Project 2 Part 2

Answer each of the following questions in BLUE colored font.

Project:

a. Include name of the team members.

Donnie Sengstack

Ashley Eads

b. Who is responsible for what? 1-2 sentences against each team member.

For Part 2 of Project 2, Ashley is responsible for creating the mega table and getting the data loaded into the mega table. Donnie is in charge of decomposing the mega table and writing create table statements for the decomposed tables. The two will work together on creating a UML diagram.

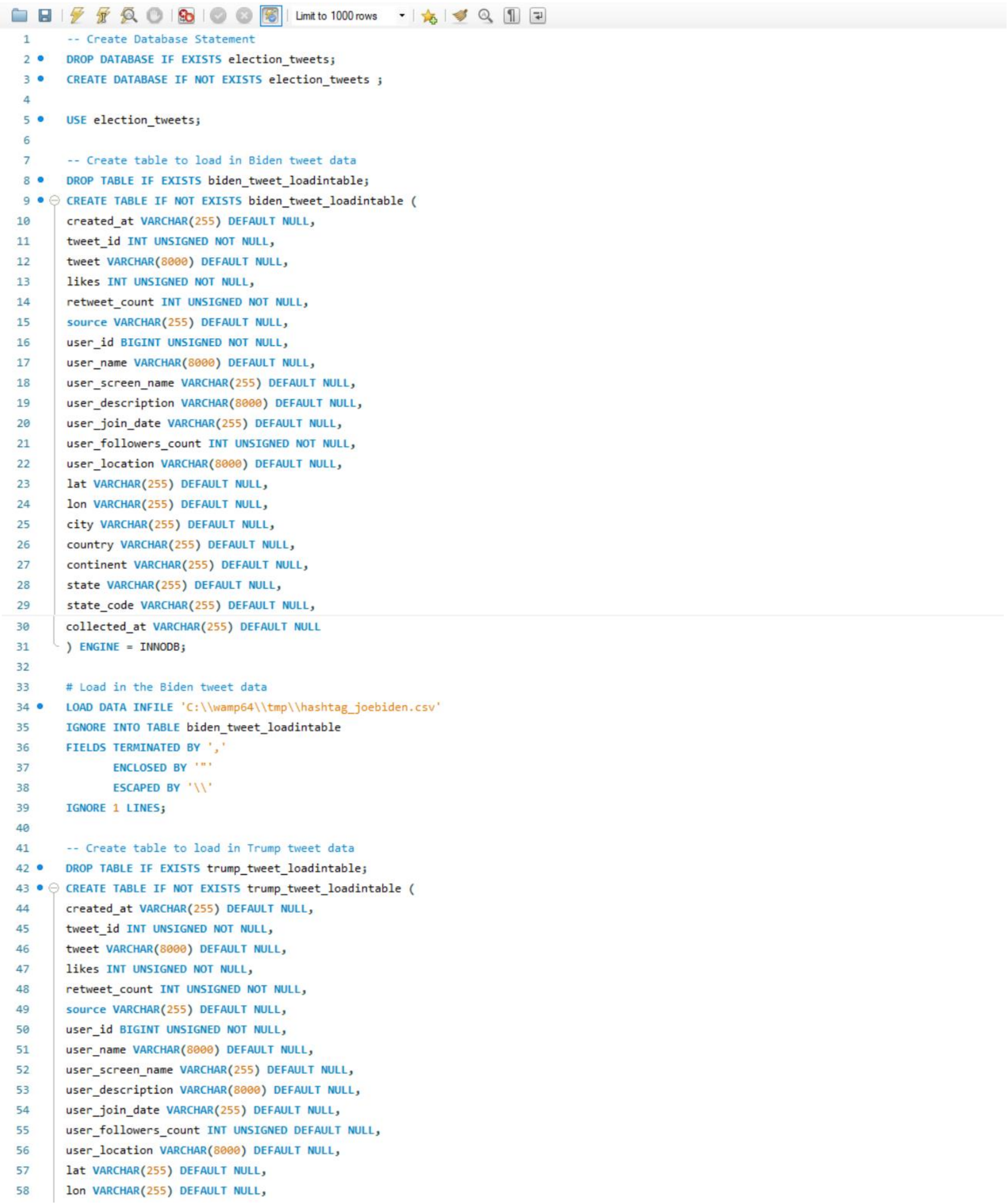
c. How are you sharing codebase? 1-2 sentences.

Donnie and Ashley will share code and addition files for the project in a GitHub repository that they have created.

