Project 2: Employee data analysis

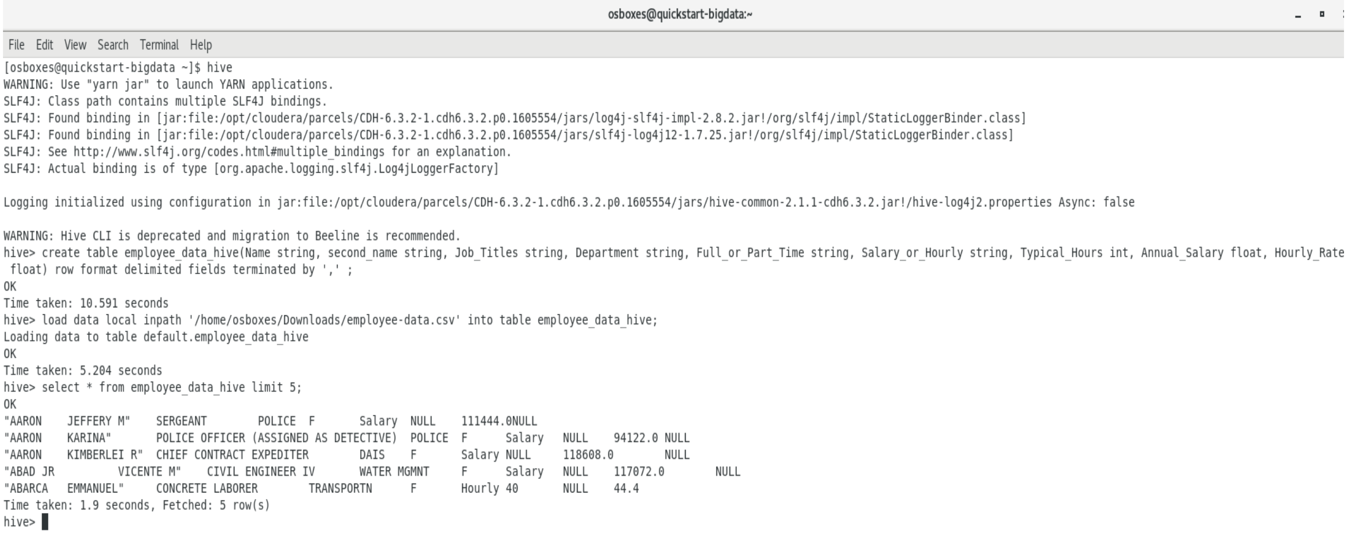
Group 10

Name	ID
Nourhan Abdelkerim	21nmma1
Ahmed Salem	21aaeh
Khaled Ahmed	21kaka
Mohamed Adam	21mrma

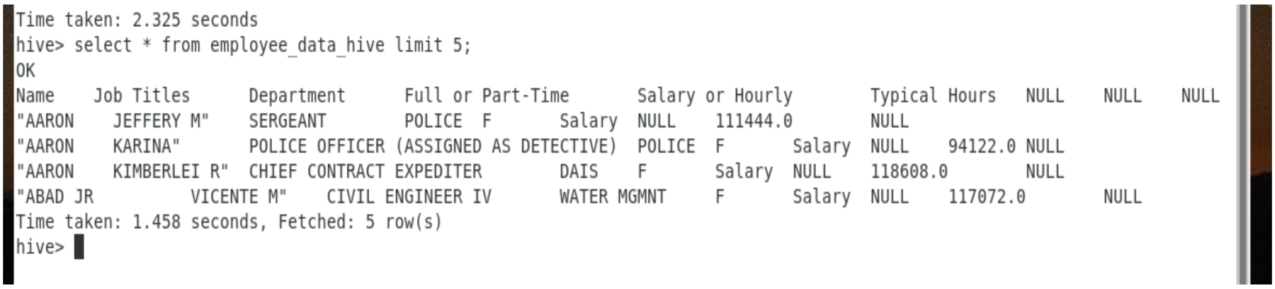
1. Create a Hive table named employee-data-hive based on the given dataset.

Sol:

- We open terminal then hive.
- We create table with the same columns in csv file.
- Import tale from local file on machine to hive system.
- Show first 5 rows.


osboxes@quickstart-bigdata:~
File Edit View Search Terminal Help

```
[osboxes@quickstart-bigdata ~]$ hive
WARNING: Use "yarn jar" to launch YARN applications.
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/opt/cloudera/parcels/CDH-6.3.2-1.cdh6.3.2.p0.1605554/jars/log4j-slf4j-impl-2.8.2.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/opt/cloudera/parcels/CDH-6.3.2-1.cdh6.3.2.p0.1605554/jars/slf4j-log4j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.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:/opt/cloudera/parcels/CDH-6.3.2-1.cdh6.3.2.p0.1605554/jars/hive-common-2.1.1-cdh6.3.2.jar!/hive-log4j2.properties Async: false

WARNING: Hive CLI is deprecated and migration to Beeline is recommended.
hive> create table employee_data_hive(Name string, second_name string, Job_Titles string, Department string, Full_or_Part_Time string, Salary_or_Hourly string, Typical_Hours int, Annual_Salary float, Hourly_Rate
 float) row format delimited fields terminated by ',' ;
OK
Time taken: 10.591 seconds
hive> load data local inpath '/home/osboxes/Downloads/employee-data.csv' into table employee_data_hive;
Loading data to table default.employee_data_hive
OK
Time taken: 5.204 seconds
hive> select * from employee_data_hive limit 5;
OK
"AARON    JEFFERY M"    SERGEANT        POLICE  F       Salary  NULL    111444.0NULL
"AARON    KARINA"       POLICE OFFICER (ASSIGNED AS DETECTIVE)  POLICE  F       Salary  NULL    94122.0 NULL
"AARON    KIMBERLEI R"  CHIEF CONTRACT EXPEDITER        DAIS    F       Salary NULL     118608.0        NULL
"ABAD JR          VICENTE M"    CIVIL ENGINEER IV       WATER MGMNT     F       Salary  NULL    117072.0        NULL
"ABARCA   EMMANUEL"     CONCRETE LABORER        TRANSPORTN      F       Hourly 40       NULL    44.4
Time taken: 1.9 seconds, Fetched: 5 row(s)
hive> █
```



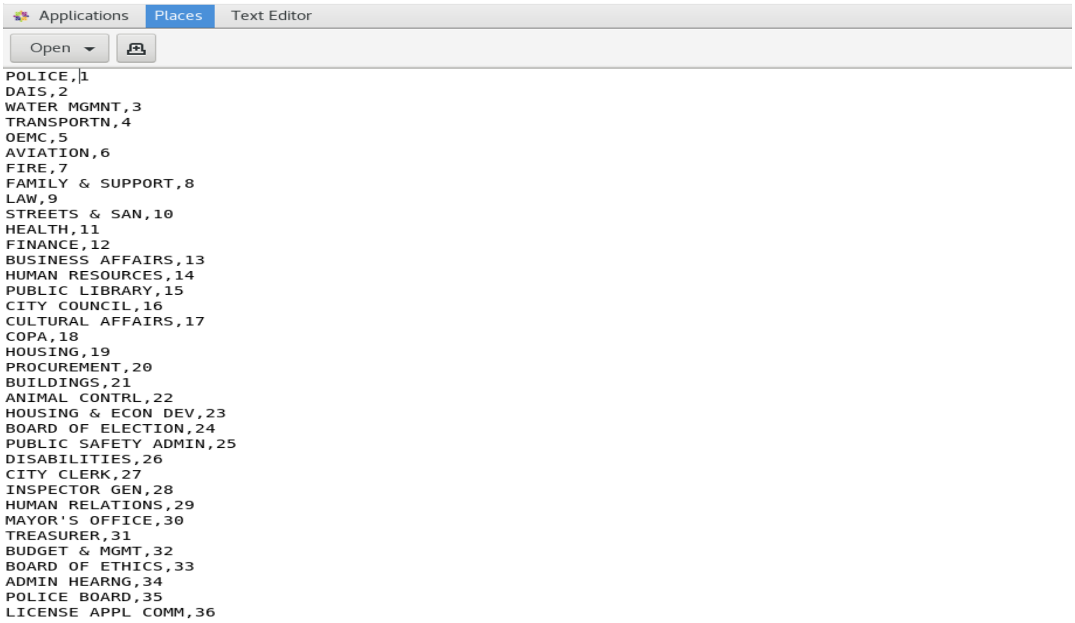
```
Time taken: 2.325 seconds
hive> select * from employee_data_hive limit 5;
OK
Name    Job Titles      Department      Full or Part-Time       Salary or Hourly        Typical Hours   NULL    NULL    NULL
"AARON    JEFFERY M"    SERGEANT        POLICE  F       Salary  NULL    111444.0        NULL
"AARON    KARINA"       POLICE OFFICER (ASSIGNED AS DETECTIVE)  POLICE  F       Salary  NULL    94122.0 NULL
"AARON    KIMBERLEI R"  CHIEF CONTRACT EXPEDITER        DAIS    F       Salary  NULL    118608.0        NULL
"ABAD JR          VICENTE M"    CIVIL ENGINEER IV       WATER MGMNT     F       Salary  NULL    117072.0        NULL
Time taken: 1.458 seconds, Fetched: 5 row(s)
hive> █
```

