask: org.apache.hadoop.hive.ql.exec.mr.MapRedTask
Launching Job 4 out of 4
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_1513491304174_0013, Tracking URL = http://localhost:8088/proxy/application_1513491304174_0013/
Kill Command = /home/acadgild/hadoop-2.6.0/bin/hadoop job  -kill job_1513491304174_0013
Hadoop job information for Stage-1: number of mappers: 3; number of reducers: 1
2017-12-17 18:37:36,640 Stage-1 map = 0%,  reduce = 0%
2017-12-17 18:38:38,193 Stage-1 map = 0%,  reduce = 0%
2017-12-17 18:40:05,969 Stage-1 map = 0%,  reduce = 0%, Cumulative CPU 19.75 sec
2017-12-17 18:40:17,999 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 36.83 sec
2017-12-17 18:40:40,069 Stage-1 map = 100%,  reduce = 67%, Cumulative CPU 40.74 sec
2017-12-17 18:40:45,199 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 42.91 sec
MapReduce Total cumulative CPU time: 42 seconds 910 msec
Ended Job = job_1513491304174_0013
MapReduce Jobs Launched:
Stage-Stage-2: Map: 1  Reduce: 1   Cumulative CPU: 13.87 sec   HDFS Read: 8294 HDFS Write: 248 SUCCESS
Stage-Stage-3: Map: 1  Reduce: 1   Cumulative CPU: 11.25 sec   HDFS Read: 9528 HDFS Write: 293 SUCCESS
Stage-Stage-1: Map: 3  Reduce: 1   Cumulative CPU: 42.91 sec   HDFS Read: 24882 HDFS Write: 174 SUCCESS
Total MapReduce CPU Time Spent: 1 minutes 8 seconds 30 msec
OK
10      Siddharath
9       Emma
4       Arpitha
8       Vihan
Time taken: 405.366 seconds, Fetched: 4 row(s)
hive>
```