**GQ5426 – MAY WANDYEZ**

**Wayne State University**

**CSC 4110 - Software Engineering**

**Homework 1**

**INTRODUCTION:**

For ease of access, and of grading, the submission has been formatted to fit on this word document containing the description of the assignment.

**Directions:**

**Do all problems by the due date. Follow instructions explicitly. See submission requirements.**

**\*\*\*\* Adhere to ‘General Requirements’**

**Assignment**

**TASK ONE**

**Read support documents carefully to Install Python. And then install Visual  
 Studio Code (VSC).Finally, configure VSC for Python.**

**TASK TWO**

**Go to following link, and list three interesting aspects of  
 Python. https://www.pluralsight.com/blog/software-development/why-  
 python (Links to an external site.) (or find similar article)  
 Note: if link doesn't work, do a general Google search.**

**1.)Python is more readable than other programming languages – and doesn’t require semicolons on every line of code. I cannot emphasize enough how appealing this is to me considering most of my error reports are from invalid syntax due to lacking a semicolon.**

**2.)Python is the second most used language on GitHub. This means if you want to integrate an open source project into your work, it helps to know python.**

**3.)Python is cross platform – if you have the interpreter, it can be run. There is also a version of python called ‘MicroPython’ that cuts out many features of python such as deep copies so that it can run on small devices.**

**TASK THREE  
Go to ‘command line interface’ and then get to “Interactive Mode.”  
 Type "import this"  
 Pick three statements that seem interesting; explain them.**

**Typing ‘import this’ has summoned the poem “The Zen of Python” by Time Peters.**

**Here are some statements that I found interesting:**

**1.)Flat is Better Than Nested**

**This sentence means that nesting if statements and functions is confusing and difficult to understand and may lead to slower coding due to overcomplicating the process. This can be solved by ‘flattening’ the code and trying to avoid nesting wherever possible.**

**2.)If the implementation is hard to explain, it’s a bad idea.**

**This means that difficult and convoluted implementations will likely lead to problems – making things complicated usually means making things unnecessarily complicated.**

**3.)Readability counts**

**This emphasizes that the purpose of code is to be understood by humans, and therefore a programming language should reflect that. A language for computers is great for computers, however it is difficult from the humans that have to program them because humans are not computers. Furthermore, code will at some point have to be examined by a human – at which point readability matters in order to understand what is programmed and how to work with it.**

**TASK FOUR**

**Go to:  
 https://www.python.org/  
 Go back to homepage on the python.org site. Try scripts that are shown in  
 boxes 1 through 5. Paste code here:**

**I am genuinely confused as to what you are asking for this task. It seems like you want me to literally copy the scripts shown on the python.org website. I do not understand why – however I will blindly comply anyway.**

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer code

Description automatically generated

**TASK FIVE - please see Documentation, under Assignment Guidelines**

**Write a python script that accepts 'any' user input at least 9 characters long that prints three characters, in three lines, in uppercase letters. USE ONLY METHODOLOGIES GONE OVER IN WEEK ONE VIDEOS.**

**TESTED INPUT must be a combination of upper- and lower-case letters. All must be grouped to be aligned and on a separate line.**

**One input statement and ONE print statement.**

**Example:**

**asdFghjklvb**

**Becomes:**

**asd   
 fgh   
 jkl**

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

**SUBMISSION REQUIREMENTS:**

**Note1: for all screen shots/ screen scrapes, paste them all in THIS document.**

**Note2: see General Requirements**

**You will be submitting THIS document.**

**Assignment Guidelines**

1. Submit a screenshot of the code, itself for all four routines (must be commented per requirements)
2. A ‘screen scrape’ of the output for all routines
3. As required, documentation describing your opinion on the most efficient programming method.
4. Also, in your documentation, describe an ALTERNATE way you COULD have performed this if you had more time and unlimited funds.
5. See “General Requirements”

**General Requirements:**

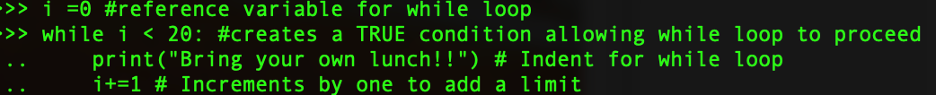
**Add labeling/ comments (name, date, revision #); add in-line requirements where appropriate (such as syntax usage).**

#Indicate coding begin and end

Example acceptable code comment:

**# Revision number BEGIN/ START DATE**

**## Begin John D. Student here (date)**



**# Revision number FINAL DATE**

**## End John D. Student here**

**# Group / manager/ lead tech/ project # ←-Where appropriate**

**Adhere to the following coding style (from PEP8):**

1. Wrap lines so that they don’t exceed 79 characters.

2. Use blank lines to separate functions and classes, and larger blocks of code inside functions

3. When possible, put comments on a line of their own.

4. Where appropriate, name your classes and functions consistently; the convention is to use UpperCamelCase for classes and lowercase\_with\_underscores for functions and methods.