2. Create a department-data-hive table by selecting unique department names from the employee-data-hive and adding a column named deptID in the new department-data-hive table, and put unique values in the deptID column.

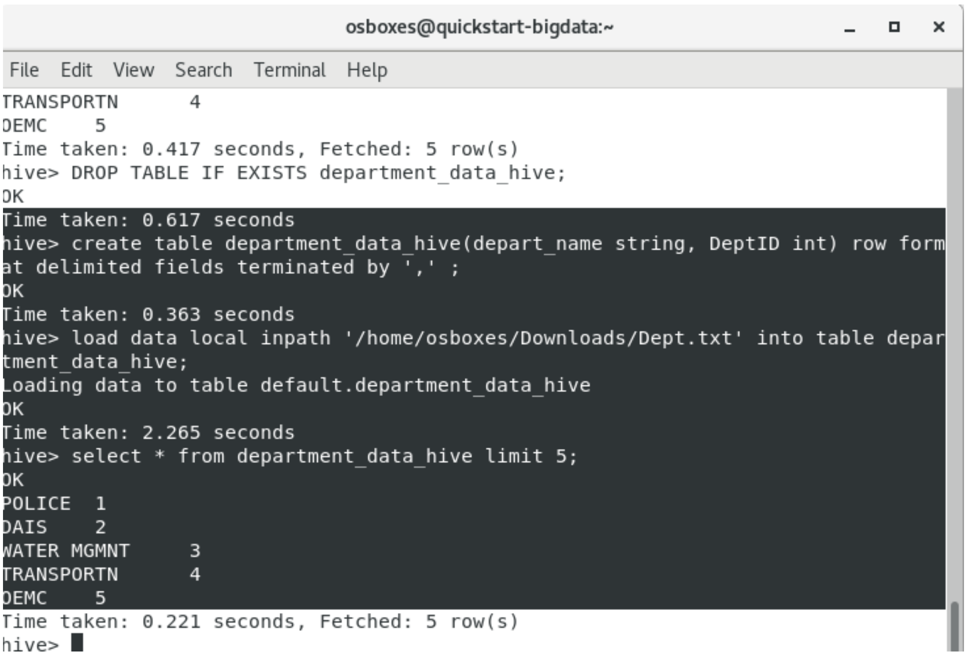
Alternatively, you can pre-process the employee-data and select the unique department names, add DeptID column and assign unique value in the new colum using excel or mySQL database separately, and then consider this structure (depart-name, DeptID) to create the department-data-hive table.

Sol.

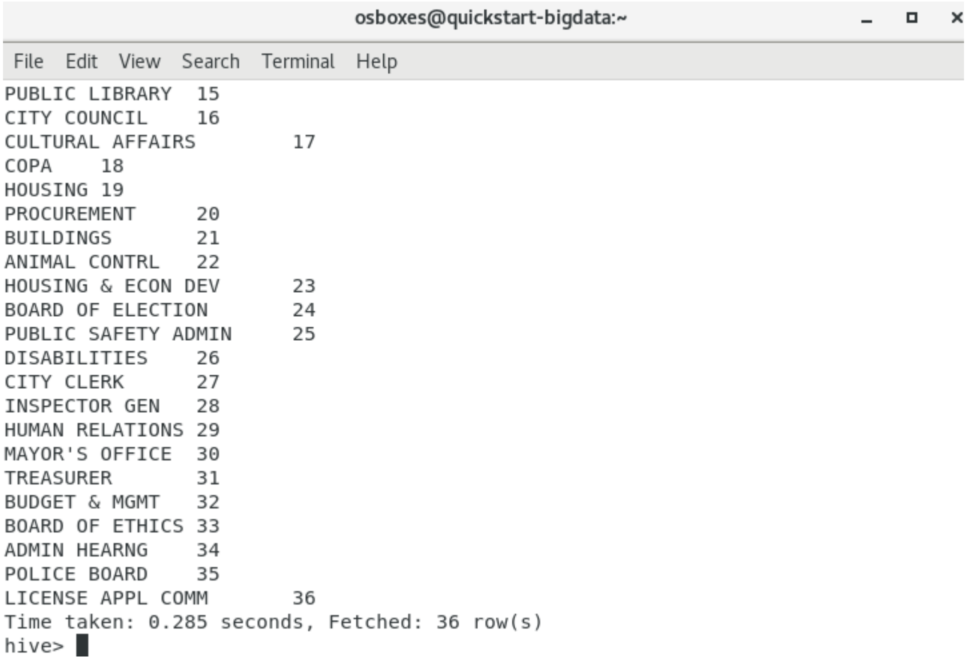
- We used excel to create new file.
- We selected unique values from department column and give each one a specific number.
- We moved file to local machine.
- We created department_data_hive table with the required columns.
- Load file into table.
- Show data inside table.
- As we see, we have 36 different department.



```
POLICE,1
DAIS,2
WATER MGMNT,3
TRANSPORTN,4
OEMC,5
AVIATION,6
FIRE,7
FAMILY & SUPPORT,8
LAW,9
STREETS & SAN,10
HEALTH,11
FINANCE,12
BUSINESS AFFAIRS,13
HUMAN RESOURCES,14
PUBLIC LIBRARY,15
CITY COUNCIL,16
CULTURAL AFFAIRS,17
COPA,18
HOUSING,19
PROCUREMENT,20
BUILDINGS,21
ANIMAL CONTRL,22
HOUSING & ECON DEV,23
BOARD OF ELECTION,24
PUBLIC SAFETY ADMIN,25
DISABILITIES,26
CITY CLERK,27
INSPECTOR GEN,28
HUMAN RELATIONS,29
MAYOR'S OFFICE,30
TREASURER,31
BUDGET & MGMT,32
BOARD OF ETHICS,33
ADMIN HEARNG,34
POLICE BOARD,35
LICENSE APPL COMM,36
```



```
TRANSPORTN      4
OEMC    5
Time taken: 0.417 seconds, Fetched: 5 row(s)
hive> DROP TABLE IF EXISTS department_data_hive;
OK
Time taken: 0.617 seconds
hive> create table department_data_hive(depart_name string, DeptID int) row form
at delimited fields terminated by ',' ;
OK
Time taken: 0.363 seconds
hive> load data local inpath '/home/osboxes/Downloads/Dept.txt' into table depar
tment_data_hive;
Loading data to table default.department_data_hive
OK
Time taken: 2.265 seconds
hive> select * from department_data_hive limit 5;
OK
POLICE  1
DAIS    2
WATER MGMNT     3
TRANSPORTN      4
OEMC    5
Time taken: 0.221 seconds, Fetched: 5 row(s)
hive> 
```



osboxes@quickstart-bigdata:~
File Edit View Search Terminal Help

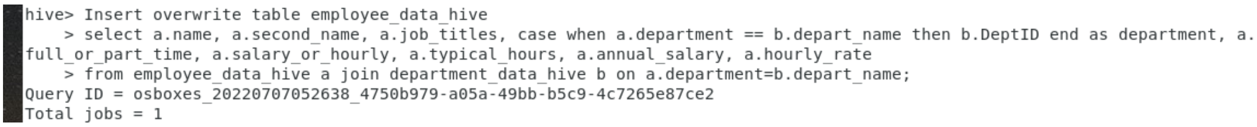
```
PUBLIC LIBRARY  15
CITY COUNCIL    16
CULTURAL AFFAIRS        17
COPA    18
HOUSING 19
PROCUREMENT     20
BUILDINGS       21
ANIMAL CONTRL   22
HOUSING & ECON DEV      23
BOARD OF ELECTION       24
PUBLIC SAFETY ADMIN     25
DISABILITIES    26
CITY CLERK      27
INSPECTOR GEN   28
HUMAN RELATIONS 29
MAYOR'S OFFICE  30
TREASURER       31
BUDGET & MGMT   32
BOARD OF ETHICS 33
ADMIN HEARNG    34
POLICE BOARD    35
LICENSE APPL COMM       36
Time taken: 0.285 seconds, Fetched: 36 row(s)
hive> 
```

3.