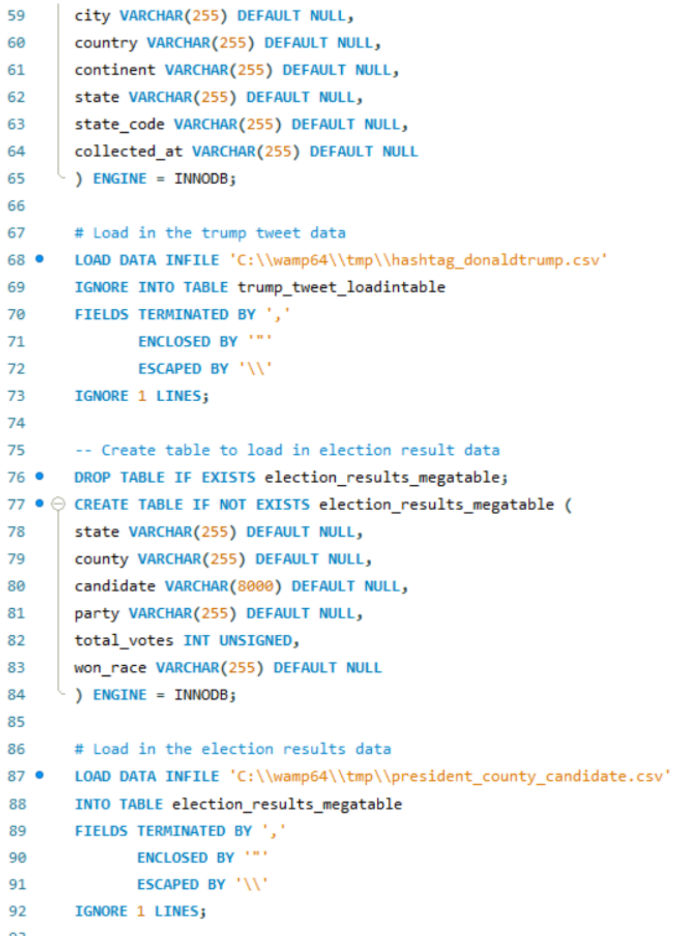
Database:

d. By this assignment submission date, you should have your raw data set imported in MySQL server in a megatable. You should also have started working on decomposition. Include screenshots of CREATE TABLE statements that you have completed/working on. This includes your megatable (required) and any other tables that you have designed. Include screenshot of any other relevant code blocks. Clearly label the screenshots.

We have finished loading the raw data into a mega table in MySQL. We loaded in three files containing raw data into the mega tables and you can see that in the screenshots below.

