Coding Standards

Code standards are rules, techniques, and guidelines to [create cleaner](https://community.codacy.com/t/tips-for-writing-clean-code/842), better readable, and more efficient code with minimal bugs and errors.

1. **Limited use of globals:**  
   These rules tell about which types of data that can be declared global and the data that can’t be.
2. **Standard headers for different modules:** For better understanding and maintenance of the code, the header of different modules should follow some standard format and information.

* Name of the module
* Date of module creation
* Author of the module
* Modification history
* Synopsis of the module about what the module does
* Different functions supported in the module along with their input output parameters
* Global variables accessed or modified by the module

1. **Naming conventions for local variables, global variables, constants and functions:**

* Meaningful and understandable variables name helps anyone to understand the reason of using it.
* Local variables should be named using camel case lettering starting with small letter (e.g. **localData**) whereas Global variables names should start with a capital letter (e.g. **GlobalData**). Constant names should be formed using capital letters only (e.g. **CONSDATA**).
* It is better to avoid the use of digits in variable names.
* The names of the function should be written in camel case starting with small letters.
* The name of the function must describe the reason of using the function clearly and briefly.

1. **Indentation:**

* There must be a space after giving a comma between two function arguments.
* Each nested block should be properly indented and spaced.
* Proper Indentation should be there at the beginning and at the end of each block in the program.
* All braces should start from a new line and the code following the end of braces also start from a new line.

1. **Avoid using a coding style that is too difficult to understand**
2. **Length of functions should not be very large**