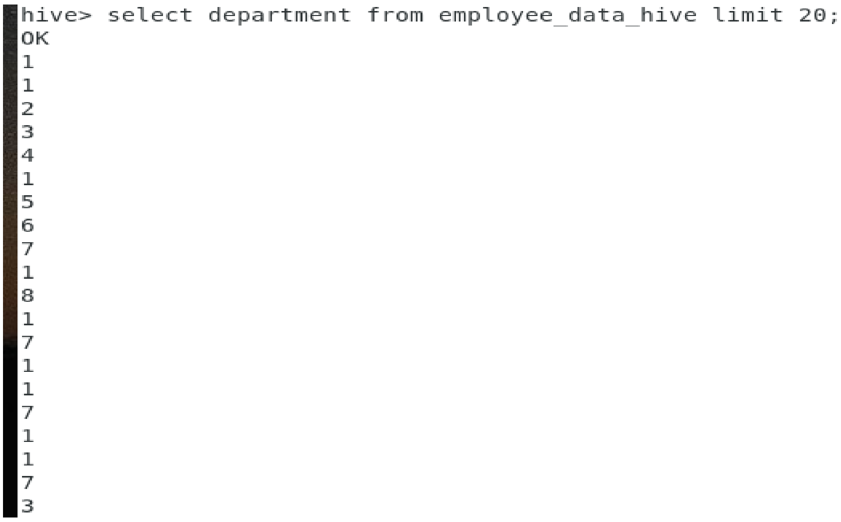
a. Update the employee-data-hive table by replacing the department field data with the deptID values as created in the department-data-hive table.

Sol.

- A full join was carried out between the 2 tables on department column in employees table and depart_name column in departments table.
- The needed columns were selected.



```
hive> Insert overwrite table employee_data_hive
    > select a.name, a.second_name, a.job_titles, case when a.department == b.depart_name then b.DeptID end as department, a.
full_or_part_time, a.salary_or_hourly, a.typical_hours, a.annual_salary, a.hourly_rate
    > from employee_data_hive a join department_data_hive b on a.department=b.depart_name;
Query ID = osboxes_20220707052638_4750b979-a05a-49bb-b5c9-4c7265e87ce2
Total jobs = 1
```



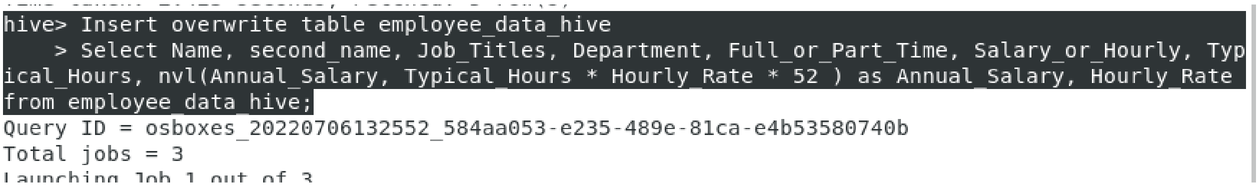
```
hive> select department from employee_data_hive limit 20;
OK
1
1
2
3
4
1
5
6
7
1
8
1
7
1
1
7
1
1
7
3
```

Here we selected the first 20 values from the department column in the employees table to make sure they were replaced with ID's.

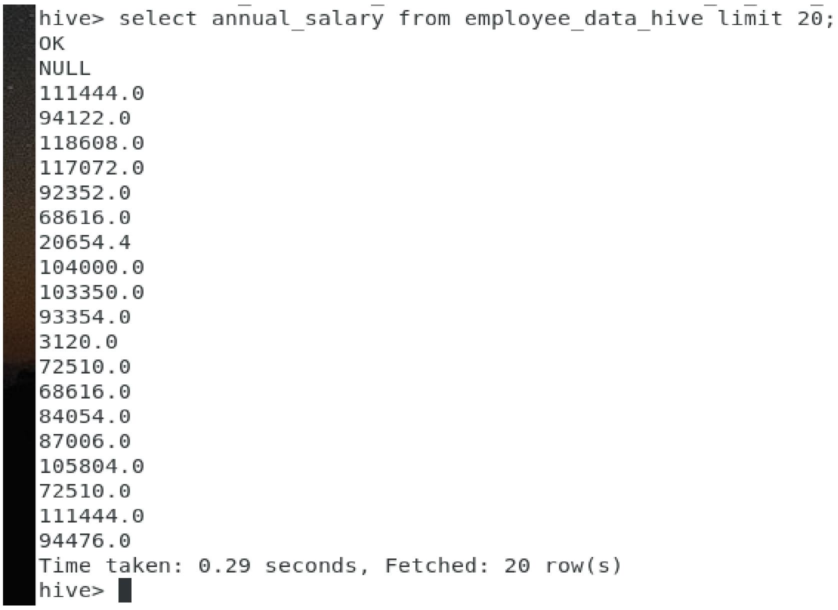
b. Also update the employee-data-hive table 'annual salary' field based on the 'Typical Hours' * 'Hourly Rate' * 52 if the annual salary field is empty.

Sol.

