---

1. Create table tweets_exp.

```
CREATE EXTERNAL TABLE IF NOT EXISTS tweets_exp
(
id BIGINT,
tweet STRING,
user_name STRING,
user_statuses_count INT,
user_followers_count INT,
user_location STRING,
user_verified BOOLEAN,
tweet_fav_count INT,
retweet_count INT,
tweet_date STRING
)
```

2. Load the data in tweets table

```
load data inpath '/BDAFINAL/final_white_plus_yellow_final.csv' into TABLE
tweets_exp;
```

3. Display the table

```
Select * from table tweets_exp limit 10;
```

4. Create table dictionary

```
CREATE EXTERNAL TABLE dictionary
 (
 type string,
 length int,
 word string,
 pos string,
 stemmed string,
 polarity string
 )
 ROW FORMAT DELIMITED
 FIELDS TERMINATED BY '\t';
```

5. DESC table

```
desc dictionary;
OK
type string
length int
word string
pos string
```

stemmed string
polarity string
Time taken: 0.315 seconds, Fetched: 6 row(s)

6. Display the table dictionary
Select * from dictionary limit 10

7. Create view temp 1
```
create view temp_1 as select
id,
tweets_exp.tweet,
words
from tweets_exp
lateral view explode(sentences(lower(tweet))) dummy as words;
```

8. Create view temp 2
```
create view temp_2 as select
id,
temp_1.tweet,
word
from temp_1
lateral view explode(words) dummy as word;
```

9. Create view temp 3
```
create view temp_3 as select
id,
temp_2.tweet,
temp_2.word,
case s_d.polarity
when 'negative' then -1
when 'positive' then 1
else 0
end as polarity
from temp_2 left outer join dictionary s_d on temp_2.word = s_d.word;
```

10. Create table sentiment
```
create table tweets_sentiment as select
id,
case
when sum( polarity ) > 0 then 'positive'
when sum( polarity ) < 0 then 'negative'
else 'neutral'
end as sentiment
from temp_3 group by id;
```

11. Output with tweets and id

```
create table tweet_sentiment as select
id,tweet,
case
when sum( polarity ) > 0 then 'positive'
when sum( polarity ) < 0 then 'negative'
else 'neutral'
end as sentiment
from temp_3 group by id,tweet;
```
