Gupta Bhandari

Student id: c0899873

To analyze the given data from a **"Realestate.csv"** file, I first created a database namely **"assignments"**. Then, I created a table **"real_estate_records"** with fields: Street, city, zip, state, beds, bath, sq_ft, type, sale_ date, price, latitude, longitude to store the data from **"Realestate.csv"** file. After that, I loaded the data from **"Realestate.csv"** file to the **"real_estate_records"** table. All these steps are reflected by a snapshot below:

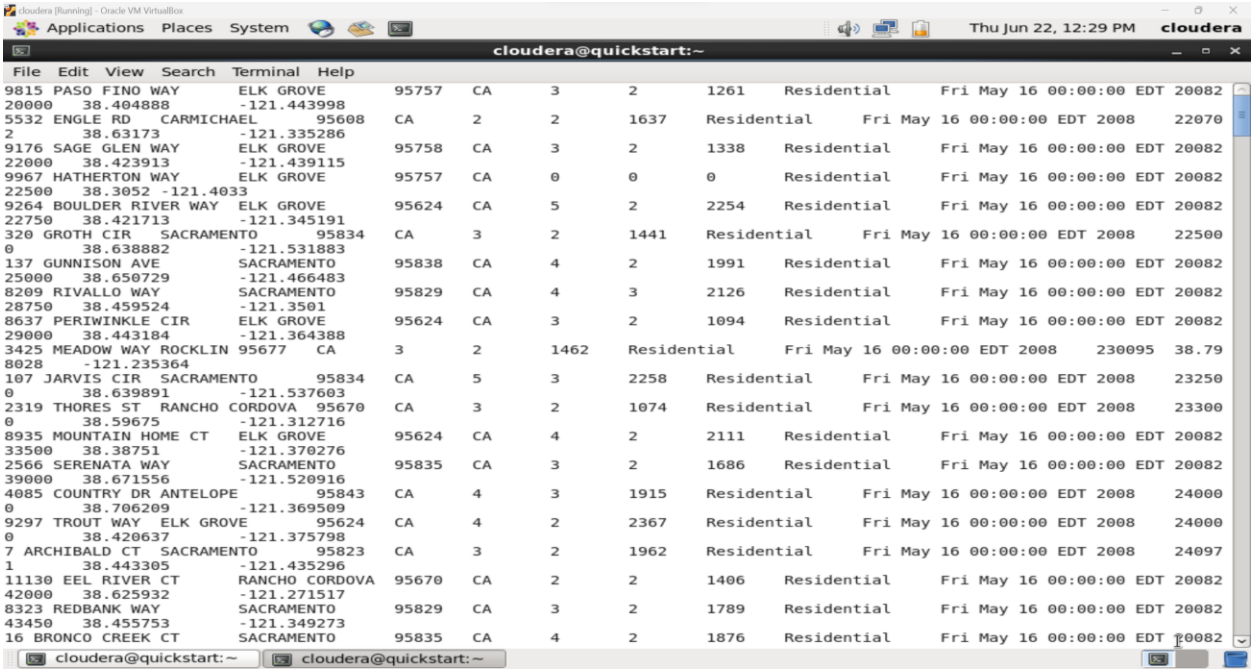
```
hive> use assignments;
OK
Time taken: 0.157 seconds
hive> set hive.cli.print.current.db=True
    > ;
hive (assignments)> show tables
                  > ;
OK
Time taken: 0.425 seconds
hive (assignments)> create table real_estate_records(street string, city string, zip int, state string, beds int, bath int, s
q_ft int, type string, sale_date string, price int, latitude double, longitude double)
                  > row format delimited
                  > fields terminated by ','
                  > TBLPROPERTIES("skip.header.line.count" = "1");
OK
Time taken: 0.417 seconds
hive (assignments)> describe real_estate_records;
OK
street                  string
city                    string
zip                     int
state                   string
beds                    int
bath                    int
sq_ft                   int
type                    string
sale_date               string
price                   int
latitude                double
longitude               double
Time taken: 0.342 seconds, Fetched: 12 row(s)
hive (assignments)> load data local inpath 'Realestate.csv' into real_estate_records;
FAILED: ParseException line 1:45 missing TABLE at 'real_estate_records' near '<EOF>'
hive (assignments)> load data local inpath 'Realestate.csv' into table real_estate_records;
Loading data to table assignments.real_estate_records
Table assignments.real_estate_records stats: [numFiles=1, totalSize=113183]
OK
Time taken: 0.979 seconds
hive (assignments)>
```