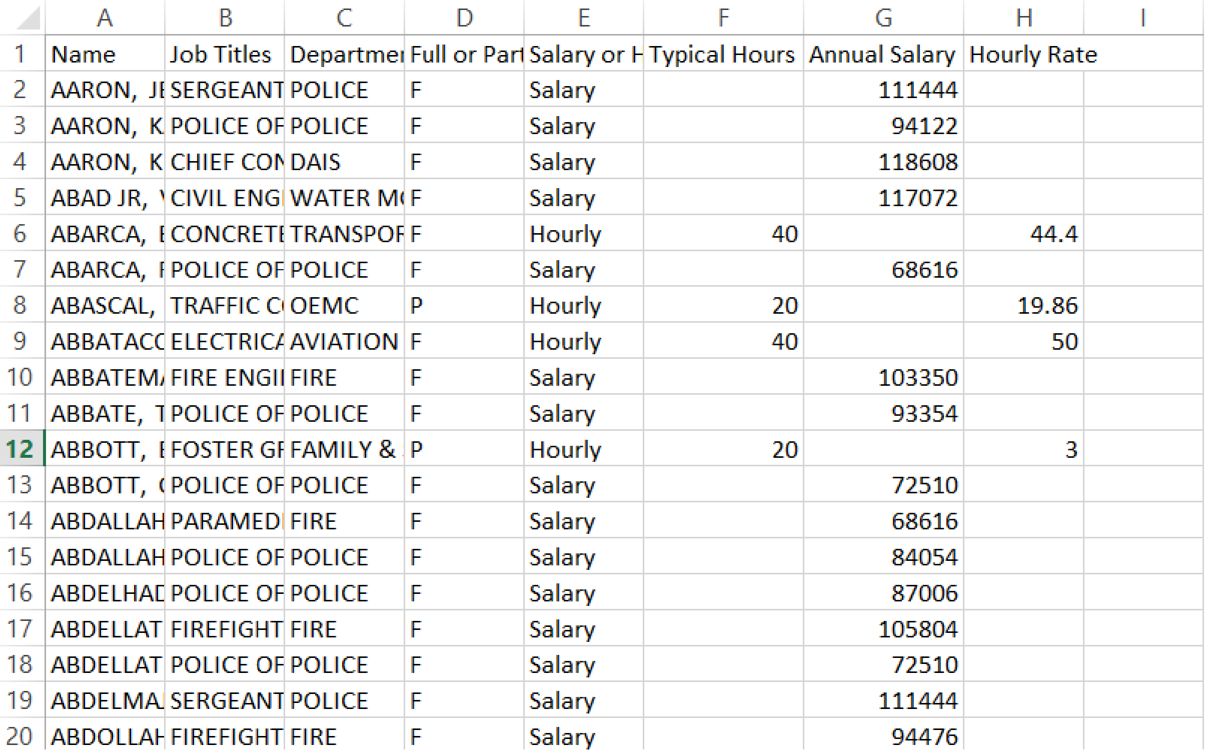
- We used insert overwrite to overwrite new data on annual salary column.
- We replaced nulls in this column with the value of (typical_hour* hourly_rate* 52).
- And leaved the rows that contain values as they are.



```
hive> Insert overwrite table employee_data_hive
    > Select Name, second_name, Job_Titles, Department, Full_or_Part_Time, Salary_or_Hourly, Typ
ical_Hours, nvl(Annual_Salary, Typical_Hours * Hourly_Rate * 52 ) as Annual_Salary, Hourly_Rate
from employee_data_hive;
Query ID = osboxes_20220706132552_584aa053-e235-489e-81ca-e4b53580740b
Total jobs = 3
Launching Job 1 out of 3
```



```
hive> select annual_salary from employee_data_hive limit 20;
OK
NULL
111444.0
94122.0
118608.0
117072.0
92352.0
68616.0
20654.4
104000.0
103350.0
93354.0
3120.0
72510.0
68616.0
84054.0
87006.0
105804.0
72510.0
111444.0
94476.0
Time taken: 0.29 seconds, Fetched: 20 row(s)
hive> █
```

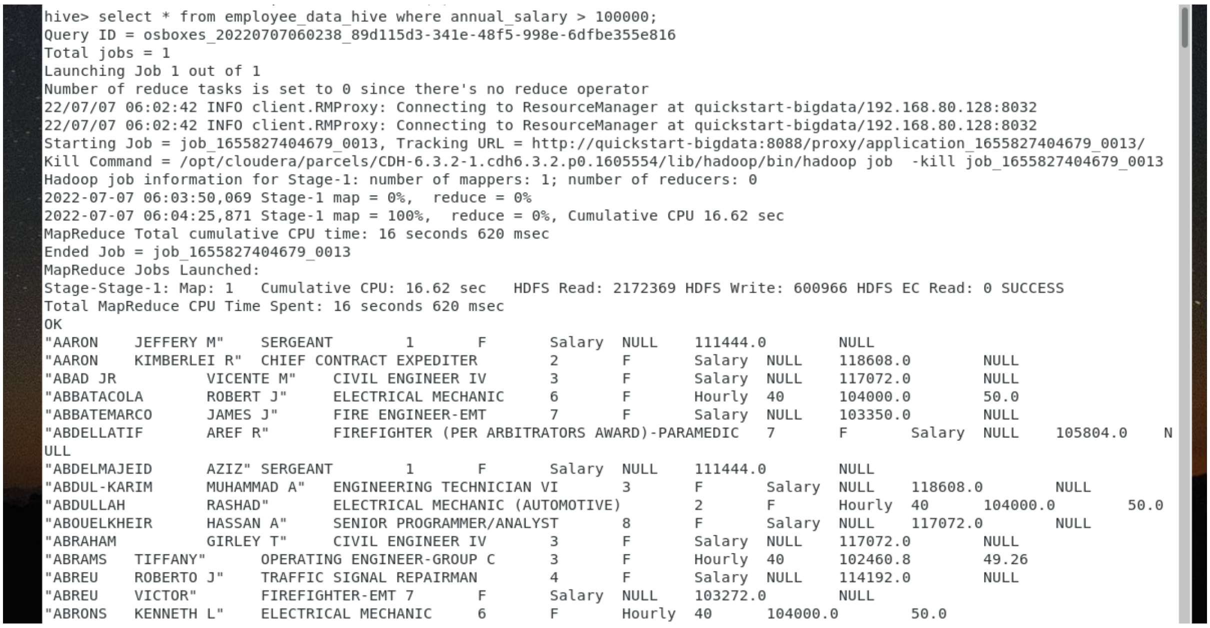


	A	B	C	D	E	F	G	H	I
1	Name	Job Titles	Departmei	Full or Part	Salary or H	Typical Hours	Annual Salary	Hourly Rate	
2	AARON, JI	SERGEANT	POLICE	F	Salary		111444		
3	AARON, K	POLICE OF	POLICE	F	Salary		94122		
4	AARON, K	CHIEF CON	DAIS	F	Salary		118608		
5	ABAD JR, \	CIVIL ENG	WATER M(	F	Salary		117072		
6	ABARCA, E	CONCRETE	TRANSPOF	F	Hourly	40		44.4	
7	ABARCA, F	POLICE OF	POLICE	F	Salary		68616		
8	ABASCAL,	TRAFFIC C	OEMC	P	Hourly	20		19.86	
9	ABBATACC	ELECTRICA	AVIATION	F	Hourly	40		50	
10	ABBATEMA	FIRE ENGII	FIRE	F	Salary		103350		
11	ABBATE, T	POLICE OF	POLICE	F	Salary		93354		
12	ABBOTT, E	FOSTER GF	FAMILY & .	P	Hourly	20		3	
13	ABBOTT, (	POLICE OF	POLICE	F	Salary		72510		
14	ABDALLAH	PARAMED	FIRE	F	Salary		68616		
15	ABDALLAH	POLICE OF	POLICE	F	Salary		84054		
16	ABDELHAD	POLICE OF	POLICE	F	Salary		87006		
17	ABDELLAT	FIREFIGHT	FIRE	F	Salary		105804		
18	ABDELLAT	POLICE OF	POLICE	F	Salary		72510		
19	ABDELMA	SERGEANT	POLICE	F	Salary		111444		
20	ABDOLLAH	FIREFIGHT	FIRE	F	Salary		94476		

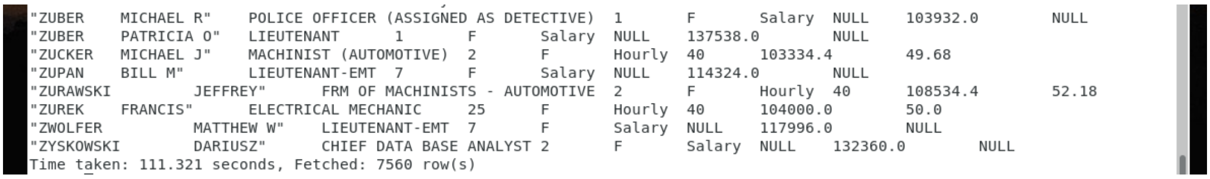
As we can see, the first 20 entries in the “Annual Salary” column had some missing values, but after using the above command and viewing the first 20 values, no null values were found. Note, the first value was null because it has the column name, but the first value is the same as in the excel screenshot.

4.

a. Display all employees list with salary more than $100,000 based on employee-data-hive table.



