InfoTc 1040 Introduction to Problem Solving and Programming
Final Project: Tweet Manager

For the final project you are to create a Tweet Manager in Python according to the requirements specified in this document.

Description

Twitter is a social networking service that provides users with a platform to post short messages, called tweets.

As an example, here is the University of Missouri's official Twitter account: https://twitter.com/mizzou

You will create a menu-driven, Twitter-like program that allows users to make, view, and search through tweets.

***Note:** Unlike Twitter, we will not be publishing Tweets to the Internet. We will be storing, saving, and loading Tweet-like messages using Python.

Requirements

Tweet Class

Write a class named Tweet and save it as a file named *Tweet.py*. The Tweet class should have the following data attributes:

__author	To store the name of whoever is making the tweet.
__text	To store the tweet itself.
__age	To store the time when the tweet was created.

The Tweet class should also have the following methods:

__init__	This method should take two parameters, the author and text of the tweet. It should create __author and __text based on those parameters and it should create __age based on the current time.
get_author	Returns the value of the __author field.
get_text	Returns the value of the __text field.
get_age	This method calculates the difference between the current time and the value of the __age field. It should return a string based on the age of the tweet. For example, this method should return the following:

30s if the tweet was posted 30 seconds ago
15m if the tweet was posted 15 minutes ago
1h if the tweet was posted 1 hour, 10 minutes, and 42 seconds ago

* Note that the string returned from __get_age is only concerned with the largest unit of time. If the tweet is 0 to 59 seconds old, the value returned from __get_age will end with an "s" for "seconds". If the tweet is some number of minutes old, it will end with an "m" and ignore the seconds. Likewise, if the tweet is 1 or more hours old, it will ignore both the minutes and seconds.

To work with time, you'll need to use Python's time module. The documentation for that module is available here: https://docs.python.org/3.4/library/time.html

Tweet Manager

Write a program named *twitter.py*. This program will have 4 functional parts:

Tweet Menu
---
1. Make a Tweet
2. View Recent Tweets
3. Search Tweets
4. Quit

1. Make a Tweet

If the user chooses to make a tweet, they will be asked for their name and what they would like to tweet.

Twitter limits the length of tweets to 140 characters. Likewise, your program should check the length of the user's tweet and re-prompt them if it is greater than 140 characters.

Use the user's input and the Tweet class to create a new Tweet object.

2. View Recent Tweets

If the user chooses to view recent tweets, then display the 5 most recently created tweets including each tweet's author, text, and age.

If there are no tweets, display a message indicating so.

InfoTc 1040 Introduction to Problem Solving and Programming
Final Project: Tweet Manager

3. Search Tweets

Check to make sure there are tweets available to search. If there are no tweets, display a message indicating so.

If there are tweets, ask the user what they would like to search for. Look through the tweets and display **only** the tweets that contain that search.

The tweets should be displayed in chronological order, starting with the most recently created tweet.

If no tweets contain the user's search, display a message indicating so.

4. Quit

If the user chooses to quit, the program is to exit.

Storing, Saving, and Loading Tweets

In order to view and search through your tweets, you'll need to organize them in some way. One method for doing that would be to use a list. Whenever Tweet objects are created, add them to your list.

When the program ends, we don't want to lose our tweets. Each time you run the program, it should be possible to go back and view/search past tweets. To accomplish that, we'll need to save and load our list of tweets.

You can develop your own strategy of doing that if you'd like. Here are a few of our suggestions: Whenever a new Tweet object is created, add it to your list. Then serialize your list and save it to a file. Section 9.3 in the textbook provides information on serializing objects. "Serializing an object is the process of converting the object to a stream of bytes that can be saved to a file for later retrieval."

Then each time the program starts, check for that file of tweets. If it does not exist, start with a new, empty list. If it does exist, you can read that file and de-serialize it back into a list of Tweet objects.

Study *9.3 Serializing Objects* to learn how serialization works. Also, refer to *10.3 Working with Instances* for examples on how to serialize objects.

