**Coding Standards**

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

Some of the naming conventions are given below:

Meaningful and understandable variable names help anyone to understand the reason for using it.

Local variables should be named using camel case lettering starting with a 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 a camel case starting with small letters. Underscore can be used incase of two words

The name of the function must describe the reason for using the function clearly and briefly.

The name of the class should start with a capital letter

Avoid using an identifier for multiple purposes:

Each variable should be given a descriptive and meaningful name indicating the reason behind using it. This is not possible if an identifier is used for multiple purposes and thus it can lead to confusion for the reader. Moreover, it leads to more difficulty during future enhancements.

**Indentation:**

Proper indentation is very important to increase the readability of the code. For making the code readable, programmers should use White spaces properly. Some of the spacing conventions are given below:

There must be a space after giving a comma between two function arguments.

• It's better to write elements in the next line so that all elements are arranged vertically so that it is easy to read and find elements

**Code should be well documented:**

The code should be properly commented for easy understanding. Comments regarding the statements increase the understandability of the code.

At the top of the code we should describe about the program

Unnecessary commenting should be avoided

**Dry (Don't repeat yourself) :**

The same piece of code should not be repeated since it increases length of the code making ít hard to read

Avoid writing lengthy code

Unnecessary code should not be written

Use of list comprehension is encouraged

Avoid too many levels of nesting

**Length of functions should not be very large:**

Lengthy functions are very difficult to understand. That’s why functions should be small enough to carry out small work and lengthy functions should be broken into small ones for completing small tasks.