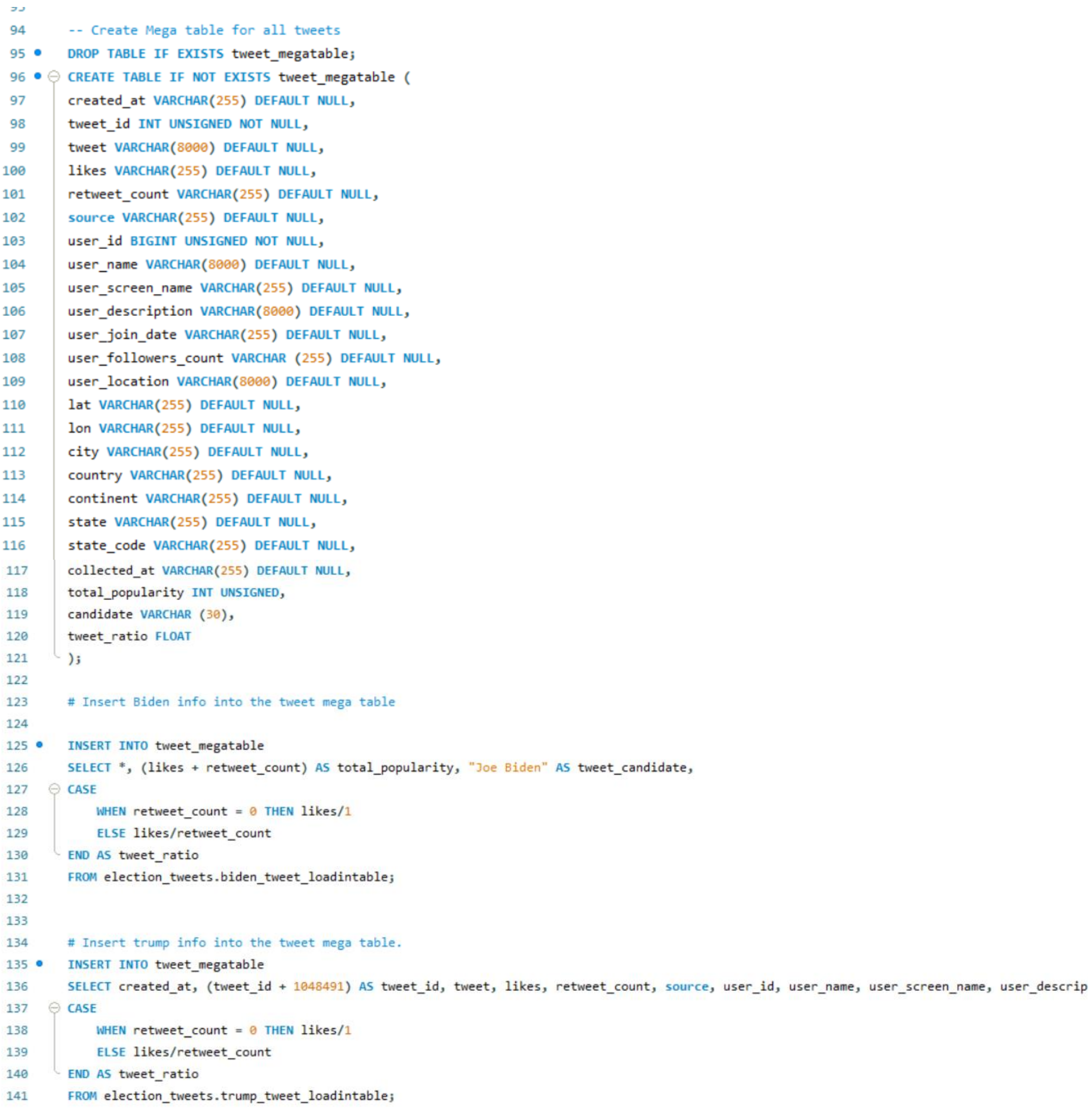


```sql
-- Create Database Statement
DROP DATABASE IF EXISTS election_tweets;
CREATE DATABASE IF NOT EXISTS election_tweets ;

USE election_tweets;

-- Create table to load in Biden tweet data
DROP TABLE IF EXISTS biden_tweet_loadintable;
CREATE TABLE IF NOT EXISTS biden_tweet_loadintable (
created_at VARCHAR(255) DEFAULT NULL,
tweet_id INT UNSIGNED NOT NULL,
tweet VARCHAR(8000) DEFAULT NULL,
likes INT UNSIGNED NOT NULL,
retweet_count INT UNSIGNED NOT NULL,
source VARCHAR(255) DEFAULT NULL,
user_id BIGINT UNSIGNED NOT NULL,
user_name VARCHAR(8000) DEFAULT NULL,
user_screen_name VARCHAR(255) DEFAULT NULL,
user_description VARCHAR(8000) DEFAULT NULL,
user_join_date VARCHAR(255) DEFAULT NULL,
user_followers_count INT UNSIGNED NOT NULL,
user_location VARCHAR(8000) DEFAULT NULL,
lat VARCHAR(255) DEFAULT NULL,
lon VARCHAR(255) DEFAULT NULL,
city VARCHAR(255) DEFAULT NULL,
country VARCHAR(255) DEFAULT NULL,
continent VARCHAR(255) DEFAULT NULL,
state VARCHAR(255) DEFAULT NULL,
state_code VARCHAR(255) DEFAULT NULL,
collected_at VARCHAR(255) DEFAULT NULL
) ENGINE = INNODB;

# Load in the Biden tweet data
LOAD DATA INFILE 'C:\\wamp64\\tmp\\hashtag_joebiden.csv'
IGNORE INTO TABLE biden_tweet_loadintable
FIELDS TERMINATED BY ','
       ENCLOSED BY '"'
       ESCAPED BY '\\'
IGNORE 1 LINES;

-- Create table to load in Trump tweet data
DROP TABLE IF EXISTS trump_tweet_loadintable;
CREATE TABLE IF NOT EXISTS trump_tweet_loadintable (
created_at VARCHAR(255) DEFAULT NULL,
tweet_id INT UNSIGNED NOT NULL,
tweet VARCHAR(8000) DEFAULT NULL,
likes INT UNSIGNED NOT NULL,
retweet_count INT UNSIGNED NOT NULL,
source VARCHAR(255) DEFAULT NULL,
user_id BIGINT UNSIGNED NOT NULL,
user_name VARCHAR(8000) DEFAULT NULL,
user_screen_name VARCHAR(255) DEFAULT NULL,
user_description VARCHAR(8000) DEFAULT NULL,
user_join_date VARCHAR(255) DEFAULT NULL,
user_followers_count INT UNSIGNED DEFAULT NULL,
user_location VARCHAR(8000) DEFAULT NULL,
lat VARCHAR(255) DEFAULT NULL,
lon VARCHAR(255) DEFAULT NULL,
```



