

## PEP8 in Python

What is PEP8?

PEP (Python Enterprise Proposal): ~~is~~ writing code with proper logic ~~and~~ is a key factor in programming, but there are many other factors that are important.

Why is PEP8 Important?

It enhances the readability of a code ~~which~~. but why is readability important? Writing a code can take a few minutes, a few hours, or a whole day, but once written, we cannot rewrite it again, also we need to read the code again. The code should reflect the meaning of each line. That's why readability is important.

Naming Convention

When we write a code, we need to assign name to many variables, functions, classes, packages, and a lot of more things. Selecting a proper name will save time and energy. When we look at the file after some time, we can easily recall what a certain variable, function or class represents.

Example.

- Single lowercase letter.  
→ `a = 10.`
- Single uppercase letter.  
→ `A = 10`
- lowercase.  
→ `var = 10.`
- lowercase\_with\_underscore.  
→ `number_of_apple = 5`
- UPPERCASE  
→ `var = 6.`
- UPPERCASE\_WITH\_UNDERSCORE.  
→ `NUM_OF_CAR = 20.`
- Capitalized words → Number of Books = 100.

Code Layout.Indentation

The Indentation is used to define the code block in Python. It is the important part of the Python programming language and it determines the level of lines of codes. Generally, the use of 4 space for Indentation.

Example.

```
x = 5
```

```
if x == 5:
```

```
    print('x is larger than 5')
```

The indented space print statement will get executed if the condition of if is true. This indentation defines the code block and tells us what statement executed when a function is called or condition trigger.

Tab vs Space.

Tabs are used to provide the consecutive space to indicate the indentation, but white space are the most preferable.

Indentation following line break.

It is essential to use indentation when using line continuation. It keeps the line to fewer than 79 characters. It provides flexibility to determine between two codes and a single line that extends to two lines.

Example.

```
obj = func_name (argument one, argument two,
                 threeargument two, argument four)
```

```
→ def function name (
```

```
    argument one, argument two, argument three,
    argument four)
```

```
    print (argument two)
```

## Doc String

Usage of tripple quotes to define a single line or multiple line quotes. Basically, these are used to describe the function of a particular program.

Example.

```
def add(a,b):
    """This is simple add method"""
    """This is
    a
    simple add program to add the two numbers"""
```

## Importing Module

→ `import pygame`  
`import os`  
`import sys`

Import is written at the start of the code or just after a module comment.

## Close Bracket Braces

We can break lines inside parantheses, brackets using the line continuation, ~~for 8 on~~

→ Example

`list_numbers = [`

`5,4]`

`2,3,6`

`7,8,9`

`] "`



### Comments.

Comments are the best way to explain the code. When the code is documented with proper comments, anyone can easily understand the code.

- Start with capital letter, and write complete sentence.
- Update the comment in case of a change in code.
- Limit the line length of comments and doestings to 72 characters

### Block Comment.

Block comments are the good choice for the small section of code. Such comments are useful when we write several line codes to perform a single action such as iterating a loop.

### Inline Comments.

Inline Comment are used to explain the single statement in a piece of code. We can get the idea of why we wrote that particular line of code.

Start with `##` and single space.

Use it carefully

We should separate the inline comments on the same line as the statement referred.