CS 112 Lab Assignment

Instructor: Dan Fleck
Lab: Web Favorites (to learn how to create and use Classes)
Due Date: As with all labs, this lab is due one week from your lab section!

Overview

This lab will familiarize you with creating and using Classes and Objects. You have already used Objects throughout all of your graphics programming. You will now learn the basics of creating your own Class and then instantiating/constructing your own Objects of that class.

Using the power of Python you will create a list of web bookmarks that you can choose from and open in a web browser. To do this you will be creating a Class called WebBookmark. This class will have data and operations. (That's what classes do… they hold data and define operations on that data!) Your WebBookmarks class will store a title and URL for a website, and allow you to open that website in a browser.

Lastly, you will create a menu that shows your favorites and lets the user choose one to open. That menu can be a text-based menu (like we did in the earlier labs) or a graphical menu using widgets…. it is up to you!

Assignment

WebBookmark class:

This class has two data attributes.

- A string that holds the title of the website
- A string that holds the URL of the website

Constructor: The constructor must take as formal parameters the title and URL. The constructor must then store them for use later as attributes of the object.

Operations:

- showWebpage() --- You must have a showWebpage operation. This operation will open the URL for this object in the default browser. Hint: To do this, all you need to do is:
  - import webbrowser
  - webbrowser.open("http://cs.gmu.edu/")
- getURL() – should return the URL attribute of this object to the caller
- getTitle() – should return the title attribute of this object to the caller

Menu module:
In a second file (named menu.py) you will create a menu for the user to choose a bookmark to see. The menu module must:

- Create 5 instances of the WebBookmark class (just pick your favorite 5 URLs and give them any title you want). These can be coded into your program, you do not need to read them from a file or let the user input them, etc… (in a bigger project you would.. but this is a lab).
- Create a menu that displays the title of each WebBookmark and lets the user choose one. (Either text or a graphical menu is fine.)
- Whichever one the user chooses must then be opened in a browser by calling the showWebpage operation and then repeat (ask the user again for a web page to view). If you use a text based menu you must not use recursion… you should use a while loop!

A simple example of running the program using a text menu is this:

```
Web Bookmarks Menu
------------------
0. Google
1. Dan Fleck Homepage
2. GMU Homepage
3. CNN
4. Ascii Art

Open which bookmark (or q to quit):1


Web Bookmarks Menu
------------------
0. Google
1. Dan Fleck Homepage
2. GMU Homepage
3. CNN
4. Ascii Art

Open which bookmark (or q to quit):
```

Grading Requirements (max points: 100)

1. 60 pts - WebBookmark Class:
    a. 25pts - Constructor takes as arguments title and URL and stores them
    b. 20pts - showWebpage method works
    c. 10pts - getTitle method works
    d. 10pts - getURL method works

2. 35pts - Menu module:
    a. 10pts - A menu is displayed with 5 (or more) bookmarks by calling the getTitle method from each of the 5 instances
    b. 10pts - 5 instances of the WebBookmark class are created
    c. 10pts - When the user chooses one, the showWebpage method is called
    d. 5pts - Then menu allows the user to keep choosing bookmarks (the program should not end until the user decides to end it).
        i. For the text-based menu this you must use a while loop, not recursion
        ii. For a graphical-based menu this should be simple

3. 5 pts – code has correct header and appropriate comments

    * Note: Individual GTAs may modify/amend this as desired for their sections

Helpful Hints

- See chapter 11 in the book and use the slides from class
- If you create a text based menu
    - Create a list data structure, and make each element in the list a WebBookmark object.
    - Create a while loop that asks for input from the user
        - Inside the while loop create a for loop that loops through every element in your list and calls the getTitle method to print out the title for the user to choose.

What to turn in:

1. menu.py - source code for the menu (don't forget the header in the file!)

2. webbookmark.py – source code that contains the WebBookmark class definition