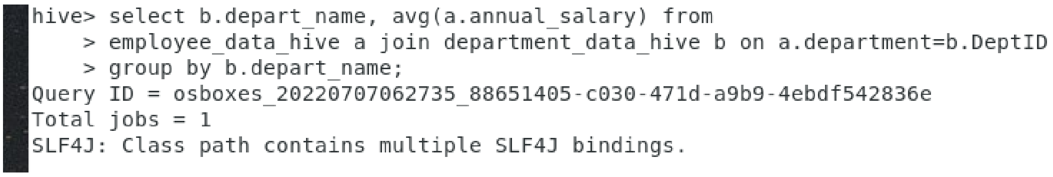
```
hive> select * from employee_data_hive where annual_salary > 100000;
Query ID = osboxes_20220707060238_89d115d3-341e-48f5-998e-6dfbe355e816
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks is set to 0 since there's no reduce operator
22/07/07 06:02:42 INFO client.RMProxy: Connecting to ResourceManager at quickstart-bigdata/192.168.80.128:8032
22/07/07 06:02:42 INFO client.RMProxy: Connecting to ResourceManager at quickstart-bigdata/192.168.80.128:8032
Starting Job = job_1655827404679_0013, Tracking URL = http://quickstart-bigdata:8088/proxy/application_1655827404679_0013/
Kill Command = /opt/cloudera/parcels/CDH-6.3.2-1.cdh6.3.2.p0.1605554/lib/hadoop/bin/hadoop job  -kill job_1655827404679_0013
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 0
2022-07-07 06:03:50,069 Stage-1 map = 0%,  reduce = 0%
2022-07-07 06:04:25,871 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 16.62 sec
MapReduce Total cumulative CPU time: 16 seconds 620 msec
Ended Job = job_1655827404679_0013
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1   Cumulative CPU: 16.62 sec   HDFS Read: 2172369 HDFS Write: 600966 HDFS EC Read: 0 SUCCESS
Total MapReduce CPU Time Spent: 16 seconds 620 msec
OK
"AARON    JEFFERY M"    SERGEANT        1       F       Salary  NULL    111444.0        NULL
"AARON    KIMBERLEI R"  CHIEF CONTRACT EXPEDITER        2       F       Salary  NULL    118608.0        NULL
"ABAD JR          VICENTE M"    CIVIL ENGINEER IV       3       F       Salary  NULL    117072.0        NULL
"ABBATACOLA       ROBERT J"     ELECTRICAL MECHANIC     6       F       Hourly  40      104000.0        50.0
"ABBATEMARCO      JAMES J"      FIRE ENGINEER-EMT       7       F       Salary  NULL    103350.0        NULL
"ABDELLATIF       AREF R"       FIREFIGHTER (PER ARBITRATORS AWARD)-PARAMEDIC   7       F       Salary  NULL    105804.0    N
ULL
"ABDELMAJEID      AZIZ" SERGEANT        1       F       Salary  NULL    111444.0        NULL
"ABDUL-KARIM      MUHAMMAD A"   ENGINEERING TECHNICIAN VI       3       F       Salary  NULL    118608.0        NULL
"ABDULLAH         RASHAD"       ELECTRICAL MECHANIC (AUTOMOTIVE)        2       F       Hourly  40      104000.0        50.0
"ABOUELKHEIR      HASSAN A"     SENIOR PROGRAMMER/ANALYST       8       F       Salary  NULL    117072.0        NULL
"ABRAHAM          GIRLEY T"     CIVIL ENGINEER IV       3       F       Salary  NULL    117072.0        NULL
"ABRAMS   TIFFANY"      OPERATING ENGINEER-GROUP C      3       F       Hourly  40      102460.8        49.26
"ABREU    ROBERTO J"    TRAFFIC SIGNAL REPAIRMAN        4       F       Salary  NULL    114192.0        NULL
"ABREU    VICTOR"       FIREFIGHTER-EMT 7       F       Salary  NULL    103272.0        NULL
"ABRONS   KENNETH L"    ELECTRICAL MECHANIC     6       F       Hourly  40      104000.0        50.0
```



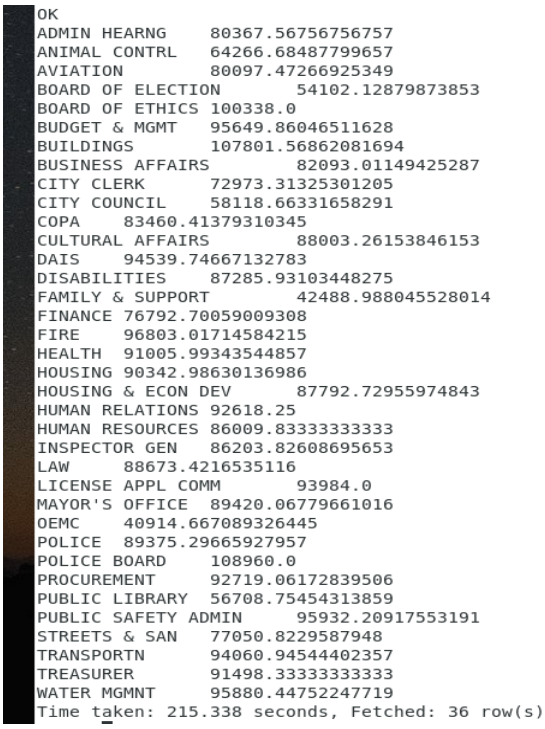
```
"ZUBER    MICHAEL R"    POLICE OFFICER (ASSIGNED AS DETECTIVE)  1       F       Salary  NULL    103932.0        NULL
"ZUBER    PATRICIA O"   LIEUTENANT      1       F       Salary  NULL    137538.0        NULL
"ZUCKER   MICHAEL J"    MACHINIST (AUTOMOTIVE)  2       F       Hourly  40      103334.4        49.68
"ZUPAN    BILL M"       LIEUTENANT-EMT  7       F       Salary  NULL    114324.0        NULL
"ZURAWSKI         JEFFREY"      FRM OF MACHINISTS - AUTOMOTIVE  2       F       Hourly  40      108534.4        52.18
"ZUREK    FRANCIS"      ELECTRICAL MECHANIC     25      F       Hourly  40      104000.0        50.0
"ZWOLFER          MATTHEW W"    LIEUTENANT-EMT  7       F       Salary  NULL    117996.0        NULL
"ZYSKOWSKI        DARIUSZ"      CHIEF DATA BASE ANALYST 2       F       Salary  NULL    132360.0        NULL
Time taken: 111.321 seconds, Fetched: 7560 row(s)
```

As we can see from the second screenshot, 7560 rows were selected.

b. join the employee-data-hive and department-data-hive table to show the average salary of employees by department name



```
hive> select b.depart_name, avg(a.annual_salary) from
    > employee_data_hive a join department_data_hive b on a.department=b.DeptID
    > group by b.depart_name;
Query ID = osboxes_20220707062735_88651405-c030-471d-a9b9-4ebdf542836e
Total jobs = 1
SLF4J: Class path contains multiple SLF4J bindings.
```



```
OK
ADMIN HEARNG    80367.56756756757
ANIMAL CONTRL   64266.68487799657
AVIATION        80097.47266925349
BOARD OF ELECTION       54102.12879873853
BOARD OF ETHICS 100338.0
BUDGET & MGMT   95649.86046511628
BUILDINGS       107801.56862081694
BUSINESS AFFAIRS        82093.01149425287
CITY CLERK      72973.31325301205
CITY COUNCIL    58118.66331658291
COPA    83460.41379310345
CULTURAL AFFAIRS        88003.26153846153
DAIS    94539.74667132783
DISABILITIES    87285.93103448275
FAMILY & SUPPORT        42488.988045528014
FINANCE 76792.70059009308
FIRE    96803.01714584215
HEALTH  91005.99343544857
HOUSING 90342.98630136986
HOUSING & ECON DEV      87792.72955974843
HUMAN RELATIONS 92618.25
HUMAN RESOURCES 86009.83333333333
INSPECTOR GEN   86203.82608695653
LAW     88673.4216535116
LICENSE APPL COMM       93984.0
MAYOR'S OFFICE  89420.06779661016
OEMC    40914.667089326445
POLICE  89375.29665927957
POLICE BOARD    108960.0
PROCUREMENT     92719.06172839506
PUBLIC LIBRARY  56708.75454313859
PUBLIC SAFETY ADMIN     95932.20917553191
STREETS & SAN   77050.8229587948
TRANSPORTN      94060.94544402357
TREASURER       91498.33333333333
WATER MGMNT     95880.44752247719
Time taken: 215.338 seconds, Fetched: 36 row(s)
```