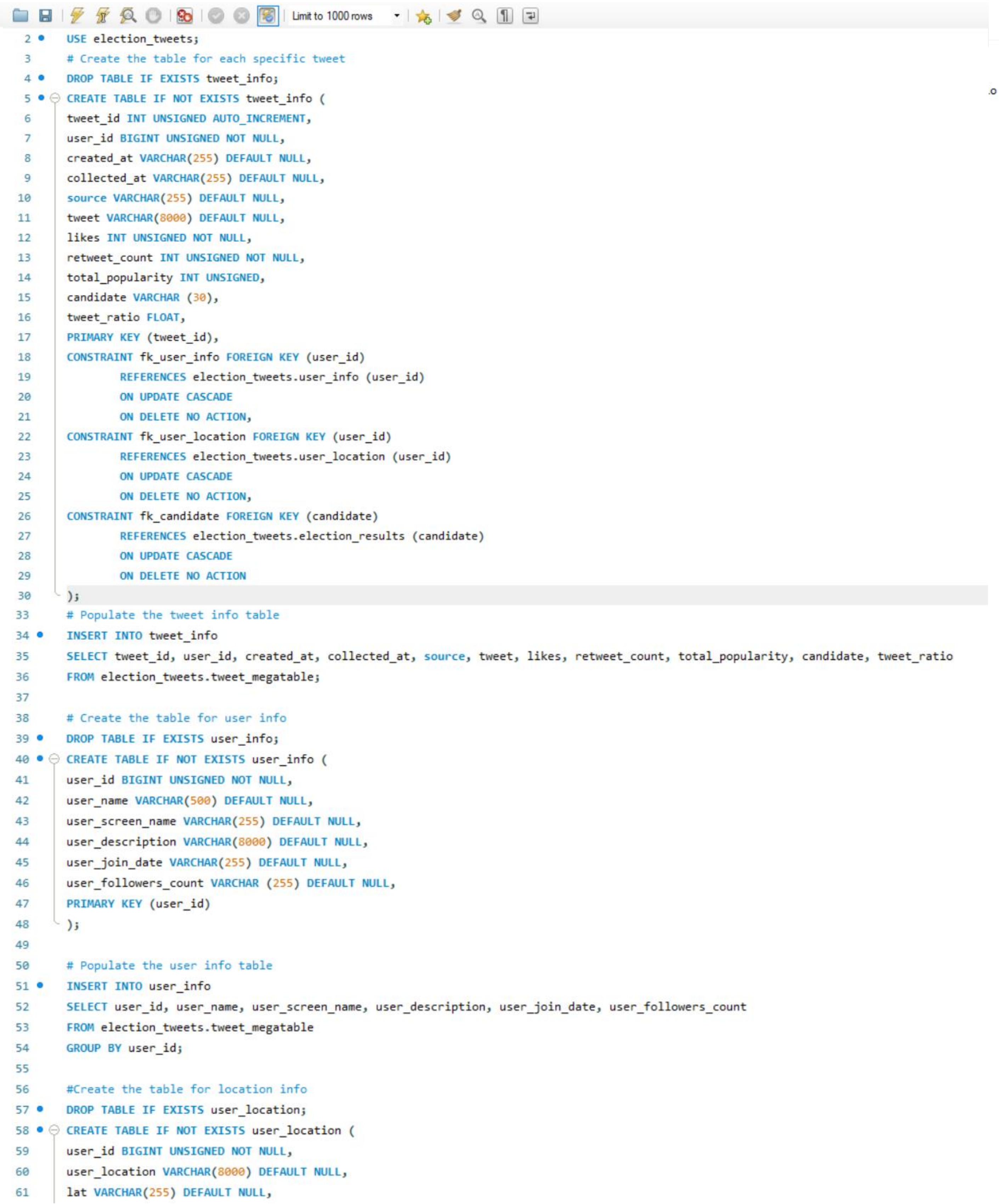
```
city VARCHAR(255) DEFAULT NULL,
country VARCHAR(255) DEFAULT NULL,
continent VARCHAR(255) DEFAULT NULL,
state VARCHAR(255) DEFAULT NULL,
state_code VARCHAR(255) DEFAULT NULL,
collected_at VARCHAR(255) DEFAULT NULL
) ENGINE = INNODB;

# Load in the trump tweet data
LOAD DATA INFILE 'C:\\wamp64\\tmp\\hashtag_donaldtrump.csv'
IGNORE INTO TABLE trump_tweet_loadintable
FIELDS TERMINATED BY ','
       ENCLOSED BY '"'
       ESCAPED BY '\\'
IGNORE 1 LINES;

-- Create table to load in election result data
DROP TABLE IF EXISTS election_results_megatable;
CREATE TABLE IF NOT EXISTS election_results_megatable (
state VARCHAR(255) DEFAULT NULL,
county VARCHAR(255) DEFAULT NULL,
candidate VARCHAR(8000) DEFAULT NULL,
party VARCHAR(255) DEFAULT NULL,
total_votes INT UNSIGNED,
won_race VARCHAR(255) DEFAULT NULL
) ENGINE = INNODB;

# Load in the election results data
LOAD DATA INFILE 'C:\\wamp64\\tmp\\president_county_candidate.csv'
INTO TABLE election_results_megatable
FIELDS TERMINATED BY ','
       ENCLOSED BY '"'
       ESCAPED BY '\\'
IGNORE 1 LINES;
```

