**Good Coding Styles and Standards Guide**

Good clean code should follow the following guidance:

1. Naming Conventions: All variable names should use Camel Case naming conventions. Variable names should not exceed 20 characters. Functions should be named using snake case and should also not exceed 30 characters in length. Function names should be specific towards what the functions purpose is.
2. Indentation: All indentation must be in groups of 4 spaces. Please be aware that the tab key does not always indent 4 spaces and so the spacebar should be used instead.
3. Line Spacing: There should be 1 line of spacing between functions inside of a class. Between separate classes there should be 2 lines of spacing.
4. Docstrings: All classes and functions and the program module itself must have a docstring explaining about the function, class or module. In addition, the module docstring must include name of the programmer, the date it was initially created and the title of the project the module is for. A docstring must be indented to the correct level in each class, module or function.
5. Line Length: A line of code should not be more than 80 characters in total. Do not include comments when checking this limit.
6. Comments: comments should be preceded by 2 spaces before and one space after the ‘#’ symbol.
7. Line continuations: Line continuations should be used whenever required. It is advisable to add a comment of ‘# continued on next line’ whenever a line continuation is to be used. In addition, correct indentation must be followed for the new line to bring it in line with the previous line of code.
8. SOLID: Make sure to use SOLID principles as much as possible. Single Responsibility, Open/Closed Principle are the most important and should be considered at all times when writing code.
9. All code must be DRY. Any repetition of code should be cleaned using DRY principles and functions named accordingly.
10. Extracting Variables: Where possible. Never return results directly always store the results in a suitably named variable and return said variable.