After this, I processed the problem statements to analyze data as listed below:

1. **<u>List all the residential which is not less than 10,000:</u>**
   For this problem statement, I ran a query in HQL(Hive Query Language) as : **select * from real_estate_records where type = "Residential" and price >= 10000;** as shown in snapshot below:

```
hive (assignments)> select * from real_estate_records where type = "Residential" and price >= 10000;
```

   After running above mentioned query, I got all the Residentials whose price is more than or quals to 10,000 as shown in snapshot below:



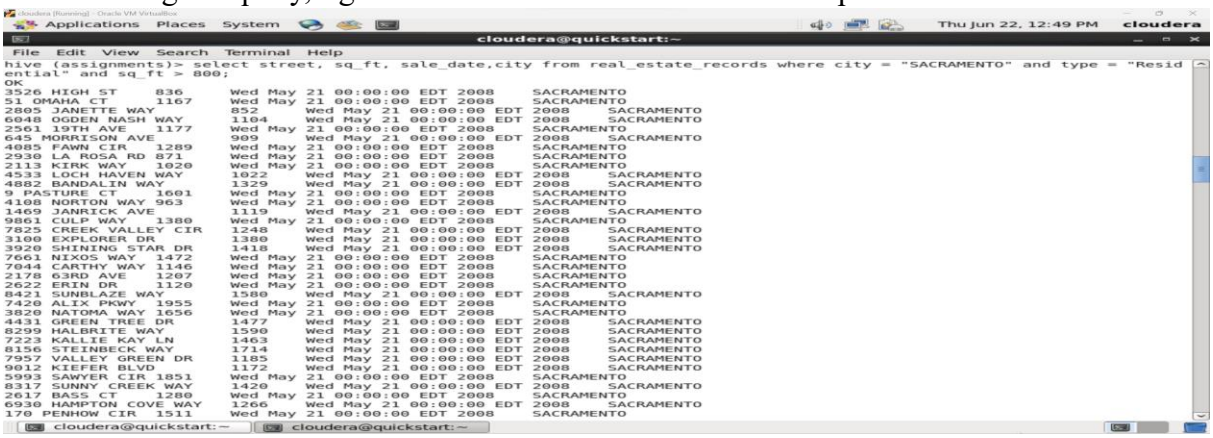
```
9815 PASO FINO WAY       ELK GROVE        95757   CA      3       2       1261    Residential     Fri May 16 00:00:00 EDT 20082
20000   38.404888        -121.443998
5532 ENGLE RD   CARMICHAEL       95608   CA      2       2       1637    Residential     Fri May 16 00:00:00 EDT 2008    22070
2       38.63173         -121.335286
9176 SAGE GLEN WAY       ELK GROVE        95758   CA      3       2       1338    Residential     Fri May 16 00:00:00 EDT 20082
22000   38.423913        -121.439115
9967 HATHERTON WAY       ELK GROVE        95757   CA      0       0       0       Residential     Fri May 16 00:00:00 EDT 20082
22500   38.3052 -121.4033
9264 BOULDER RIVER WAY   ELK GROVE        95624   CA      5       2       2254    Residential     Fri May 16 00:00:00 EDT 20082
22750   38.421713        -121.345191
320 GROTH CIR   SACRAMENTO       95834   CA      3       2       1441    Residential     Fri May 16 00:00:00 EDT 2008    22500
0       38.638882        -121.531883
137 GUNNISON AVE         SACRAMENTO       95838   CA      4       2       1991    Residential     Fri May 16 00:00:00 EDT 20082
25000   38.650729        -121.466483
8209 RIVALLO WAY         SACRAMENTO       95829   CA      4       3       2126    Residential     Fri May 16 00:00:00 EDT 20082
28750   38.459524        -121.3501
8637 PERIWINKLE CIR      ELK GROVE        95624   CA      3       2       1094    Residential     Fri May 16 00:00:00 EDT 20082
29000   38.443184        -121.364388
3425 MEADOW WAY ROCKLIN 95677   CA      3       2       1462    Residential     Fri May 16 00:00:00 EDT 2008    230095  38.79
8028    -121.235364
107 JARVIS CIR  SACRAMENTO       95834   CA      5       3       2258    Residential     Fri May 16 00:00:00 EDT 2008    23250
0       38.639891        -121.537603
2319 THORES ST  RANCHO CORDOVA  95670   CA      3       2       1074    Residential     Fri May 16 00:00:00 EDT 2008    23300
0       38.59675         -121.312716
8935 MOUNTAIN HOME CT    ELK GROVE        95624   CA      4       2       2111    Residential     Fri May 16 00:00:00 EDT 20082
33500   38.38751         -121.370276
2566 SERENATA WAY        SACRAMENTO       95835   CA      3       2       1686    Residential     Fri May 16 00:00:00 EDT 20082
39000   38.671556        -121.520916
4085 COUNTRY DR ANTELOPE         95843   CA      4       3       1915    Residential     Fri May 16 00:00:00 EDT 2008    24000
0       38.706209        -121.369509
9297 TROUT WAY  ELK GROVE        95624   CA      4       2       2367    Residential     Fri May 16 00:00:00 EDT 2008    24000
0       38.420637        -121.375798
7 ARCHIBALD CT  SACRAMENTO       95823   CA      3       2       1962    Residential     Fri May 16 00:00:00 EDT 2008    24097
1       38.443305        -121.435296
11130 EEL RIVER CT       RANCHO CORDOVA  95670   CA      2       2       1406    Residential     Fri May 16 00:00:00 EDT 20082
42000   38.625932        -121.271517
8323 REDBANK WAY         SACRAMENTO       95829   CA      3       2       1789    Residential     Fri May 16 00:00:00 EDT 20082
43450   38.455753        -121.349273
16 BRONCO CREEK CT       SACRAMENTO       95835   CA      4       2       1876    Residential     Fri May 16 00:00:00 EDT 20082
```