After loading the raw data into mega tables, we then combined the two mega tables containing tweets into one mega table. To create the mega table for all tweet data we used the code below.

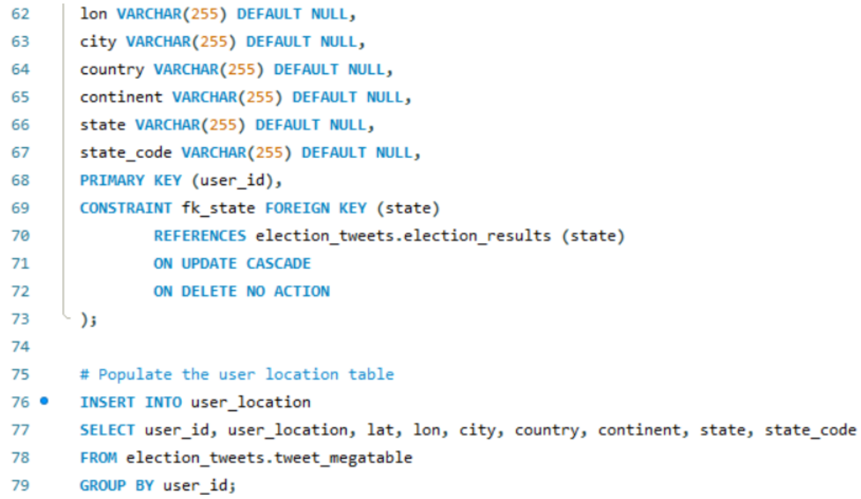


```
-- Create Mega table for all tweets
DROP TABLE IF EXISTS tweet_megatable;
CREATE TABLE IF NOT EXISTS tweet_megatable (
created_at VARCHAR(255) DEFAULT NULL,
tweet_id INT UNSIGNED NOT NULL,
tweet VARCHAR(8000) DEFAULT NULL,
likes VARCHAR(255) DEFAULT NULL,
retweet_count VARCHAR(255) DEFAULT NULL,
source VARCHAR(255) DEFAULT NULL,
user_id BIGINT UNSIGNED NOT NULL,
user_name VARCHAR(8000) DEFAULT NULL,
user_screen_name VARCHAR(255) DEFAULT NULL,
user_description VARCHAR(8000) DEFAULT NULL,
user_join_date VARCHAR(255) DEFAULT NULL,
user_followers_count VARCHAR (255) DEFAULT NULL,
user_location VARCHAR(8000) DEFAULT NULL,
lat VARCHAR(255) DEFAULT NULL,
lon VARCHAR(255) DEFAULT NULL,
city VARCHAR(255) DEFAULT NULL,
country VARCHAR(255) DEFAULT NULL,
continent VARCHAR(255) DEFAULT NULL,
state VARCHAR(255) DEFAULT NULL,
state_code VARCHAR(255) DEFAULT NULL,
collected_at VARCHAR(255) DEFAULT NULL,
total_popularity INT UNSIGNED,
candidate VARCHAR (30),
tweet_ratio FLOAT
);

# Insert Biden info into the tweet mega table

INSERT INTO tweet_megatable
SELECT *, (likes + retweet_count) AS total_popularity, "Joe Biden" AS tweet_candidate,
CASE
    WHEN retweet_count = 0 THEN likes/1
    ELSE likes/retweet_count
END AS tweet_ratio
FROM election_tweets.biden_tweet_loadintable;


# Insert trump info into the tweet mega table.
INSERT INTO tweet_megatable
SELECT created_at, (tweet_id + 1048491) AS tweet_id, tweet, likes, retweet_count, source, user_id, user_name, user_screen_name, user_descrip
CASE
    WHEN retweet_count = 0 THEN likes/1
    ELSE likes/retweet_count
END AS tweet_ratio
FROM election_tweets.trump_tweet_loadintable;
```

After that we decomposed the tweet data into third normal form to make sure each attribute was dependent on the table's primary key, the whole key, and nothing but the key. To do that we used the code below.