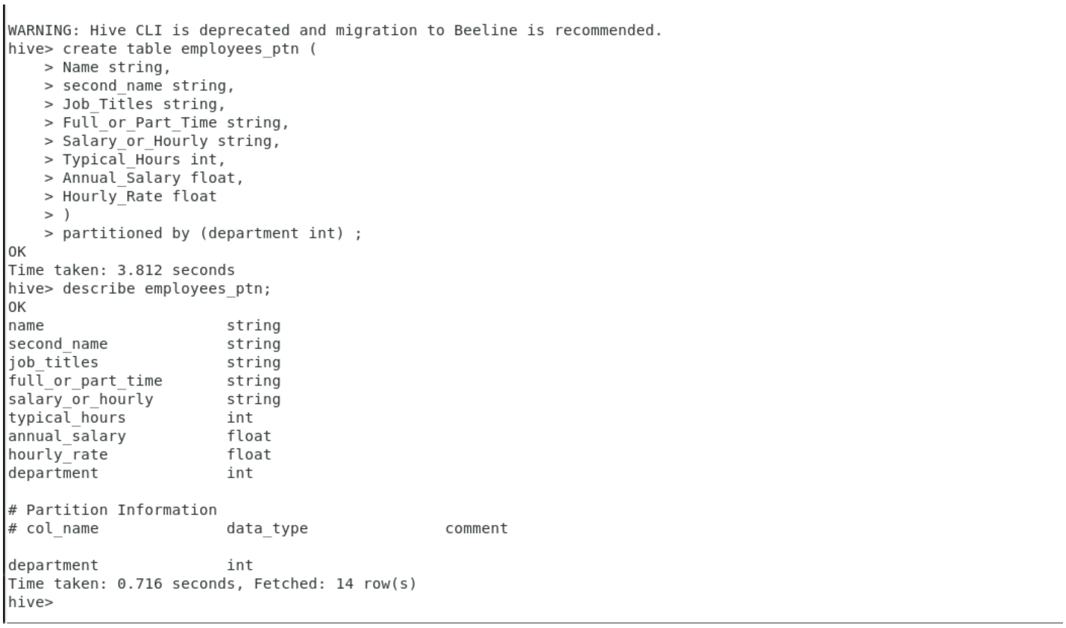
As shown in the previous screenshot, we have the average annual salary for each department name.

5.

a. Create 5 partitions in a employees_ptn table to store 5 departments in the appropriate partition.

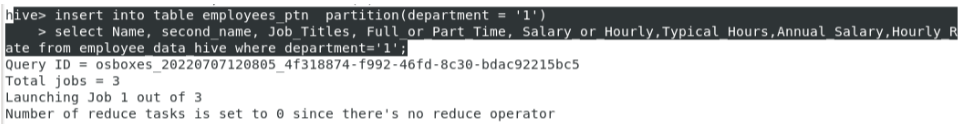
Sol.

- Create new table and partition by department.

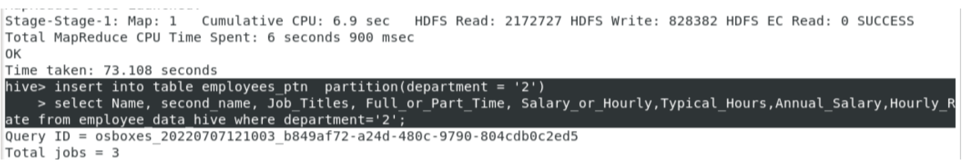


```
WARNING: Hive CLI is deprecated and migration to Beeline is recommended.
hive> create table employees_ptn (
    > Name string,
    > second_name string,
    > Job_Titles string,
    > Full_or_Part_Time string,
    > Salary_or_Hourly string,
    > Typical_Hours int,
    > Annual_Salary float,
    > Hourly_Rate float
    > )
    > partitioned by (department int) ;
OK
Time taken: 3.812 seconds
hive> describe employees_ptn;
OK
name                    string
second_name             string
job_titles              string
full_or_part_time       string
salary_or_hourly        string
typical_hours           int
annual_salary           float
hourly_rate             float
department              int

# Partition Information
# col_name              data_type               comment

department              int
Time taken: 0.716 seconds, Fetched: 14 row(s)
hive>
```

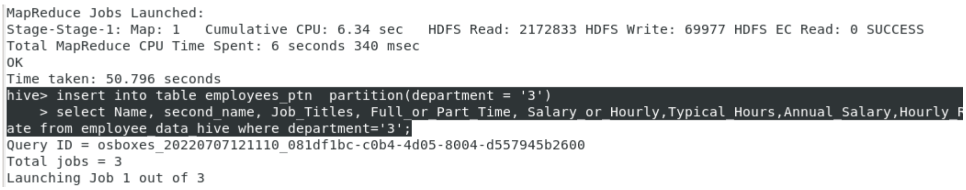
- Create partitions for first 5 department from 1 --> 5 as we converted department names to numbers



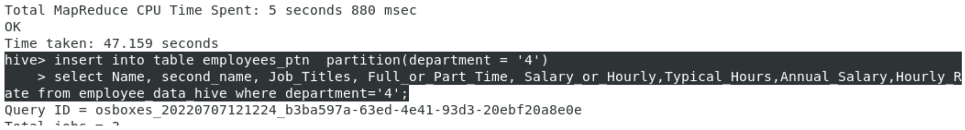
```
hive> insert into table employees_ptn  partition(department = '1')
    > select Name, second_name, Job_Titles, Full_or_Part_Time, Salary_or_Hourly,Typical_Hours,Annual_Salary,Hourly_R
ate from employee_data_hive where department='1';
Query ID = osboxes_20220707120805_4f318874-f992-46fd-8c30-bdac92215bc5
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks is set to 0 since there's no reduce operator
```



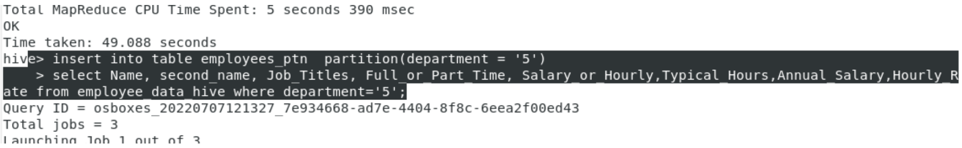
```
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1   Cumulative CPU: 6.9 sec   HDFS Read: 2172727 HDFS Write: 828382 HDFS EC Read: 0 SUCCESS
Total MapReduce CPU Time Spent: 6 seconds 900 msec
OK
Time taken: 73.108 seconds
hive> insert into table employees_ptn  partition(department = '2')
    > select Name, second_name, Job_Titles, Full_or_Part_Time, Salary_or_Hourly,Typical_Hours,Annual_Salary,Hourly_R
ate from employee_data_hive where department='2';
Query ID = osboxes_20220707121003_b849af72-a24d-480c-9790-804cdb0c2ed5
Total jobs = 3
```



```
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1   Cumulative CPU: 6.34 sec   HDFS Read: 2172833 HDFS Write: 69977 HDFS EC Read: 0 SUCCESS
Total MapReduce CPU Time Spent: 6 seconds 340 msec
OK
Time taken: 50.796 seconds
hive> insert into table employees_ptn  partition(department = '3')
    > select Name, second_name, Job_Titles, Full_or_Part_Time, Salary_or_Hourly,Typical_Hours,Annual_Salary,Hourly_R
ate from employee_data_hive where department='3';
Query ID = osboxes_20220707121110_081df1bc-c0b4-4d05-8004-d557945b2600
Total jobs = 3
Launching Job 1 out of 3
```