2. **<u>In SACRAMENTO city which residential type has more than 800sq ft. Display their respective details street, sq_ft, sale_date, city:</u>**

To perform this analysis, I ran a query as: **select street, sq_ft, sale_date, city from real_estate_records where city = "SECRAMENTO" and type = "Residential" and sq_ft>800;** as shown below:

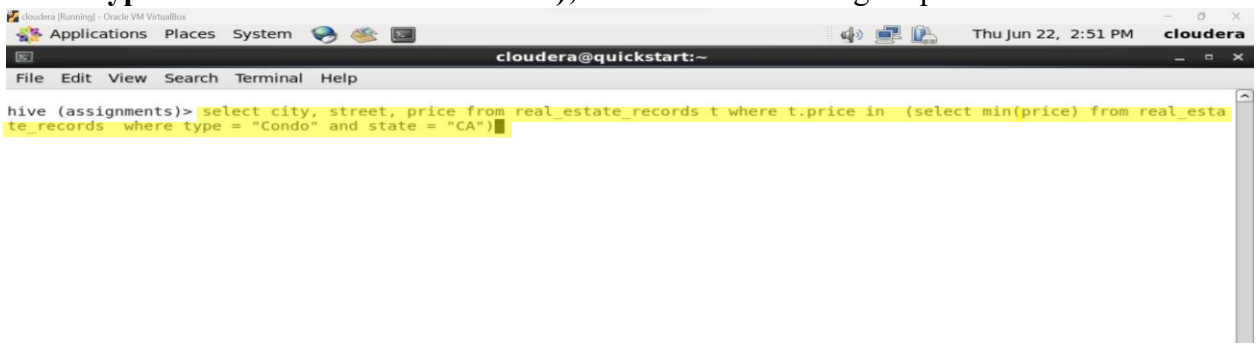
```
hive (assignments)> select street, sq_ft, sale_date,city from real_estate_records where city = "SACRAMENTO" and type = "Resid
ential" and sq_ft > 800;
```

After running the query, I got the desired result as shown in a snapshot below:

