**Assignment #3**

**Objective**

Continue learning the basics of Python programming!

**Due**

Monday at 9am on Jan 22nd.

**Background**

Please commit completed scripts to your GitHub repository in a directory named ‘Assigment03’. Simply send Dr. Ficklin a private message on slack once your assignment is ready.

This assignment will take a good amount of time. Please try to get started as soon as possible to ensure you have plenty of time to work through it***. Avoid trying to do the entire thing at one time***. Doing a few exercises and taking a break will help with memory retention. If you have questions, use Slack! Remember to post questions to the #assignments channel to give anyone the chance to respond.

Because this is the first time this course has been offered we need to know if this assignment becomes too much work over a 4-day period. If you find you are not able to work through everything because it is taking too long please let the instructors know. If we find that most of the students are struggling to finish the work in a reasonable amount of time then we will adjust the expectations. Not everyone will pick up programming at the same pace. You may find that you devote more or less time than other students.

**Tasks**

1. From Shaw’s *Learning Python the Hard Way*, read and complete exercises #18-23, 27-34 (pages 92-117 and 128-155. Skip exercises 24, 25 and 26).
   1. Be sure to create the programs the author instructs you to.
   2. Make sure that your output is like that shown in the “What You Should See” section that appears after each exercise. If you do not have the same results, then check your code, fix any mistakes and re-run until your output looks like those shown.
   3. At the end of each exercise is a “Study Drills” section.
      1. Some items in the Study Drills are meant to help you think through what you are doing. For these, take some time to think about the questions, but you do not need to write down answers or turn them in.
      2. If the Study Drill instructs you to write code, then please do so as instructed and turn that in.
   4. For the sake of time you can skip the ‘Break It’ sections when they appear.
   5. Finally, avoid the urge to skip over the “Common Student Questions” section. The answers to some of those questions may help you out!
   6. Remember:
      1. Be sure to save each exercise into a separate file. Each exercise indicates the name you should use (e.g. ex18.py for the first exercise).
      2. With the Shaw text it is easy to cut-and-paste the code and execute it. But as the author points out this does little to teach. Please take the time to type the code.

Questions:

What defines the end of a function?? (ie in R it’s the end })

Answer- it’s the indents- indents matter in this language

If you use \*args, does this mean you can put in as many arguments as you want?

What does f. do again?

Assignment 23:

* Why do we set error = sys.argv
* Soooo what if its not a line? Why not an if, else statement?
* Reads functions bottom up?
* Functions I don’t know:
  + Strip
  + Encode
  + Decode

Excercise 28

I always work from inside out and I always get the same answer