

### Task 1

Create a database named 'custom'.

Create a table named temperature\_data inside custom having below fields:

1. date (mm-dd-yyyy) format
2. zip code
3. temperature

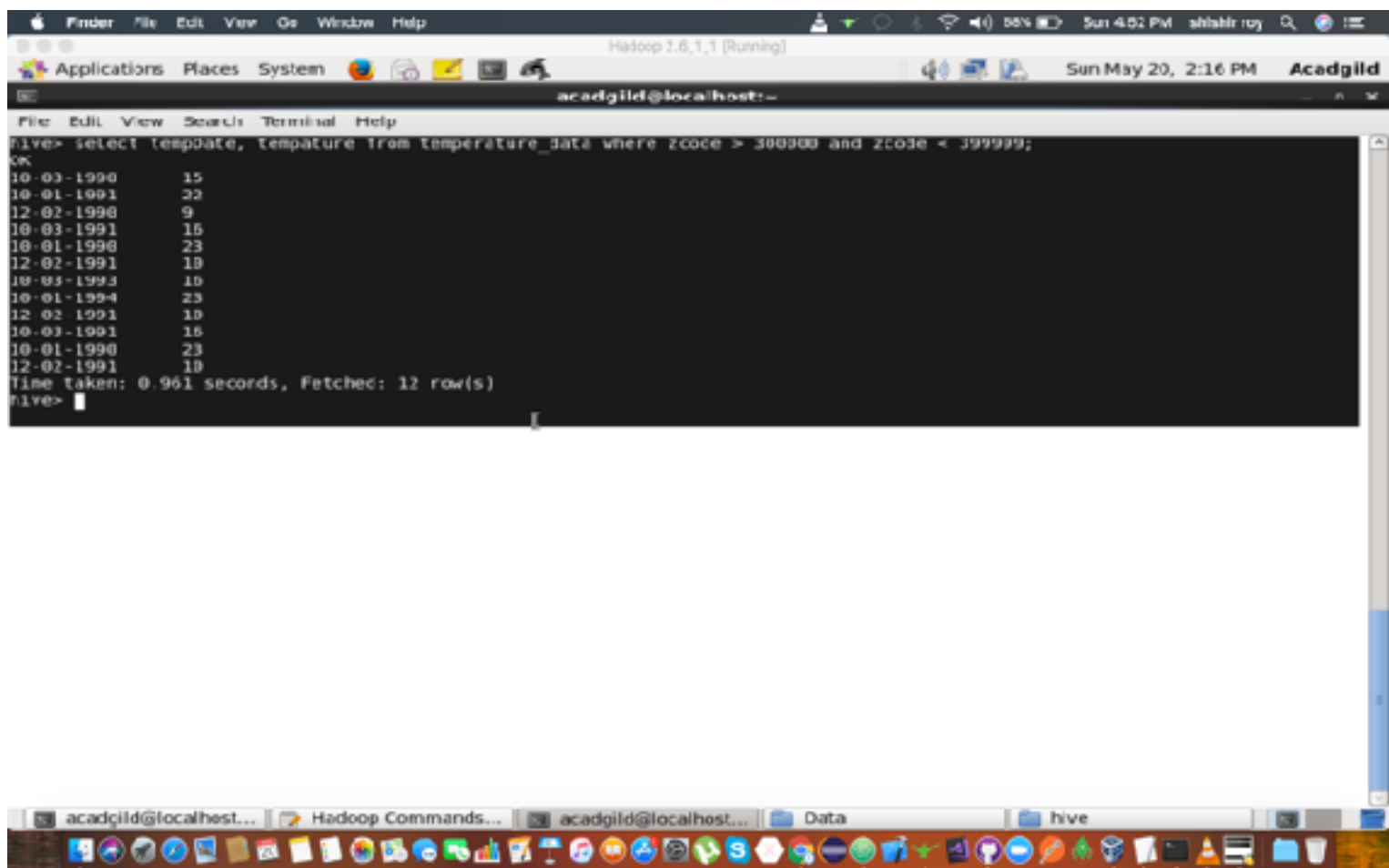
The table will be loaded from comma-delimited file.