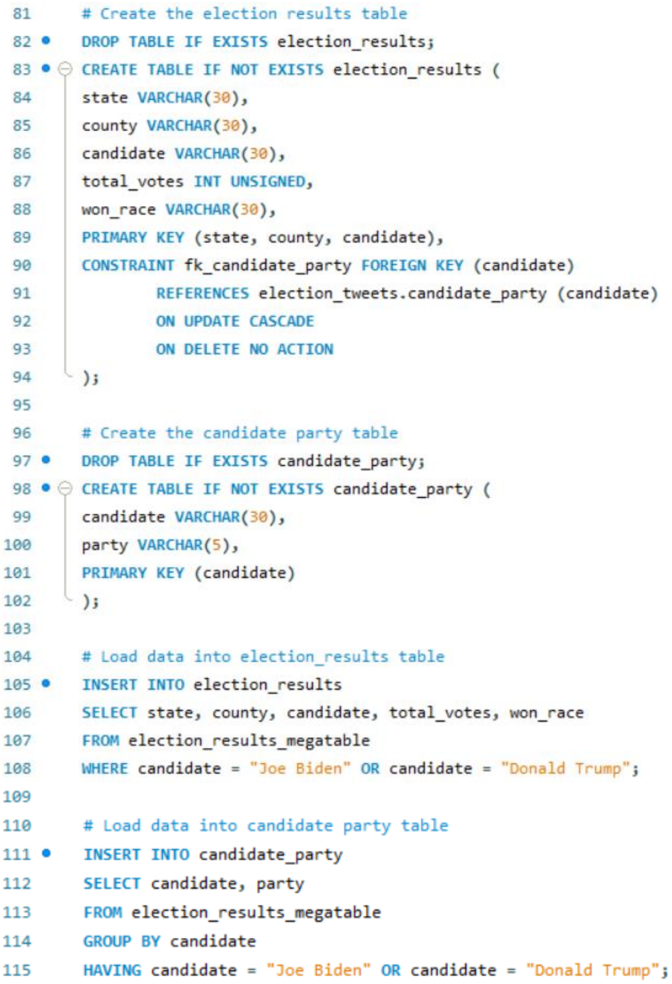


```
USE election_tweets;
# Create the table for each specific tweet
DROP TABLE IF EXISTS tweet_info;
CREATE TABLE IF NOT EXISTS tweet_info (
tweet_id INT UNSIGNED AUTO_INCREMENT,
user_id BIGINT UNSIGNED NOT NULL,
created_at VARCHAR(255) DEFAULT NULL,
collected_at VARCHAR(255) DEFAULT NULL,
source VARCHAR(255) DEFAULT NULL,
tweet VARCHAR(8000) DEFAULT NULL,
likes INT UNSIGNED NOT NULL,
retweet_count INT UNSIGNED NOT NULL,
total_popularity INT UNSIGNED,
candidate VARCHAR (30),
tweet_ratio FLOAT,
PRIMARY KEY (tweet_id),
CONSTRAINT fk_user_info FOREIGN KEY (user_id)
        REFERENCES election_tweets.user_info (user_id)
        ON UPDATE CASCADE
        ON DELETE NO ACTION,
CONSTRAINT fk_user_location FOREIGN KEY (user_id)
        REFERENCES election_tweets.user_location (user_id)
        ON UPDATE CASCADE
        ON DELETE NO ACTION,
CONSTRAINT fk_candidate FOREIGN KEY (candidate)
        REFERENCES election_tweets.election_results (candidate)
        ON UPDATE CASCADE
        ON DELETE NO ACTION
);
# Populate the tweet info table
INSERT INTO tweet_info
SELECT tweet_id, user_id, created_at, collected_at, source, tweet, likes, retweet_count, total_popularity, candidate, tweet_ratio
FROM election_tweets.tweet_megatable;

# Create the table for user info
DROP TABLE IF EXISTS user_info;
CREATE TABLE IF NOT EXISTS user_info (
user_id BIGINT UNSIGNED NOT NULL,
user_name VARCHAR(500) DEFAULT NULL,
user_screen_name VARCHAR(255) DEFAULT NULL,
user_description VARCHAR(8000) DEFAULT NULL,
user_join_date VARCHAR(255) DEFAULT NULL,
user_followers_count VARCHAR (255) DEFAULT NULL,
PRIMARY KEY (user_id)
);

# Populate the user info table
INSERT INTO user_info
SELECT user_id, user_name, user_screen_name, user_description, user_join_date, user_followers_count
FROM election_tweets.tweet_megatable
GROUP BY user_id;

#Create the table for location info
DROP TABLE IF EXISTS user_location;
CREATE TABLE IF NOT EXISTS user_location (
user_id BIGINT UNSIGNED NOT NULL,
user_location VARCHAR(8000) DEFAULT NULL,
lat VARCHAR(255) DEFAULT NULL,
```



```
lon VARCHAR(255) DEFAULT NULL,
city VARCHAR(255) DEFAULT NULL,
country VARCHAR(255) DEFAULT NULL,
continent VARCHAR(255) DEFAULT NULL,
state VARCHAR(255) DEFAULT NULL,
state_code VARCHAR(255) DEFAULT NULL,
PRIMARY KEY (user_id),
CONSTRAINT fk_state FOREIGN KEY (state)
        REFERENCES election_tweets.election_results (state)
        ON UPDATE CASCADE
        ON DELETE NO ACTION
);

# Populate the user location table
INSERT INTO user_location
SELECT user_id, user_location, lat, lon, city, country, continent, state, state_code
FROM election_tweets.tweet_megatable
GROUP BY user_id;
```

