HW 10

In this homework, you will select data from a database, process it, and create a visualization using Matplotlib. This is similar to the final steps of your pipeline for the final project.

We have provided:

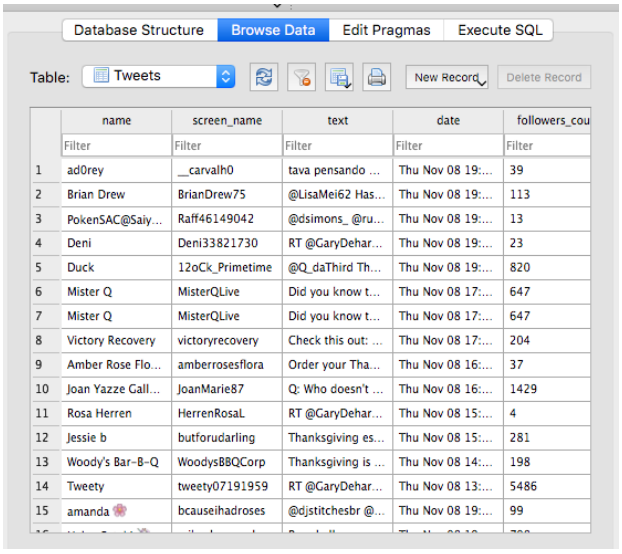
- tweet_data.sqlite - a database with tweets collected over time.
- visualize.py - starter code for the functions below.

Make sure you are using Anaconda python for this assignment (preferred), or have installed Matplotlib on your own (using `pip install matplotlib` or another installation method).

Part 1: Look at the database

Check out tweet_data.sqlite in your DB Browser for SQLite program.

1. Open DB Browser for SQLite
2. Click on “Open Database” and choose tweet_data.sqlite.
3. Click on Browse Data



	name	screen_name	text	date	followers_cou
	Filter	Filter	Filter	Filter	Filter
1	ad0rey	__carvalh0	tava pensando ...	Thu Nov 08 19:...	39
2	Brian Drew	BrianDrew75	@LisaMei62 Has...	Thu Nov 08 19:...	113
3	PokenSAC@Saiy...	Raff46149042	@dsimons_ @ru...	Thu Nov 08 19:...	13
4	Deni	Deni33821730	RT @GaryDehar...	Thu Nov 08 19:...	23
5	Duck	12oCk_Primetime	@Q_daThird Th...	Thu Nov 08 19:...	820
6	Mister Q	MisterQLive	Did you know t...	Thu Nov 08 17:...	647
7	Mister Q	MisterQLive	Did you know t...	Thu Nov 08 17:...	647
8	Victory Recovery	victoryrecovery	Check this out: ...	Thu Nov 08 17:...	204
9	Amber Rose Flo...	amberrosesflora	Order your Tha...	Thu Nov 08 16:...	37
10	Joan Yazze Gall...	JoanMarie87	Q: Who doesn't ...	Thu Nov 08 16:...	1429
11	Rosa Herren	HerrenRosaL	RT @GaryDehar...	Thu Nov 08 15:...	4
12	Jessie b	butforudarling	Thanksgiving es...	Thu Nov 08 15:...	281
13	Woody's Bar-B-Q	WoodysBBQCorp	Thanksgiving is ...	Thu Nov 08 14:...	198
14	Tweety	tweety07191959	RT @GaryDehar...	Thu Nov 08 13:...	5486
15	amanda	bcauseihadroses	@djstitchesbr @...	Thu Nov 08 19:...	99

Part 2: Process the data

Complete the *tweets_by_day(..)* function that accepts the filename of the database as a parameter, and returns a dictionary with the number of tweets that were tweeted each day of the week. The dictionary should look like:

```
{ 'Mon' : 1, 'Tue' : 2, 'Wed' : 3, … etc }
```

(If you did this already in the extra credit for HW 9, awesome! You can modify and re-use that code. Note that the keys are exactly 3 characters in this function.)

Your function must pass all the unit tests to get the full points.

Part 3: Visualize the data

Complete the function *barchart_tweets_by_day(..)*, which takes a dictionary created by the function in Part 1 and uses matplotlib functions to draw a bar chart with the days of the week on the x axis and the number of tweets on the named day on the y axis. The chart must have the appropriate x labels, y label, and title.

Submit an image file of your bar chart to Canvas, along with your repository link.

Extra credit: Visualize other data

Do users with more tweets also have more favorites? Let's make a scatterplot and see!

Complete function *scatterplot_num_tweets_vs_num_favs(..)* to plot a scatterplot of the number of favorites vs the number of tweets for each user in the dataset. Put the number of tweets (*num_statuses*) on the x axis and the number of favorites (*num_favourites*) on the y axis. The chart must have an appropriate x label, y label, and title.

Don't worry if the same user shows up twice in the dataset, because they have multiple tweets. You don't have to worry about removing duplicates.

Submit an image file of your scatterplot to Canvas.

Grading

Unit tests for tweets_by_day()	14 pts (2 pts per unit test with 7 tests)
Submission of barchart image file	5 pts

Created a bar chart from the data	26 pts
Title on bar chart	5 pts
Informative X-axis labels on bar chart	5 pts
Informative Y-axis label on bar chart	5 pts
Correct code and image file for scatterplot	*3 pts extra credit*
Total	**60 pts + 3 pts extra credit**