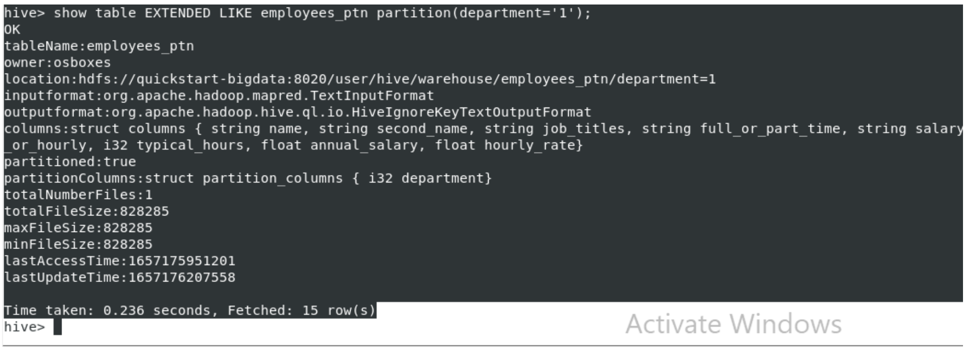


```
Total MapReduce CPU Time Spent: 5 seconds 880 msec
OK
Time taken: 47.159 seconds
hive> insert into table employees_ptn  partition(department = '4')
    > select Name, second_name, Job_Titles, Full_or_Part_Time, Salary_or_Hourly,Typical_Hours,Annual_Salary,Hourly_R
ate from employee_data_hive where department='4';
Query ID = osboxes_20220707121224_b3ba597a-63ed-4e41-93d3-20ebf20a8e0e
Total jobs = 3
```

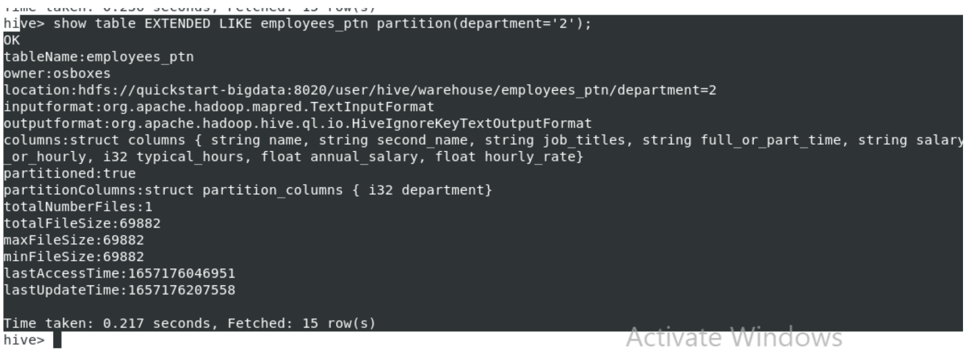


```
Total MapReduce CPU Time Spent: 5 seconds 390 msec
OK
Time taken: 49.088 seconds
hive> insert into table employees_ptn  partition(department = '5')
    > select Name, second_name, Job_Titles, Full_or_Part_Time, Salary_or_Hourly,Typical_Hours,Annual_Salary,Hourly_R
ate from employee_data_hive where department='5';
Query ID = osboxes_20220707121327_7e934668-ad7e-4404-8f8c-6eea2f00ed43
Total jobs = 3
Launching Job 1 out of 3
```

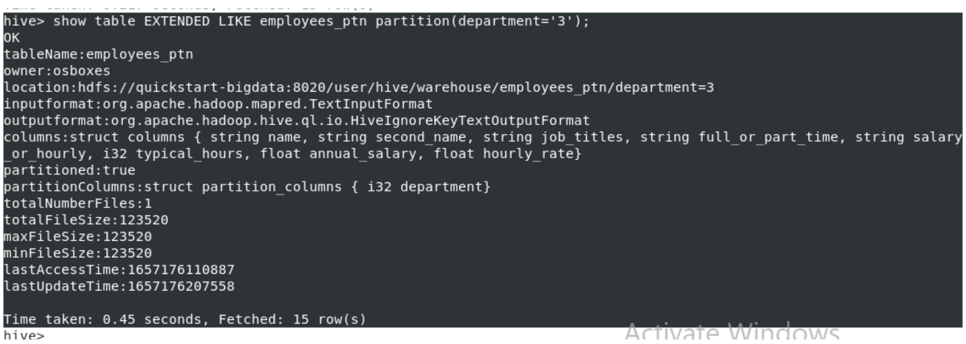
b. Display the partition structure.



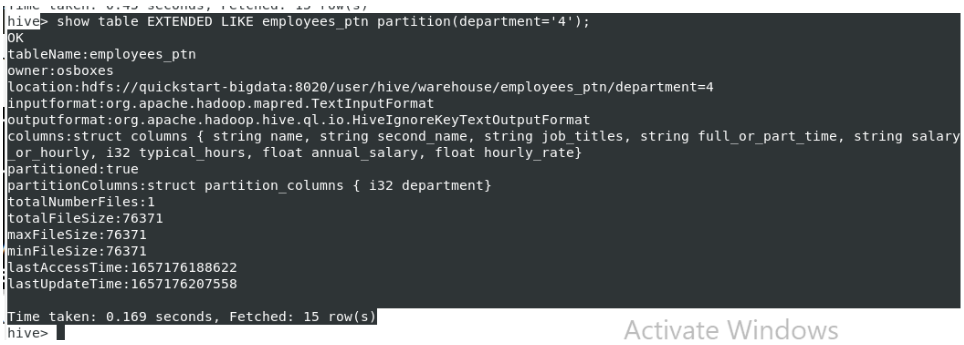
```
hive> show table EXTENDED LIKE employees_ptn partition(department='1');
OK
tableName:employees_ptn
owner:osboxes
location:hdfs://quickstart-bigdata:8020/user/hive/warehouse/employees_ptn/department=1
inputformat:org.apache.hadoop.mapred.TextInputFormat
outputformat:org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat
columns:struct columns { string name, string second_name, string job_titles, string full_or_part_time, string salary
_or_hourly, i32 typical_hours, float annual_salary, float hourly_rate}
partitioned:true
partitionColumns:struct partition_columns { i32 department}
totalNumberFiles:1
totalFileSize:828285
maxFileSize:828285
minFileSize:828285
lastAccessTime:1657175951201
lastUpdateTime:1657176207558

Time taken: 0.236 seconds, Fetched: 15 row(s)
hive>
```



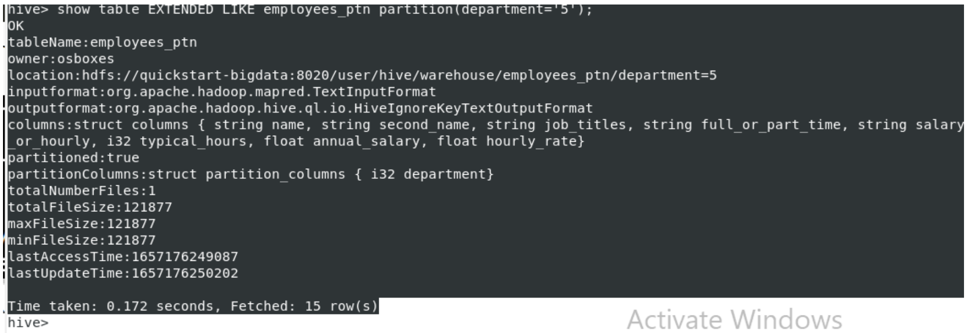
```
hive> show table EXTENDED LIKE employees_ptn partition(department='2');
OK
tableName:employees_ptn
owner:osboxes
location:hdfs://quickstart-bigdata:8020/user/hive/warehouse/employees_ptn/department=2
inputformat:org.apache.hadoop.mapred.TextInputFormat
outputformat:org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat
columns:struct columns { string name, string second_name, string job_titles, string full_or_part_time, string salary
_or_hourly, i32 typical_hours, float annual_salary, float hourly_rate}
partitioned:true
partitionColumns:struct partition_columns { i32 department}
totalNumberFiles:1
totalFileSize:69882
maxFileSize:69882
minFileSize:69882
lastAccessTime:1657176046951
lastUpdateTime:1657176207558

Time taken: 0.217 seconds, Fetched: 15 row(s)
hive>
```



