**Python, *note to self...***

**Follow**[**these instruction**](http://stackoverflow.com/questions/15844905/how-to-stop-google-indexing-my-github-repository)**to prevent Google from indexing the content of these particular directories (so that future students can't cheat)!**

**Level of expertise**

Halfway Between Beginner and Intermediate. I also recently completed a 3 hour college course that specifically focused on python, and consider myself an intermediate level OOP generalist, so I'm ok with the Halfway Between Beginner and Intermediate classification.

**Disclaimer**

The projects here are all projects from that class. Most of the projects consist of code that tests functions/objects via imported code that the professor wrote. Usually, the challenge was to exactly reproduce a given output without modifying the imported code. The only tools used was Notepad++ and a command line. It really was a lot of fun.

I am absolutely in love with the way that python handles arrays, tuples and strings. It's like, easy. Also, using doctest showed me a completely new way of testing that I didn't even know existed.

Each directory contains these files:

* **Proj#.html** - The instructions for the actual challenge.
* **Proj#.py** - This is the professor written "driver" file that we are to use to produce a given output, as stated in the instructions.
* **Proj#Runner.py** - This is the "runner" file that I wrote to satisfy the challenge.

**Projects**

**Project 1**

***The challenge:*** "Write and submit a Python module named Proj01Runner.py that works in conjunction with the driver source code contained in the file named Proj01.py to produce the output shown in Figure 1 below"

***Unique topics include:***

* import directives
* classes and objects
* methods
* control flow
* lists, strings, and tuples
* turtle graphics

**Project 2**

***The challenge:*** "Write and submit a Python module named Proj02Runner.py that works in conjunction with the driver source code contained in the file named Proj02.py to produce the output shown in (A different drawing from the 1st project)".

***Unique topics include:***

* instance methods and instance variables

**Project 3**

***The challenge:*** "Write and submit a Python module named Proj03Runner.py that works in conjunction with the driver source code contained in the file named Proj03.py to produce one of the two output images shown in Figure 1 below. (The actual output that is produced is based on a random number and may be different from one run to the next.)"

***Unique topics include:***

* class methods and class variables

**Project 4**

***The challenge:*** "Write and submit a Python test file named Proj04Runner.txt that works in conjunction with the driver source code contained in the file named Proj04.py and the command-line command provided later to produce the output shown below on the command-line screen. (Make sure that your name appears in the error message as indicated below.)"

***Unique topics include:***

* Use of the doctest module to test the behavior of a function.

**Project 5**

***The challenge:*** "Write and submit a Python module named Proj05Runner.py that works in conjunction with the driver source code contained in the file named Proj05.py to produce the output shown below on the command-line screen for the values of the variables named str and subStr shown in the file named Proj05.py."

"Your module must contain a function named run, which receives an arbitrary string and an arbitrary substring as incoming parameters. The function must display the string and the substring on consecutive lines of output.

Then the run function must locate the substring in the string. It must extract and return that substring plus three characters on either end of the substring."

***Unique topics include:***

* String methods (see The Python Standard Library - 4.7.1 String Methods)

**Project 6**

***The challenge:*** "Write and submit a Python module named Proj06Runner.py that works in conjunction with the driver source code contained in the file named Proj06.py to produce the output shown below on the command-line screen"

***Unique topics include:***

* String separators
* Arrays

**Project 7**

***The challenge:*** "Write and submit a Python module named Proj07Runner.py that works in conjunction with the driver source code contained in the file named Proj07.py to produce the output shown below on the command-line screen for the values of the variables named str and index showing in the file named Proj07.py.

Your module must contain a function named run, which receives an arbitrary string and an arbitrary numeric index as incoming parameters. The run function must display the string and the index on consecutive lines of output.

Then the run function must break the string into two strings at the index with a one-character overlap and return the two strings as two elements in a list."

***Unique topics include:***

* list methods
* slicing strings

**Project 8**

***The challenge:*** "Write and submit a Python module named Proj08Runner.py that works in conjunction with the driver source code contained in the file named Proj08.py to produce the output shown below on the command-line screen for the values of the variables named myListA and char showing in the file named Proj08.py.

Your module must contain a function named run, which receives a list containing an arbitrary number of arbitrary strings as an incoming parameter. The run function also receives an arbitrary character as an incoming parameter. The run function must display the character and the list on consecutive lines of output.

Then the run function must return a list containing all of the strings from the original list that do not contain the character in the same order that they appear in the original list."

***Unique topics include:***

* The in operator
* The not operator

**Project 9**

***The challenge:*** "Write and submit a Python module named Proj09Runner.py that works in conjunction with the driver source code contained in the file named Proj09.py to produce the output shown below on the command-line screen for the values of the variables named myTuple and myWord showing in the file named Proj09.py.

Your module must contain a function named run, which receives a tuple containing an arbitrary number of arbitrary strings as an incoming parameter. The function also receives a string as an incoming parameter. The run function must display the string and the tuple on consecutive lines of output.

Then the run function must return a filter object containing all of the strings from the original tuple that contain the individual string parameter. It is possible that the order of the strings in the filter object may not match the order of the strings as displayed in the tuple."

***Unique topics include:***

* lambda
* filter
* filter object
* regular expressions

**Project 10**

***The challenge:*** "Write and submit a Python module named Proj10Runner.py that works in conjunction with the driver source code contained in the file named Proj10.py to produce the output shown below on the command-line screen for the values of the variables named myKeys and myValues showing in the file named Proj10.py.

More specifically, your module must contain a function named run, which receives three parameters. The first parameter is a tuple containing three byte-formatted strings that are to serve as keys in a dumb database (module dbm.dumb). The second parameter is a tuple containing three byte-formatted strings that are to serve as values for the keys on the basis of matching positions within the tuples. The third parameter is a string that is to be used as the name for the database.

The run function must create and populate the database using the data in the incoming parameters and then display the keys and values in the format shown above. (Note: the print order may vary from one run to the next.)"

***Unique topics include:***

* database creation and manipulation using the dbm.dumb module
* keys
* values

**Project 11**

***The challenge:*** "Write and submit a Python module named Proj11Runner.py that works in conjunction with the driver source code contained in the file named Proj11.py to produce the output shown below on the command-line screen for the values of the variables named myKeys and myValues showing in the file named Proj11.py."

***No unique topics, just a slightly more difficult challenge***

**Project 12**

***The challenge:*** "Write and submit a Python module named Proj12Runner.py that works in conjunction with the driver source code contained in the file named Proj12.py to produce the output shown below on the command-line screen."

***No unique topics, just a slightly more difficult challenge x 2***

I got a 100 in this class! Whoo-hoo!