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.

```
[acadgild@localhost ~]$ ls -l;
total 287976
-rw-rw-r--.  1 acadgild acadgild    244438 Aug  4 14:08 airports.csv
-rw-rw-r--.  1 acadgild acadgild       437 Aug  8 00:02 dataset.txt
-rw-rw-r--.  1 acadgild acadgild 247963212 Aug  5 13:38 DelayedFlights.csv
```

As we could see that dataset.txt is having date field in 'dd-mm-yyyy' format. But we need to have date field in 'mm-dd-yyyy' format in temperature_data1 table. So we have created a temporary table first and load data from dataset.txt file into this temporary table. Then we have inserted data into 'temperature_data' table from this temporary table using insert into select statement.

```
hive> create table temporaray(temp_date string,zip_code int,temperature int) row format delimited fields terminated by ',';
OK
Time taken: 0.903 seconds
hive>

hive> load data local inpath '/home/acadgild/dataset.txt' into table temporaray;
Loading data to table default.temporaray
OK
Time taken: 1.606 seconds
hive>

hive> create table  temperature_data1(temp_date string,zip_code int, temperature int ) row format delimited  fields terminated  by ',';
OK
Time taken: 0.144 seconds
hive>
```

we have inserted data into 'temperature_data1' table from this temporary table using below insert into select statement with the help of from_unixtime and unix_timestamp functions.

```
hive> insert into table temperature_data1 select from_unixtime(unix_timestamp(temp_date, 'dd-mm-yyyy'), 'mm-dd-yyyy') ,zip_code,temperature from temporaray;
WARNING: 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 u
sing Hive 1.X releases.
Query ID = acadgild_20180808003655_f3707332-f726-4d9e-bfe0-378b22f826b9
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_1533624178420_0024, Tracking URL = http://localhost:8088/proxy/application_1533624178420_0024/
Kill Command = /home/acadgild/install/hadoop/hadoop-2.6.5/bin/hadoop job  -kill job_1533624178420_0024
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 0
2018-08-08 00:37:10,946 Stage-1 map = 0%,  reduce = 0%
2018-08-08 00:37:20,156 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 2.1 sec
MapReduce Total cumulative CPU time: 2 seconds 100 msec
Ended Job = job_1533624178420_0024
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 directory hdfs://localhost:8020/user/hive/warehouse/temperature_data1/.hive-staging_hive_2018-08-08_00-36-55_008_6592943498520485631-1/-ext-1000
0
Loading data to table default.temperature_data1
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1   Cumulative CPU: 2.1 sec   HDFS Read: 4793 HDFS Write: 501 SUCCESS
Total MapReduce CPU Time Spent: 2 seconds 100 msec
OK
Time taken: 27.311 seconds
hive>
```

```
hive> select * from temperature_data1;
OK
01-10-1990      123112  10
02-14-1991      283901  11
03-10-1990      381920  15
01-10-1991      302918  22
02-12-1990      384902  9
01-10-1991      123112  11
02-14-1990      283901  12
03-10-1991      381920  16
01-10-1990      302918  23
02-12-1991      384902  10
01-10-1993      123112  11
02-14-1994      283901  12
03-10-1993      381920  16
01-10-1994      302918  23
02-12-1991      384902  10
01-10-1991      123112  11
02-14-1990      283901  12
03-10-1991      381920  16
01-10-1990      302918  23
02-12-1991      384902  10
Time taken: 0.325 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.

hive> set hive.cli.print.header=true;

```
hive> set hive.cli.print.header=true;
hive> You have new mail in /var/spool/mail/acadgild
[acadgild@localhost ~]$
ort MobaXterm by subscribing to the professional edition here:  https://mobaxterm.mobatek.net
```

```
hive> set hive.cli.print.header=true;
hive>  select temp_date, temperature from temperature_data1 where zip_code > 300000 and zip_code < 399999;
OK
temp_date       temperature
03-10-1990      15
01-10-1991      22
02-12-1990      9
03-10-1991      16
01-10-1990      23
02-12-1991      10
03-10-1993      16
01-10-1994      23
02-12-1991      10
03-10-1991      16
01-10-1990      23
02-12-1991      10
Time taken: 0.32 seconds, Fetched: 12 row(s)
hive>
```

• Calculate maximum temperature corresponding to every year from temperature_data table.

We have used below select query by using max_temp and year as column alias for table : Output shows Maximum temperature corresponding to every year.