The mega table containing election result data also needed one more decomposition to get to 3NF. I did that in the code below.



```
# Create the election results table
DROP TABLE IF EXISTS election_results;
CREATE TABLE IF NOT EXISTS election_results (
state VARCHAR(30),
county VARCHAR(30),
candidate VARCHAR(30),
total_votes INT UNSIGNED,
won_race VARCHAR(30),
PRIMARY KEY (state, county, candidate),
CONSTRAINT fk_candidate_party FOREIGN KEY (candidate)
        REFERENCES election_tweets.candidate_party (candidate)
        ON UPDATE CASCADE
        ON DELETE NO ACTION
);

# Create the candidate party table
DROP TABLE IF EXISTS candidate_party;
CREATE TABLE IF NOT EXISTS candidate_party (
candidate VARCHAR(30),
party VARCHAR(5),
PRIMARY KEY (candidate)
);

# Load data into election_results table
INSERT INTO election_results
SELECT state, county, candidate, total_votes, won_race
FROM election_results_megatable
WHERE candidate = "Joe Biden" OR candidate = "Donald Trump";

# Load data into candidate party table
INSERT INTO candidate_party
SELECT candidate, party
FROM election_results_megatable
GROUP BY candidate
HAVING candidate = "Joe Biden" OR candidate = "Donald Trump";
```

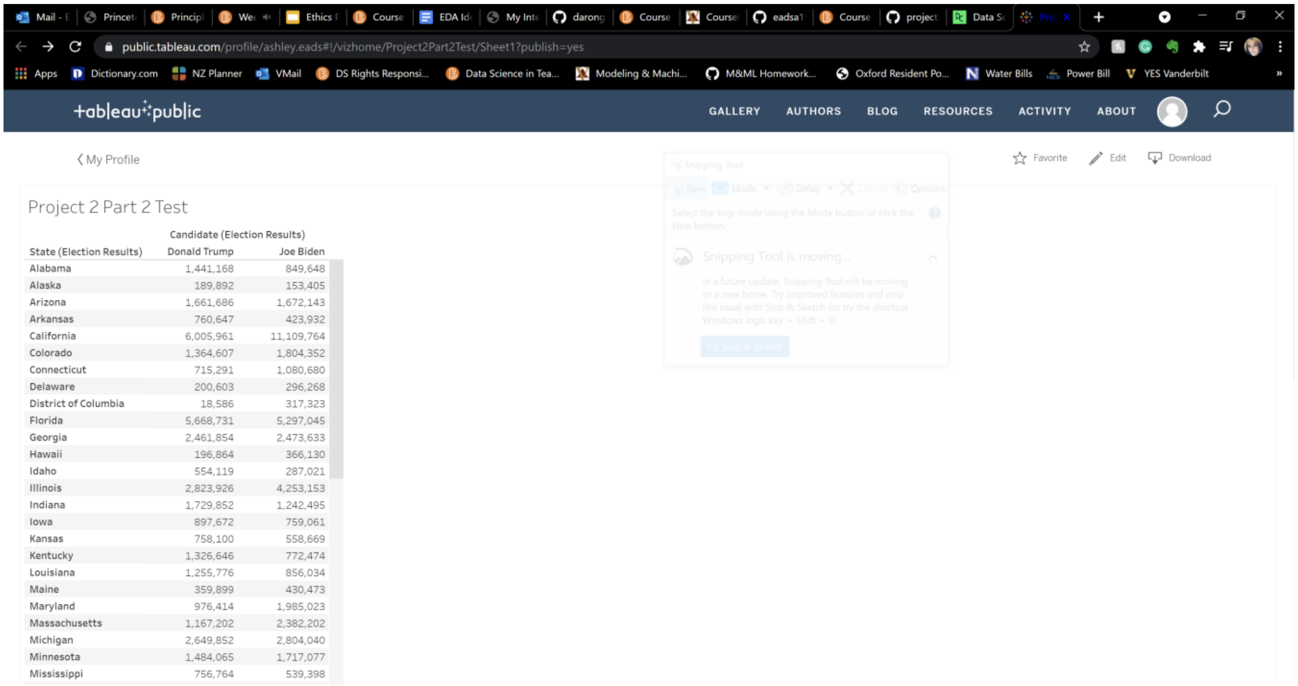
e. In terms of percentage how much you think you have completed on database side of the project? Describe the completed work in 2-3 sentences. Describe roadblocks, if any.