Load the dataset.txt (which is ',' delimited) in the table.

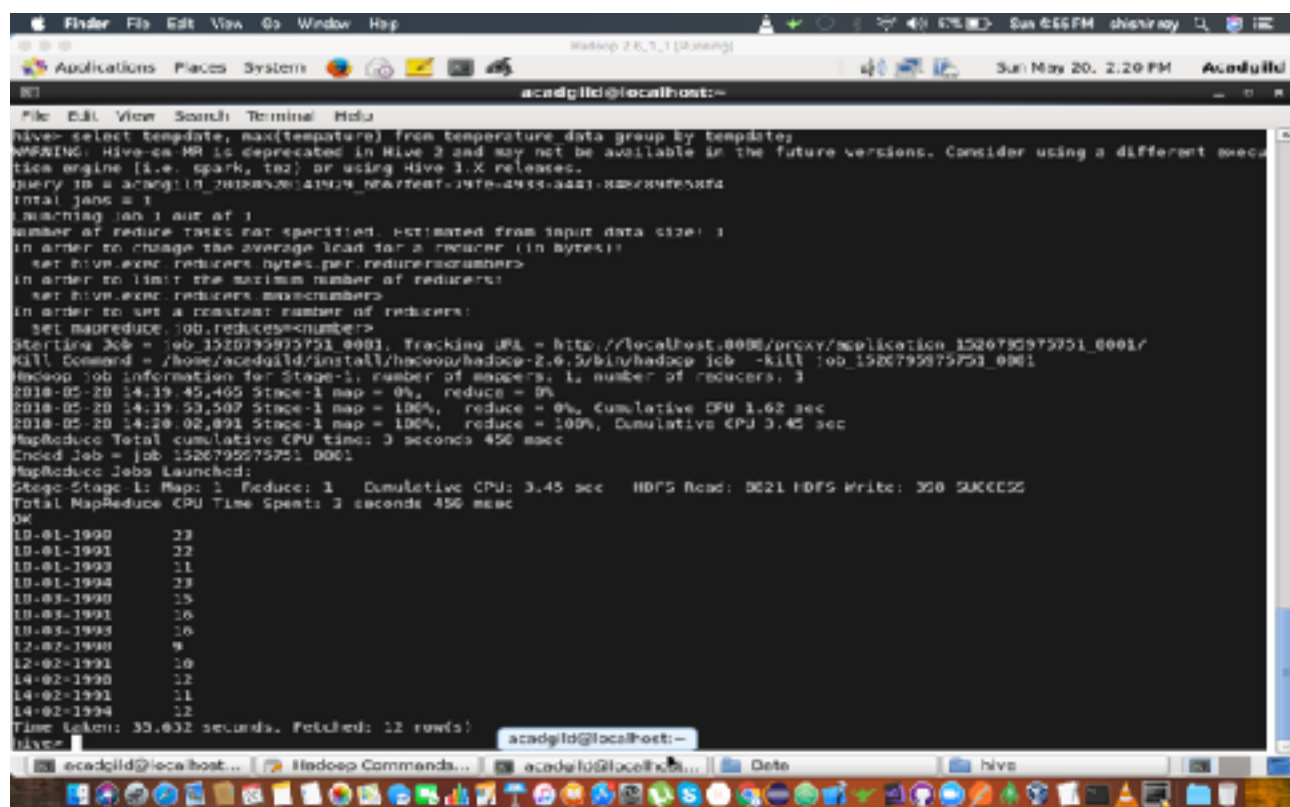
```
acacgild@localhost:~  
File Edit View Search Terminal Help  
hive> create table temperature_data(tempDate string, Zcode int, tempature int) row format delimited fields terminated by ',';  
OK  
Time taken: 0.844 seconds  
hive> load data local inpath '/home/acacgild/Desktop/data/nive/dataset.txt' into table temperature_data;  
Loading data to table custom.temperature_data  
OK  
Time taken: 2.588 seconds  
hive> select * from temperature_data;  
OK  
10-01-1990      123112  10  
14-02-1991      283901  11  
10-03-1990      381920  15  
10-01-1991      302918  22  
12-02-1990      384902  9  
10-01-1991      123112  11  
14-02-1990      283901  12  
10-03-1991      381920  16  
10-01-1990      302918  23  
12-02-1991      384902  10  
10-01-1991      123112  11  
14-02-1994      283901  12  
10-03-1993      381920  16  
10-01-1994      302918  23  
12-02-1991      384902  10  
10-01-1991      123112  11  
14-02-1990      283901  12  
10-03-1991      381920  16  
10-01-1990      302918  23  
12-02-1991      384902  10  
Time taken: 2.645 seconds, Fetched: 20 row(s)  
hive>
```