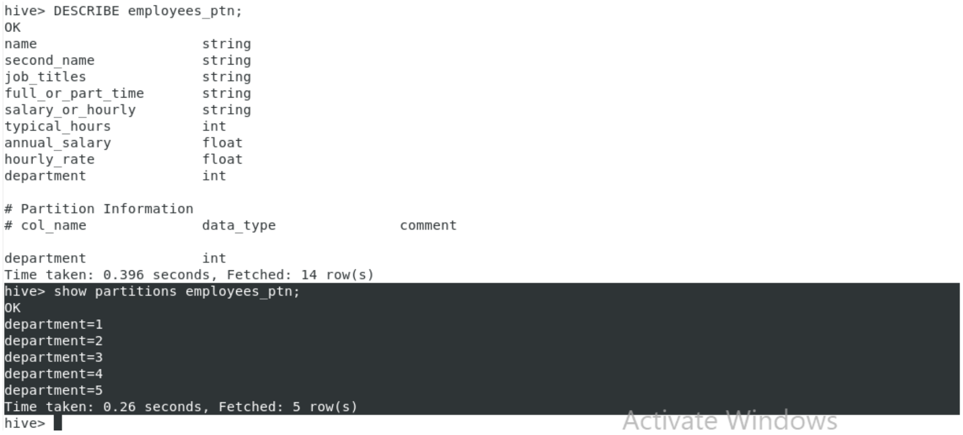
```
hive> show table EXTENDED LIKE employees_ptn partition(department='3');
OK
tableName:employees_ptn
owner:osboxes
location:hdfs://quickstart-bigdata:8020/user/hive/warehouse/employees_ptn/department=3
inputformat:org.apache.hadoop.mapred.TextInputFormat
outputformat:org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat
columns:struct columns { string name, string second_name, string job_titles, string full_or_part_time, string salary
_or_hourly, i32 typical_hours, float annual_salary, float hourly_rate}
partitioned:true
partitionColumns:struct partition_columns { i32 department}
totalNumberFiles:1
totalFileSize:123520
maxFileSize:123520
minFileSize:123520
lastAccessTime:1657176110887
lastUpdateTime:1657176207558

Time taken: 0.45 seconds, Fetched: 15 row(s)
hive>
```



```
hive> show table EXTENDED LIKE employees_ptn partition(department='4');
OK
tableName:employees_ptn
owner:osboxes
location:hdfs://quickstart-bigdata:8020/user/hive/warehouse/employees_ptn/department=4
inputformat:org.apache.hadoop.mapred.TextInputFormat
outputformat:org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat
columns:struct columns { string name, string second_name, string job_titles, string full_or_part_time, string salary
_or_hourly, i32 typical_hours, float annual_salary, float hourly_rate}
partitioned:true
partitionColumns:struct partition_columns { i32 department}
totalNumberFiles:1
totalFileSize:76371
maxFileSize:76371
minFileSize:76371
lastAccessTime:1657176188622
lastUpdateTime:1657176207558

Time taken: 0.169 seconds, Fetched: 15 row(s)
hive>
```



```
hive> show table EXTENDED LIKE employees_ptn partition(department='5');
OK
tableName:employees_ptn
owner:osboxes
location:hdfs://quickstart-bigdata:8020/user/hive/warehouse/employees_ptn/department=5
inputformat:org.apache.hadoop.mapred.TextInputFormat
outputformat:org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat
columns:struct columns { string name, string second_name, string job_titles, string full_or_part_time, string salary
_or_hourly, i32 typical_hours, float annual_salary, float hourly_rate}
partitioned:true
partitionColumns:struct partition_columns { i32 department}
totalNumberFiles:1
totalFileSize:121877
maxFileSize:121877
minFileSize:121877
lastAccessTime:1657176249087
lastUpdateTime:1657176250202

Time taken: 0.172 seconds, Fetched: 15 row(s)
hive>
```



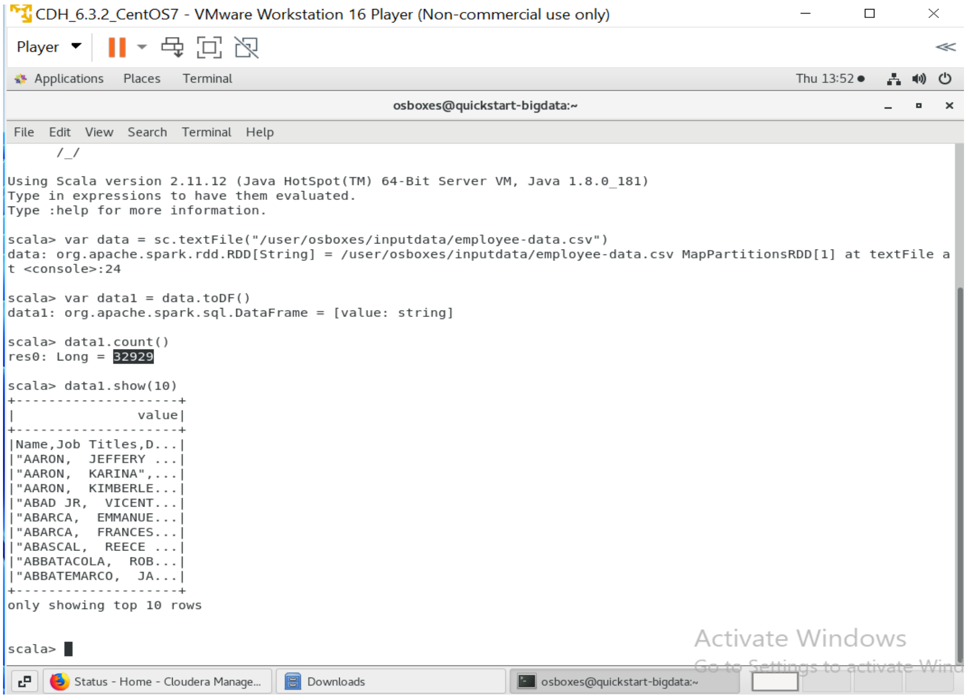
```
hive> DESCRIBE employees_ptn;
OK
name                    string
second_name             string
job_titles              string
full_or_part_time       string
salary_or_hourly        string
typical_hours           int
annual_salary           float
hourly_rate             float
department              int

# Partition Information
# col_name              data_type               comment

department              int
Time taken: 0.396 seconds, Fetched: 14 row(s)
hive> show partitions employees_ptn;
OK
department=1
department=2
department=3
department=4
department=5
Time taken: 0.26 seconds, Fetched: 5 row(s)
hive> 
```

As we see, we have 5 partitions contain 5 departments.

6. Create spark DataFrame based on the given dataset. Identify # of records in the DataFrame and show top 10 records.

Sol.

- At the first we copied data from local to hdfs.
- Opened spark-shell.
- We used sc.textfile to read data from hdfs.
- We converted text data to dataframe.
- As we see, we have 32929 records.
- At the end we displayed the first 10 records of dataframe.


CDH_6.3.2_CentOS7 - VMware Workstation 16 Player (Non-commercial use only)
Player
Applications
Places
Terminal
Thu 13:52
osboxes@quickstart-bigdata:~
File
Edit
View
Search
Terminal
Help
Status - Home - Cloudera Manage...
Downloads
osboxes@quickstart-bigdata:~

```
      /_/

Using Scala version 2.11.12 (Java HotSpot(TM) 64-Bit Server VM, Java 1.8.0_181)
Type in expressions to have them evaluated.
Type :help for more information.

scala> var data = sc.textFile("/user/osboxes/inputdata/employee-data.csv")
data: org.apache.spark.rdd.RDD[String] = /user/osboxes/inputdata/employee-data.csv MapPartitionsRDD[1] at textFile a
t <console>:24

scala> var data1 = data.toDF()
data1: org.apache.spark.sql.DataFrame = [value: string]

scala> data1.count()
res0: Long = 32929

scala> data1.show(10)
+--------------------+
|               value|
+--------------------+
|Name,Job Titles,D...|
|"AARON,  JEFFERY ...|
|"AARON,  KARINA",...|
|"AARON,  KIMBERLE...|
|"ABAD JR,  VICENT...|
|"ABARCA,  EMMANUE...|
|"ABARCA,  FRANCES...|
|"ABASCAL,  REECE ...|
|"ABBATACOLA,  ROB...|
|"ABBATEMARCO,  JA...|
+--------------------+
only showing top 10 rows

scala> 
```

- **Workload:**

Member	Steps
Nourhan Abdelkerim	1, 2, 3.b
Khaled Ahmed	3.a, 4
Ahmed Salem	5
Mohamed Adam	6