```
hive> select max(temperature) max_temp, date_format(from_unixtime(unix_timestamp(temp_date,'mm-dd-yyyy'),'yyyy-mm-dd'),'yyyy') year from temperature_data1 grou
p by date_format(from_unixtime(unix_timestamp(temp_date,'mm-dd-yyyy'),'yyyy-mm-dd'),'yyyy');
WARNING: 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 u
sing Hive 1.X releases.
Query ID = acadgild_20180808013326_679ebb30-b07a-4114-96ba-7b60845fadd9
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_1533624178420_0027, Tracking URL = http://localhost:8088/proxy/application_1533624178420_0027/
Kill Command = /home/acadgild/install/hadoop/hadoop-2.6.5/bin/hadoop job  -kill job_1533624178420_0027
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2018-08-08 01:33:35,762 Stage-1 map = 0%,  reduce = 0%
2018-08-08 01:33:44,519 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 2.22 sec
2018-08-08 01:33:53,335 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 4.17 sec
MapReduce Total cumulative CPU time: 4 seconds 170 msec
Ended Job = job_1533624178420_0027
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 4.17 sec   HDFS Read: 9785 HDFS Write: 167 SUCCESS
Total MapReduce CPU Time Spent: 4 seconds 170 msec
OK
max_temp        year
23      1990
22      1991
16      1993
23      1994
Time taken: 28.398 seconds, Fetched: 4 row(s)
hive>
```

Calculate maximum temperature from temperature_data1 table corresponding to those years which have at least 2 entries in the table

We have used below select query by using max_temp and year as column alias and count function for each year for table : Output shows Maximum temperature corresponding to every year having count of rows for each year as at least 2..

```
hive> select max(temperature) max_temp, date_format(from_unixtime(unix_timestamp(temp_date,'mm-dd-yyyy'),'yyyy-mm-dd'),'yyyy') year from temperature_data1
p by date_format(from_unixtime(unix_timestamp(temp_date,'mm-dd-yyyy'),'yyyy-mm-dd'),'yyyy')  having count(date_format(from_unixtime(unix_timestamp(temp_da
m-dd-yyyy'),'yyyy-mm-dd'),'yyyy')) >=2;
WARNING: 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)
sing Hive 1.X releases.
Query ID = acadgild_20180808040236_e7305343-56a2-4e16-a0a9-c3bf78945d45
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_1533624178420_0039, Tracking URL = http://localhost:8088/proxy/application_1533624178420_0039/
Kill Command = /home/acadgild/install/hadoop/hadoop-2.6.5/bin/hadoop job  -kill job_1533624178420_0039
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2018-08-08 04:02:46,492 Stage-1 map = 0%,  reduce = 0%
2018-08-08 04:02:55,434 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 2.34 sec
2018-08-08 04:03:05,212 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 5.06 sec
MapReduce Total cumulative CPU time: 5 seconds 60 msec
Ended Job = job_1533624178420_0039
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 5.06 sec   HDFS Read: 10657 HDFS Write: 167 SUCCESS
Total MapReduce CPU Time Spent: 5 seconds 60 msec
OK
max_temp        year
23      1990
22      1991
16      1993
23      1994
Time taken: 29.37 seconds, Fetched: 4 row(s)
hive>
```

Create a view on the top of last query, name it temperature_data_vw.

```
hive> create view temperature_data1_view as select max(temperature) max_temp,date_format(from_unixtime(unix_timestamp(temp_date,'mm-dd-yyyy'),'yyyy-mm-dd'),'yy
yy') year from temperature_data3 group by date_format(from_unixtime(unix_timestamp(temp_date,'mm-dd-yyyy'),'yyyy-mm-dd'),'yyyy') having count(date_format(from_
unixtime(unix_timestamp(temp_date,'mm-dd-yyyy'),'yyyy-mm-dd'),'yyyy'))>=2;
OK
max_temp        year
Time taken: 0.249 seconds
hive> select * from temperature_data1_view;
```