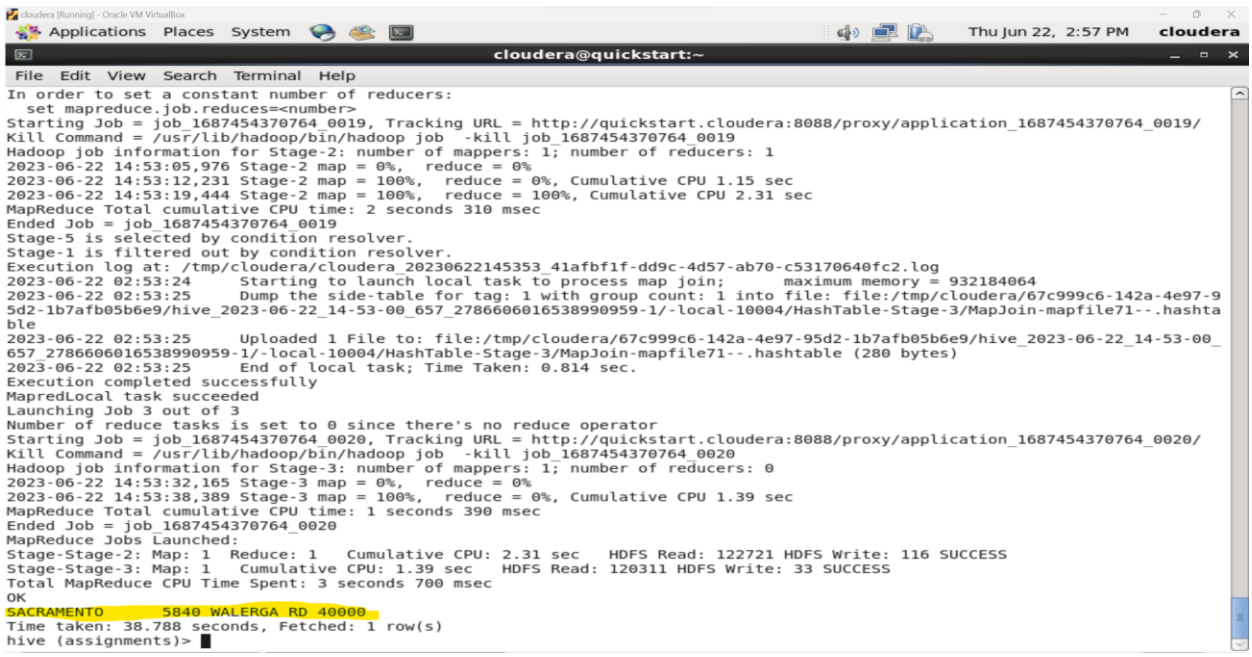


```
hive (assignments)> select street, sq_ft, sale_date,city from real_estate_records where city = "SACRAMENTO" and type = "Resid
ential" and sq_ft > 800;
OK
3526 HIGH ST     836     Wed May 21 00:00:00 EDT 2008    SACRAMENTO
51 OMAHA CT      1167    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
2805 JANETTE WAY         852     Wed May 21 00:00:00 EDT 2008    SACRAMENTO
6048 OGDEN NASH WAY      1104    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
2561 19TH AVE    1177    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
645 MORRISON AVE         909     Wed May 21 00:00:00 EDT 2008    SACRAMENTO
4085 FAWN CIR    1289    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
2930 LA ROSA RD 871      Wed May 21 00:00:00 EDT 2008    SACRAMENTO
2113 KIRK WAY    1020    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
4533 LOCH HAVEN WAY      1022    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
4882 BANDALIN WAY        1329    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
9 PASTURE CT     1601    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
4108 NORTON WAY 963      Wed May 21 00:00:00 EDT 2008    SACRAMENTO
1469 JANRICK AVE         1119    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
9861 CULP WAY    1380    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
7825 CREEK VALLEY CIR    1248    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
3100 EXPLORER DR         1380    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
3920 SHINING STAR DR     1418    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
7661 NIXOS WAY   1472    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
7044 CARTHY WAY 1146     Wed May 21 00:00:00 EDT 2008    SACRAMENTO
2178 63RD AVE    1207    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
2622 ERIN DR     1120    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
8421 SUNBLAZE WAY        1580    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
7420 ALIX PKWY   1955    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
3820 NATOMA WAY 1656     Wed May 21 00:00:00 EDT 2008    SACRAMENTO
4431 GREEN TREE DR       1477    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
8299 HALBRITE WAY        1590    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
7223 KALLIE KAY LN       1463    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
8156 STEINBECK WAY       1714    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
7957 VALLEY GREEN DR     1185    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
9012 KIEFER BLVD         1172    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
5993 SAWYER CIR 1851     Wed May 21 00:00:00 EDT 2008    SACRAMENTO
8317 SUNNY CREEK WAY     1420    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
2617 BASS CT     1280    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
6930 HAMPTON COVE WAY    1266    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
170 PENHOW CIR   1511    Wed May 21 00:00:00 EDT 2008    SACRAMENTO
```

3. **<u>Which is the cheapest Condo in CA. name the city, street, and price for the Condo:</u>**
To perform this analysis, I ran a HQL query: **select city, street, price from real_estate_records t where t.price in (select min(price) from real_estate_records where type = "Condo" and state = "CA");** as shown in following snapshot:



