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.

```
CREATE DATABASE custom;
CREATE TABLE temperature_data
        (
        full_date STRING,
        zip INT,
        temperature INT
        )
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ',';
LOAD DATA LOCAL INPATH '/home/acadgild/hadoop/dataset_Session 14.txt'
INTO TABLE custom.temperature_data;
```

```
hive (custom)>
            >
            >
            > Show tables;
OK
tab_name
temperature_data
Time taken: 0.17 seconds, Fetched: 1 row(s)
```

```
hive (custom)>
            >
            > Select*From temperature_data;
OK
temperature_data.full_date      temperature_data.zip    temperature_data.temperature
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
10-01-1993      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: 0.232 seconds, Fetched: 20 row(s)
```

Task 2

1. Fetch date and temperature from temperature_data where zip HIVE Commands is greater than 300000 and less than 399999.
2. Calculate maximum temperature corresponding to every year from temperature_data table.
3. Calculate maximum temperature from temperature_data table corresponding to those years which have at least 2 entries in the table.
4. Create a view on the top of last query, name it temperature_data_vw.
5. Export contents from temperature_data_vw to a file in local file system, such that each file is '|' delimited.

Create a DATABASE **"custom"** and create a TABLE "**temperature_data"** inside **custom** having below fields:
date (mm-dd-yyyy) format
zip code
temperature

```
hive (custom)>
            >
            >
            > Show tables;
OK
tab_name
temperature_data
Time taken: 0.17 seconds, Fetched: 1 row(s)
```

```
hive (custom)>
            >
            > Select*From temperature_data;
OK
temperature_data.full_date      temperature_data.zip    temperature_data.temperature
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
10-01-1993      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: 0.232 seconds, Fetched: 20 row(s)
```

1. Fetch date and temperature from **temperature_data** where **zip** is greater than 300000 and less than 399999.

HIVE Command

hive (custom) >Select * From temperature_data where zip BETWEEN 300000 AND 399999;

Output

```
hive>
    > Select * From temperature_data where zip BETWEEN 300000 AND 399999;
OK
10-03-1990      381920  15
10-01-1991      302918  22
12-02-1990      384902  9
10-03-1991      381920  16
10-01-1990      302918  23
12-02-1991      384902  10
10-03-1993      381920  16
10-01-1994      302918  23
12-02-1991      384902  10
10-03-1991      381920  16
10-01-1990      302918  23
12-02-1991      384902  10
Time taken: 0.241 seconds, Fetched: 12 row(s)
```

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

HIVE Command

hive (custom) > SELECT SUBSTRING(full_date,7,4), MAX(temperature) FROM custom.temperature_data GROUP BY SUBSTRING(full_date,7,4);

```
hive (custom)>
            >
            > SELECT SUBSTRING(full_date,7,4), MAX(temperature) FROM custom.temperature_data GROUP BY SUBSTRING(full_date,7,4);
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. s
Total MapReduce CPU Time Spent: 7 seconds 300 msec
OK
c0      c1
1990    23
1991    22
1993    16
1994    23
Time taken: 82.46 seconds, Fetched: 4 row(s)
```

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

HIVE Command

hive(custom)>SELECT full_date, MAX(t1.temperature) as temperature FROM (SELECT SUBSTRING(full_date,7,4) full_date, temperature FROM temperature_data)t1 GROUP BY full_date HAVING count(t1.full_date)>=2;

```
Total MapReduce CPU Time Spent: 9 seconds 610 msec
OK
1990    23
1991    22
1993    16
1994    23
Time taken: 62.17 seconds, Fetched: 4 row(s)
```

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

HIVE Command

CREATE VIEW temperature_data_vw AS SELECT full_date, MAX(t1.temperature) as temperature FROM (SELECT SUBSTRING(full_date,7,4) full_date, temperature FROM temperature_data)t1 GROUP BY full_date HAVING count(t1.full_date)>=2;

```
hive>
    >
    > CREATE VIEW temperature_data_vw AS SELECT full_date, MAX(t1.temperature) as temperature FROM (SELECT SUBSTRING(full_date,7,4) full_date, t
emperature FROM temperature_data)t1 GROUP BY full_date HAVING count(t1.full_date)>=2;
OK
Time taken: 0.866 seconds
hive>
```

```
hive>
    >
    > SELECT * FROM temperature_data_vw;
Total MapReduce CPU Time Spent: 8 seconds 200 msec
OK
1990    23
1991    22
1993    16
1994    23
Time taken: 59.647 seconds, Fetched: 4 row(s)
```

5. Export contents from temperature_data_vw to a file in local file system, such that each file is '|' delimited.

HIVE Command

INSERT OVERWRITE LOCAL DIRECTORY '/home/acadgild/hadoop/temperature_data_vw.txt' ROW FORMAT DELIMITED FIELDS TERMINATED BY '|' SELECT * FROM temperature_data_vw;

```
    >
    > INSERT OVERWRITE LOCAL DIRECTORY '/home/acadgild/hadoop/temperature_data_vw.txt' ROW FORMAT DELIMITED FIELDS TERMINATED BY '|' SELECT * FR
OM temperature_data_vw;
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 8.78 sec   HDFS Read: 1
Total MapReduce CPU Time Spent: 8 seconds 780 msec
OK
temperature_data_vw.full_date   temperature_data_vw.temperature
Time taken: 59.561 seconds
```

Output

```
  12 -rw-rw-r--. 1 acadgild acadgild     10236 Sep 24 12:34 pig_1506236182845.log
  12 -rw-rw-r--. 1 acadgild acadgild     10875 Oct 18 18:16 pig_1508330414609.log
  24 -rw-rw-r--. 1 acadgild acadgild     23144 Oct 18 18:32 pig_1508331024608.log
  28 -rw-rw-r--. 1 acadgild acadgild     25006 Oct 18 19:27 pig_1508332146570.log
  16 -rw-rw-r--. 1 acadgild acadgild     12386 Oct 18 20:19 pig_1508336301801.log
   8 -rw-rw-r--. 1 acadgild acadgild      6821 Oct 18 21:45 pig_1508341927699.log
 384 -rw-rw-r--. 1 acadgild acadgild    391461 Oct 22 12:25 piggybank-0.15.0.jar
   4 drwxrwxr-x. 2 acadgild acadgild      4096 Oct 18 19:27 pigout
   4 drwxrwxr-x. 2 acadgild acadgild      4096 Oct 18 21:45 pigoutassignment
   4 drwxrwxr-x. 2 acadgild acadgild      4096 Nov  5 13:05 player1.txt
   4 drwxrwxr-x. 2 acadgild acadgild      4096 Nov  5 13:16 player2.txt
  40 -rw-rw-r--. 1 acadgild acadgild     37792 Oct 24 11:46 Pokemon.csv
  24 -rw-rw-r--. 1 acadgild acadgild     21007 Sep 23 20:33 sample_temperature_dataset.csv
   4 -rw-rw-r--. 1 acadgild acadgild      2938 Oct 31 17:45 television.txt
   4 drwxrwxr-x. 2 acadgild acadgild      4096 Nov  7 12:44 temperature_data_vw.txt
   4 -rw-rw-r--. 1 acadgild acadgild       189 Oct 19 12:58 wordcountpig.txt
   4 -rw-rw-r--. 1 acadgild acadgild      1958 Oct 13 18:55 WordCount.txt
```

cat /home/acadgild/hadoop/temperature_data_vw.txt/*

```
[acadgild@localhost hadoop]$ cat /home/acadgild/hadoop/temperature_data_vw.txt/*
1990|23
1991|22
1993|16
1994|23
[acadgild@localhost hadoop]$
```