```
hive> create view temperature_data1_view as select max(temperature) max_temp,date_format(from_unixtime(unix_timestamp(temp_date,'mm-dd-yyyy'),'yyyy-mm-dd'),'
yy') year from temperature_data3 group by date_format(from_unixtime(unix_timestamp(temp_date,'mm-dd-yyyy'),'yyyy-mm-dd'),'yyyy') having count(date_format(fro
unixtime(unix_timestamp(temp_date,'mm-dd-yyyy'),'yyyy-mm-dd'),'yyyy'))>=2;
OK
max_temp        year
Time taken: 0.249 seconds
hive> select * from temperature_data1_view;
WARNING: 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
sing Hive 1.X releases.
Query ID = acadgild_20180808040839_d9cc0ce5-4f5f-4ff5-994d-cc2ac93f067f
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_1533624178420_0040, Tracking URL = http://localhost:8088/proxy/application_1533624178420_0040/
Kill Command = /home/acadgild/install/hadoop/hadoop-2.6.5/bin/hadoop job  -kill job_1533624178420_0040
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2018-08-08 04:08:48,164 Stage-1 map = 0%,  reduce = 0%
2018-08-08 04:08:57,135 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 2.32 sec
2018-08-08 04:09:08,034 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 4.95 sec
MapReduce Total cumulative CPU time: 4 seconds 950 msec
Ended Job = job_1533624178420_0040
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 4.95 sec   HDFS Read: 10733 HDFS Write: 167 SUCCESS
Total MapReduce CPU Time Spent: 4 seconds 950 msec
OK
temperature_data1_view.max_temp temperature_data1_view.year
23      1990
22      1991
16      1993
23      1994
Time taken: 30.139 seconds, Fetched: 4 row(s)
hive>
```

Export contents from temperature_data_vw to a file in local file system, such that each field is '|' delimited. We have used below insert statement to insert data into export directory with fields w insert statement to insert data into export directory with fields separated by '|'.

```
hive> insert overwrite  local directory  '/home/acadgild/export'row format delimited fields terminated by '|' select * from temperature_data1_view;
WARNING: 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 u
sing Hive 1.X releases.
Query ID = acadgild_20180808042453_5292bfe1-3a4b-48a6-97a2-2fcd3708c463
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_1533624178420_0041, Tracking URL = http://localhost:8088/proxy/application_1533624178420_0041/
Kill Command = /home/acadgild/install/hadoop/hadoop-2.6.5/bin/hadoop job  -kill job_1533624178420_0041
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2018-08-08 04:25:03,986 Stage-1 map = 0%,  reduce = 0%
2018-08-08 04:25:13,046 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 2.4 sec
2018-08-08 04:25:25,475 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 5.86 sec
MapReduce Total cumulative CPU time: 5 seconds 860 msec
Ended Job = job_1533624178420_0041
Moving data to local directory /home/acadgild/export
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 5.86 sec   HDFS Read: 10335 HDFS Write: 32 SUCCESS
Total MapReduce CPU Time Spent: 5 seconds 860 msec
OK
temperature_data1_view.max_temp temperature_data1_view.year
Time taken: 33.427 seconds
```

```
drwxrwxr-x.  4 acadgild acadgild      4096 Aug  4 10:17 eclipse-workspace
-rw-rw-r--.  1 acadgild acadgild     10824 Jul  7 13:21 employee.java
drwxrwxr-x.  2 acadgild acadgild      4096 Aug  8 04:25 export
-rw-rw-r--.  1 acadgild acadgild  16951772 Aug  7 19:33 genome-scores.csv
-rw-rw-r--.  1 acadgild acadgild     18103 Aug  7 19:33 genome-tags.csv
```

```
[acadgild@localhost ~]$ ls -l;
total 287976
-rw-rw-r--.  1 acadgild acadgild    244438 Aug  4 14:08 airports.csv
-rw-rw-r--.  1 acadgild acadgild       437 Aug  8 00:02 dataset.txt
-rw-rw-r--.  1 acadgild acadgild 247963212 Aug  5 13:38 DelayedFlights.csv
drwxr-xr-x.  3 acadgild acadgild      4096 Jul  9 16:49 Desktop
drwxr-xr-x.  2 acadgild acadgild      4096 Feb  2  2018 Documents
drwxr-xr-x.  2 acadgild acadgild      4096 Feb 13 14:24 Downloads
drwxrwxr-x.  3 acadgild acadgild      4096 Dec 29  2017 eclipse
drwxrwxr-x.  4 acadgild acadgild      4096 Aug  4 16:17 eclipse-workspace
-rw-rw-r--.  1 acadgild acadgild     10824 Jul  7 13:21 employee.java
drwxrwxr-x.  2 acadgild acadgild      4096 Aug  8 04:25 export
```

Below you can see that file ‘000000_0’ has been generated into export directory . Content of file ‘000000_0’ shows the output with field separated by ‘|’

```
[acadgild@localhost ~]$ ls -l export
total 4
-rw-r--r--. 1 acadgild acadgild 32 Aug  8 04:25 000000_0
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost ~]$
```

```
[acadgild@localhost ~]$ cat export/000000_0
23|1990
22|1991
16|1993
23|1994
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost ~]$
```