Your tweets should be saved in a file named *tweets.dat*

InfoTc 1040 Introduction to Problem Solving and Programming
Final Project: Tweet Manager

Sample Program Operation
* User input is highlighted in **orange**

Tweet Menu
---
1. Make a Tweet
2. View Recent Tweets
3. Search Tweets
4. Quit

What would you like to do? **1**

What is your name? **Sally**
What would you like to tweet? **This is my first tweet!**
Sally, your tweet has been saved.

Tweet Menu
---
1. Make a Tweet
2. View Recent Tweets
3. Search Tweets
4. Quit

What would you like to do? **2**

Recent Tweets

Sally - 6s
This is my first tweet!

Tweet Menu
---
1. Make a Tweet
2. View Recent Tweets
3. Search Tweets
4. Quit

What would you like to do? **1**
What is your name? **David**
What would you like to tweet? **I also like to tweet. This is #fun**
David, your tweet has been saved.

InfoTc 1040 Introduction to Problem Solving and Programming
Final Project: Tweet Manager

```
Tweet Menu
----------
1. Make a Tweet
2. View Recent Tweets
3. Search Tweets
4. Quit

What would you like to do? 2

Recent Tweets
-------------------
David - 2s
I also like to tweet. This is #fun

Sally - 3m
This is my first tweet!


Tweet Menu
----------
1. Make a Tweet
2. View Recent Tweets
3. Search Tweets
4. Quit

What would you like to do? 3

What would you like to search for? fun

Search Results
-------------------
David - 1m
I also like to tweet. This is #fun


Tweet Menu
----------
1. Make a Tweet
2. View Recent Tweets
3. Search Tweets
4. Quit

What would you like to do? 4
Thank you for using the Tweet Manager!
```

InfoTc 1040 Introduction to Problem Solving and Programming
Final Project: Tweet Manager

Notes

The output of your program should match the format of the sample program operation above.

If no tweets exist, the view and search options should display appropriate feedback:

```
Tweet Menu
----------
1. Make a Tweet
2. View Recent Tweets
3. Search Tweets
4. Quit

What would you like to do? 2

Recent Tweets
-------------
There are no recent tweets.

Tweet Menu
----------
1. Make a Tweet
2. View Recent Tweets
3. Search Tweets
4. Quit

What would you like to do? 3

There are no tweets to search.

Tweet Menu
----------
1. Make a Tweet
2. View Recent Tweets
3. Search Tweets
4. Quit

What would you like to do?
```

InfoTc 1040 Introduction to Problem Solving and Programming
Final Project: Tweet Manager

If a search yields no results, display an appropriate message like so:

```
Tweet Menu
----------
1. Make a Tweet
2. View Recent Tweets
3. Search Tweets
4. Quit

What would you like to do? 3

What would you like to search for? IT1040

Search Results
--------------
No tweets contained IT1040

Tweet Menu
----------
1. Make a Tweet
2. View Recent Tweets
3. Search Tweets
4. Quit

What would you like to do?
```

The user's menu choices should be handled for errors. Exceptions should not cause your program to crash. For example:

```
Tweet Menu
----------
1. Make a Tweet
2. View Recent Tweets
3. Search Tweets
4. Quit

What would you like to do? one
Please enter a numeric value.

What would you like to do? -1
Please select a valid option.
```

What would you like to do? **5**
Please select a valid option.

What would you like to do?

Remember to check the length of each tweet. Tweets cannot be longer than 140 characters. For example:

Tweet Menu
---
1. Make a Tweet
2. View Recent Tweets
3. Search Tweets
4. Quit

What would you like to do? **1**

What is your name? **Julie**
What would you like to tweet? **This is a long tweet #toolong. The program should prevent me from posting this because it is longer than one hundred and forty characters. Thanks!**

Tweets can only be 140 characters!

What would you like to tweet?

InfoTc 1040 Introduction to Problem Solving and Programming
Final Project: Tweet Manager

Testing

Make sure you have thoroughly tested your program!

Submission

Put your *tweet.py* and *twitter.py* files in a folder named **<lastname><firstname>TweetManager** and zip the folder. Do not include characters other than a-z and A-Z in the folder name. The zip file is to be submitted for this assignment.

If you need additional help, post your questions on the class Help discussion board or contact an eLearning mentor.