At this point in the project, we are finished with creating our databases and decomposing the tables into 3rd normal form. We have loaded the data into mega tables then decomposed those tables into 3rd normal form. All we have left to do is create our views, stored procedures, and triggers. I would say we are around 60% of the way done with the database side.

Front end:

f. By this assignment submission date, you should have decided on the front-end application programming language and have a successful connection established between the front end and the project database. Attach a screenshot of the browser showing a successful connection displaying some data from any table of your project (can be megatable). Clearly label the screenshots.

We have made the decision to use Tableau for our front-end application and we have been able to establish a successful connection between our application and the project database. Below is a screenshot of our browser showing a successful connection where our application is displaying the election_results table from the database.


public.tableau.com/profile/ashley.eads#!/vizhome/Project2Part2Test/Sheet1?publish=yes
+ableau+public
GALLERY
AUTHORS
BLOG
RESOURCES
ACTIVITY
ABOUT
My Profile
Favorite
Edit
Download

Project 2 Part 2 Test

	Candidate (Election Results)	
State (Election Results)	Donald Trump	Joe Biden
Alabama	1,441,168	849,648
Alaska	189,892	153,405
Arizona	1,661,686	1,672,143
Arkansas	760,647	423,932
California	6,005,961	11,109,764
Colorado	1,364,607	1,804,352
Connecticut	715,291	1,080,680
Delaware	200,603	296,268
District of Columbia	18,586	317,323
Florida	5,668,731	5,297,045
Georgia	2,461,854	2,473,633
Hawaii	196,864	366,130
Idaho	554,119	287,021
Illinois	2,823,926	4,253,153
Indiana	1,729,852	1,242,495
Iowa	897,672	759,061
Kansas	758,100	558,669
Kentucky	1,326,646	772,474
Louisiana	1,255,776	856,034
Maine	359,899	430,473
Maryland	976,414	1,985,023
Massachusetts	1,167,202	2,382,202
Michigan	2,649,852	2,804,040
Minnesota	1,484,065	1,717,077
Mississippi	756,764	539,398

g. Which front end application programming language are you working with? Has anyone from team has prior knowledge of working with front end? Has anyone from team has prior knowledge of working with the chosen front end application programming language. 2-3 sentences.

For this project we are going to use Tableau for our front-end application. Neither of us have prior knowledge of working with front end. Ashley does have prior experience working with Tableau which will be very helpful when it comes to formatting our front end.

h. What is the status of front-end application? In terms of percentage how much you think you have completed on front-end side of the project?

At this point in the project, we have completed a good chunk of the front-end application. We still need to do some additional formatting to make data presentation on the front-end look nice, but we believe we are around 60% of the way done.

Next deliverable

- What are your next steps? In a week time what do you plan to complete? Define clear goals.

  Our next steps on the database side are to create views, stored procedures, and triggers. We need to create these views, stored procedures, and triggers in a week's time in order to make sure that our database can interact with our front-end application. As far as the front end is concerned, in a week's time we need to make sure it can run the views, procedures, and triggers we create and display interesting information in a visually pleasing way.

Submission:

Complete this document and save it as pdf. You must submit a PDF file named **p2-part2-lastname1-lastname2.pdf** (For example if I submit this document with John Smith, I would name it p2-part2-singh-smith.pdf). Submit your files on Brightspace.

You must include your name and your partner name in the Brightspace submission text box.

Each member of the team must make the submission of same file.

Grading:

This Assignment will be graded on the following criteria:

1. Completeness of document.
2. Completeness of required components at this stage of project.
3. Clear evidence of work completed.

NO grading will be done on file/s sent through email or not uploaded to Brightspace.