### Task 2

- Fetch date and temperature from temperature\_data where zip code is greater than 300000 and less than 399999.



Calculate maximum temperature corresponding to every year from temperature\_data table.



- Calculate maximum temperature from temperature\_data table corresponding to those years which have at least 2 entries in the table.

```

hadoop 2.6.1_1 [Running]
Applications Places System
acadgild@localhost:~$
File Edit View Search Terminal Help
hive> select tempDate, max(t1.temperature) as temperature from (select substring(tempDate,7,4) tempDate, temperature
re data) t1 group by tempDate having count(t1.tempDate) > 2;
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_20180520142201_f3bc68f8-a411-4ab3-86db-fa242be90f1b
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.reducers=<number>
Starting Job = job_1526795975751_0002, Tracking URL = http://localhost:8088/proxy/application_1526795975751_0002/
Kill Command = /home/acadgild/install/hadoop/hadoop-2.6.5/bin/hadoop job -kill job_1526795975751_0002
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2018-05-20 14:22:12,459 Stage-1 map = 0%, reduce = 0%
2018-05-20 14:22:20,379 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 1.90 sec
2018-05-20 14:22:29,364 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 4.51 sec
MapReduce Total cumulative CPU time: 4 seconds 510 msec
Ended Job = job_1526795975751_0002
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 4.51 sec HDFS Read: 10108 HDFS Write: 127 SUCCESS
Total MapReduce CPU Time Spent: 4 seconds 510 msec
OK
1990      23
1991      22
Time taken: 19.915 seconds, Fetched: 2 row(s)
hive>

```

acadgild@localhost... \*Hadoop Command... acadgild@localhost... Data hive

- Create a view on the top of last query, name it temperature\_data\_vw.

```

hive> create view temperature_data_vw as select tempdate, max(t1.temperature) as temperature from (select substr(tempdate,2,4)
tempdate, temperature from temperature_data) t1 group by tempdate having count(t1.tempdate) > 2;
OK
Time taken: 0.525 seconds
hive> select * from temperature_data_vw;
Warning: Hive-0.10.0 is deprecated in Hive 2 and may not be available in the future versions. Consider using a different exec
ution engine (i.e. Spark, Tez) or using Hive 1.x releases.
Query ID = acedgild_781856544444_2726434-117a-d8e8-a5fe-aa5a4a4e4e4e
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.reducers=number
Starting Job = job_1526795075751_0003, Tracking URL = http://localhost:8080/proxy/application_1526795075751_0003/
Kill Command = /home/acedgild/install/hadoop/hadoop-2.6.5/bin/hadoop job -kill job_1526795075751_0003
Hadoop job information for Stage-1: number of mappers: 1, number of reducers: 1
2018-05-20 14:24:46,814 Stage-1 map = 0%, reduce = 0%, Cumulative CPU 1.99 sec
2018-05-20 14:24:54,490 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 1.99 sec
2018-05-20 14:25:05,400 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 5.03 sec
MapReduce Total cumulative CPU time: 5 seconds 30 msec
Ended Job = job_1526795075751_0003
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 5.03 sec HDFS Read: 16230 HDFS Write: 127 SUCCESS
Total MapReduce CPU Time Spent: 5 seconds 30 msec
OK
Hive>

```

- Export contents from temperature\_data\_vw to a file in local file system, such that each file is ‘|’ delimited.