Table Python Coding Standard

Create by Dongning Li

Purpose	To guide the development of python programs
Program Headers	Begin all programs with a descriptive header.
Header Format	``` ''' Create by Dongning Li (Donnie) Course Name: SWE 5001 Software Engineering 1 Assignment Number: Assignment 1 Create Date: Sep/23/2016 ''' ```
Listing Contents	Provide a summary of the listing contents.

Contents Example	```
	'''
	
	Listing Contents:
	Reuse instructions
	Modification instructions
	Compilation instructions
	Import
	Function Define:
	  getMean()
	  getMedian()
	  Main()
	Call main function to run whole program:
	  Main()
	
	'''
	``

Reuse Instructions	• Describe how the program is used. Provide the declaration format, parameter values and types, and parameter limits. • Provide warnings of illegal values, overflow conditions, or other conditions that could potentially result in improper operation.
Reuse Example	``` ''' getMax(aList) Purpose: get the max number of the arraylist Limitation: each element in aList should be number and the length of aList should NOT 0. Return: max number if each elements in aList is number and the length of aList is NOT 0, else return ERROR. ''' ```
Identifiers	Use descriptive names for all variables, function names, constants, and other identifiers. Avoid abbreviations or single-letter variables like I/L/O. Name variables camelCase. Function name should start with a verb.

Identifier Example	```
	studentNum = 0 #This is GOOD
	x4 = 0 #These is BAD
	stunumber = 0 #These is BAD
	``

Comments	• Document the code so that the reader can understand its operation.

	• Comments should explain both the purpose and behavior of the code. • Comment variable declarations to indicate their purpose.
Good Comment	`if (recordCount > limit) #have all the records been processed?`
Bad Comment	`if (recordCount > limit) #check if recordCount greater than #limit`

Table Python Code Review Checklist

Create by Dongning Li
PROGRAM NAME AND #:   SWE1ASGA1

Purpose	To guide you in conducting an effective code review.				
General	- As you complete each review step, check that item in the box to the right. - Complete the checklist for one program unit before you start to review the next.				
Complete	Verify that the code covers all the design.				
Import	- Check imports are at the top of the file. - Check imports are on separate lines. - Verify that imports are complete.				
Initialization	Check variable and parameter initialization: - at program initiation - at start of while loop - at function/procedure entry				
Calls	Check function call formats: - parameters - use of '&'				
Names	Check name spelling and use: - is it consistent? - is it within declared scope? - do all variables and functions named in camelCase? - do function name starts with a verb?				
Loop	- Check variable used for while loop counting initialize - Check "while loop" adds step each loop manually while "for loop" NOT				
String	- Don't use the fact that empty sequences. Use if sequence/ if not sequence.				
Boolean Comparison	Don't compare Boolean values to True or False using ==				
Output	- The 4-space rule is optional for continuation lines. - use [''] instead of [""] for printing plain text.				
Return	No value is returned should explicitly state this as 'return None'				
Logic Operators	Verify the proper use of ==, =, ||, and so on.				
Line by Line Check	Check every LOC for - instruction syntax and - proper indentation. - NO semicolon exists				

Standards	Ensure that the code conforms to the coding standards.				
File Open and Close	Verify that all files are - properly declared, - opened - read/ written.				