After executing the above mentioned HQL query, I got a result as reflected by snapshot below:



4. **<u>List top 5 residency details which lie in the budget of 50000-100000, an area more than 1250, min bedroom 3 and, min bathroom 2:</u>**
For this analysis, I ran a HQL query as: **select * from real_estate_records where price between 50000 and 100000**
and sq_ft > 1250
and beds >= 3
and bath >= 2
limit 5;
and got a result as shown below in screenshot:

```
hive (assignments)> select * from real_estate_records where price between 50000 and 100000
                  > and sq_ft > 1250
                  > and beds >=3
                  > and bath>=2
                  > limit 5;
OK
483 ARCADE BLVD SACRAMENTO      95815   CA      4       2       1316    Residential     Tue May 20 00:00:00 EDT 2008    89000
38.623571       -121.454884
2820 DEL PASO BLVD      SACRAMENTO      95815   CA      4       2       1404    Multi-Family    Mon May 19 00:00:00 EDT 20081
00000   38.617718       -121.440089
7401 TOULON LN  SACRAMENTO      95828   CA      4       2       1512    Residential     Thu May 15 00:00:00 EDT 2008    56950
38.488628       -121.387759
Time taken: 0.06 seconds, Fetched: 3 row(s)
hive (assignments)>
```

5. **<u>Create a new partitioned table having separate list of residential apartments with more than 2 beds. Table should have following attributes/fields:- i) Cityname ii) Baths iii) sq_feet iv) price v) flat_type vi) beds:</u>**

For this problem statement, I firstly created a partitioned table **"real_estate_partitioned_records"** as shown in snapshot below:

```
hive (assignments)> create table real_estate_partitioned_records(cityname string, baths int, sq_feet int, price int, beds int
)
                  > partitioned by (flat_type string)
                  > row format delimited
                  > fields terminated by ','
                  > stored as textfile;
OK
Time taken: 0.046 seconds
hive (assignments)>
```

Then, to load the data to this partitioned table, I overwrite it with the data from the original table **"real_estate_records"** by running a query as: **from real_estate_records t insert overwrite table real_estate_partitioned_records partition(flat_type) select t.city, t.bath, t.sq_ft, t.price, t.beds, t.type where t.type = "Residential" and t.beds > 2;** The code snippet is shown below in snapshot:

```
hive (assignments)> from real_estate_records t insert overwrite table real_estate_partitioned_records partition(flat_type) se
lect t.city, t.bath, t.sq_ft, t.price, t.beds, t.type  where t.type = "Residential" and t.beds > 2;
Query ID = cloudera_20230622155151_21be13ee-5cc8-4983-8392-e438f9f13a90
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks is set to 0 since there's no reduce operator
Starting Job = job_1687454370764_0022, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1687454370764_0022/
Kill Command = /usr/lib/hadoop/bin/hadoop job  -kill job_1687454370764_0022
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 0
2023-06-22 15:51:41,849 Stage-1 map = 0%,  reduce = 0%
2023-06-22 15:51:49,056 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 1.54 sec
MapReduce Total cumulative CPU time: 1 seconds 540 msec
Ended Job = job_1687454370764_0022
Stage-4 is selected by condition resolver.
Stage-3 is filtered out by condition resolver.
Stage-5 is filtered out by condition resolver.
Moving data to: hdfs://quickstart.cloudera:8020/user/hive/warehouse/assignments.db/real_estate_partitioned_records/.hive-stag
ing_hive_2023-06-22_15-51-34_820_7782934776152345673-1/-ext-10000
Loading data to table assignments.real_estate_partitioned_records partition (flat_type=null)
         Time taken for load dynamic partitions : 122
         Loading partition {flat_type=Residential}
         Time taken for adding to write entity : 1
Partition assignments.real_estate_partitioned_records{flat_type=Residential} stats: [numFiles=1, numRows=715, totalSize=18990
, rawDataSize=18275]
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1   Cumulative CPU: 1.54 sec   HDFS Read: 119308 HDFS Write: 19117 SUCCESS
Total MapReduce CPU Time Spent: 1 seconds 540 msec
OK
Time taken: 15.769 seconds
hive (assignments)>
```

After loading the data to the partitioned table **“real_estate_partitioned_records”**, I verified the data by running a HQL query: **select * from real_estate_partitioned_records;**