

**E-Voting Portal**

Software Coding Standards

**Group Members -**

**1.G Srisha Anagh (IIT2016030)**

**2.Sai Charan Teja (IIT2016039)**

**3.Anmol Singh Sethi (IIT2016040)**

**4.Manavdeep Singh (IIT2016042)**

**5.Nairit Banerjee (IIT2016505)**

Table of Contents

1. **General Rules**
2. **Indenting and Whitespaces**
3. **Control Structures**
4. **Function Calls**
5. **Object Creation**
6. **Documentation**
7. **Naming Conventions**
   1. **Constants**
   2. **Variables**
8. **Classes**

The language used for the development of the Website is Flask in the backend. We have followed the following conventions while writing the application code.

1. General Rules :

* The python language is used for advanced problem solving. We use the language for the backend operations.
* The semicolon used in other common programming languages is not allowed in python.
* After an if statement or declaration of a function, a colon ‘:’ is used to specify about the function. Further all the statements till the end of the function are one-tab away from the function declaration to depict the block.
* Variable declarations are not required. It is by default dynamic in allocation.

For Example :-

def function\_name():

variable1 = 0

1. Indenting and Whitespaces :

* Python is very particular towards indentation. A wrong indentation is considered as an Error through the Exception feature of IndentError.
* There should be no trailing white spaces in a statement.
* Any two words of the code shouldn’t have more than one whitespace between them.
* Since a standard indentation is required for the functioning of the Flask code, A basic understanding of indentation is required.
* The variables indentation is also necessary. It is made sure that not more than one statement is written in one line.

1. Control Structures :

* Control Structures are very necessary in python or else the readability will get drastically affected. For statements like if, else, for etc, the indentation shows the control structure. We use the colon to depict the control statement and the next further instructions whichever are under a one-tab away indent than the control statement, are considered to be the block that is controlled.
* Multiple Conditions are written in a single line in a control structure.

For eg,

if (condition1 || condition2):  
 action1  
  
else if (condition3 && condition4):  
 action2

else :  
 defaultaction

1. Function Calls :

Functions should be called with no spaces between the function name, the opening parenthesis, and the first parameter; spaces between commas and each parameter, and no space between the last parameter and the closing parenthesis. Here's an example:

db.insert\_data(name,time,date,freq)

1. Object Creation :

* Instance variables are owned by instances of the class. This means that for each object or instance of a class, the instance variables are different.
* Unlike class variables, instance variables are defined within methods.
* The object created points towards a class. This makes access to class methods possible.

1. Documentation :

Proper commenting is done before the start of every function and class to explain the purpose of the construct as well as the variables used in it.

1. Naming Conventions :
   1. General : Avoid using names that are too general or too wordy. Strike a good balance between the two.
   2. **Package :** Package names should be all lower case. When multiple words are needed, an underscore should separate them.
   3. Variable Names : Instance variable names should be all lower case. Words in an instance variable name should be separated by an underscore.
   4. Classes Names : Class names should follow the UpperCaseCamelCase convention. Python’s built-in classes, however are typically lowercase words. Exception classes should end in “Error”.

## Methods Names : Method names should be all lower case. Words in an method name should be separated by an underscore. Non-public method should begin with a single underscore. If a method name needs to be mangled, two underscores may begin its name.

1. Classes :

Classes are created to make certain collections of data as a particular user-defined type of structure. We can create objects as instances of the class and store data as collections. This feature of python is similar to Java and C++. We use the classes to make the code more readable. Standard protocols of the naming conventions of classes were used wherever necessary.

The naming of the instance variables and classes were according to standard protocol to increase readability among programmers and to assure reliability as well. Since references are different from pointers they do not cause errors such as Null access errors.