**[M6T1 - Repl.it](https://faytechcc.blackboard.com/webapps/assignment/uploadAssignment?content_id=_6159321_1&course_id=_35059_1&group_id=&mode=view)**

For our first Flask site, we're going to keep things very simple, using repl.it.

Here's the in-class demo: <https://youtu.be/Y_x5cLwVLgQ>

**Part 1**

Log into your repl.it account, and enter the example listed here: <https://www.youtube.com/watch?v=9yU-lj20zd4>

Confirm that the user view count works properly when you refresh the page.

**Part 2 (Bonus, Optional)**

Create a different URL endpoint (so, another @app.route() call followed by another method), and have that print a few paragraphs of text (this can be placeholder text like "loren ipsum"). Don't use web.authenticated on this link.

Next, modify the M5T3 BeautifulSoup code so that it connects to this URL instead of the URL used in that example.

Confirm that you're able to webscrape data from your own web page.

[**M6LAB1**](https://faytechcc.blackboard.com/webapps/assignment/uploadAssignment?content_id=_6162842_1&course_id=_35059_1&group_id=&mode=view)

M6LAB1 - Repl.it Flask Twitter-Style app

For this assignment, we'll be following the instructions here:

[Building Websites and APIs — replit-py 3.2.2 documentation](https://replit-py.readthedocs.io/en/latest/web_tutorial.html#building-repltweet) <https://replit-py.readthedocs.io/en/latest/web_tutorial.html#building-repltweet>

Starting at "Building ReplTweet" and go to the end of the tutorial listed.

Rather than having to install the /static and /templates files yourself, you can instead fork this REPL, as it contains everything but the completed main.py:

<https://replit.com/@FTCCProgramming/repltweet#main.py>

Here's a very short video showing how I fork the REPL, to get you started.

<https://youtu.be/MPvLxVLa3aQ>

[**M6HW1\_Flask**](https://faytechcc.blackboard.com/webapps/assignment/uploadAssignment?content_id=_5950161_1&course_id=_35059_1&group_id=&mode=view)

Due 12/6. The purpose of this assignment is to get used to working within the constraints of a Flask application.

It is likely that your capstone project, if done in Python, will use a web-based front end (either Flask or Django), and so getting practice with the idiosyncracies will hopefully be useful.

1. open the link below

<https://blog.pythonanywhere.com/169/>

(opens in new window)

This assignment is divided into two parts.

**Part1: (100 points)**

1. follow the instructions provided in the link above and stop at where it says "**Pause for Breath..**." Make sure the application is running at this point

2. Right above where it says "Enter your numbers here", **add your first and last name followed by <br>**. Example of what the code should look like is shown below  . Make sure you save and reload the file and the application is displayed without errors

                 Enter your numbers:

**Part 2 : (25 points) - Bonus Points**

4.  follow the instructions provided after the part you completed in Part 1 and stop at where it says "**Processing Files**." Make sure the application is running at this point

5. Once completed, take a screen shot of the application created, your name should show, and submit the screen shot in the link below