

PYTHON - PEP 8.

- What is PEP 8?

→ PEP in Python stands for Python Enhancement Proposal. The PEP 8 is basically Python's style guide. It helps in writing code to specific rules making it helpful for large codebases having multiple writers by bringing a uniform and predictive writing style. PEP or Python Enhancement Proposal is a set of rules that specify how to format Python code for maximum readability. It is an official design document that provides relevant information to the Python community, such as describing a new Python feature or a Python process. PEP 8 is an important document that includes the style guidelines for Python code. Anyone who wishes to contribute to the Python open-source community must strictly abide by these guidelines.

→ Python language is one of the best deciphered dialects. At the point when you compose a Python language content, it doesn't have to get ordered before execution. Hardly any other deciphered dialects are PHP and Javascript.

Advantages of Python language

Python language is a dynamic language. It implies that you don't have to specify the information sort of factors during their affirmation. It permits to set factors like var1 = 101 and var2 = "You are a designer." with no mistake.

- Python language upholds object oriented programming as you can characterize classes alongside the piece and legacy. It doesn't utilize access specifiers like open or private).
- Capacities in Python language resemble top of the lines objects. It recommends you can relegate them to factors, come back from different techniques and go as contentions.
- Creating utilizing Python language is speedy however running it is frequently more slow than arranged dialects. Fortunately, Python language empowers to incorporate the "C" language augmentations so you can

streamline your contents.

- Python language has a few uses like electronic applications, test mechanization, information displaying enormous information investigation and considerably more. Then again you can use it as a "stick" layer to work with different dialects.
- In order to compile new extensions without any errors, compiling and linking is used in Python. Linking initiates only and only when the compilation is complete.
- In the case of dynamic loading, the process of compilation and linking depends on the style that is provided with the concerned system. In order to provide dynamic loading of the configuration setup files and rebuilding the interpreter the Python interpreter is used.

The dynamic modifications made to a class or module at runtime are termed as monkey patching in Python. Consider the following code snippet:

```
#m.py
class MyClass:
    def f(self):
        print "f ()"
```

We can monkey-patch the program something like this:

```
import m
def monkey_f(self):
    print "monkey-f ()"
m.MyClass.f = monkey_f
obj = m.MyClass()
obj.f()
```

The output for the program will be monkey-f().

The examples demonstrate changes made in the behaviour of f() in MyClass using the function we defined i.e. monkey-f() outside of the module m.

→ PEP 8 enhances the readability of the Python code, but why is readability so important?

Creator of Python, Guido van Rossum said, "Code is much more often than it is written." The code can be written in a few minutes, a few hours, or a whole day but once we have written the code, we will never rewrite it again. But sometimes we need to read the code again and again.

At this point, we must have an idea of why we wrote the particular line in the code. The code should reflect the meaning of each line. That's why readability is so much important.

• Naming convention

When we write the code, we need to assign name to many things such as variables, functions, classes, packages, and a lot more things.

Selecting a proper name will save time and energy. When we look back

to the file after sometime, we can easily recall what a certain variable, function, or class represents! Developers should avoid choosing inappropriate names.

The naming convention in Python is slightly messy, but there are certain conventions that we can follow easily.

single lowercase letter

a = 10

single uppercase letter

A = 10

lowercase

var = 10

lower-case-with underscores

number_of_apple = 5

UPPERCASE

VAR = 6

streamline your contents.

- Python Language has a few uses like electronic applications, test mechanization, information displaying enormous information investigation and considerably more. Then again you can use it as a "stick" layer to work with different dialects.
- In order to compile new extensions without any error, compiling and linking is used in Python. Linking initiates only and only when the compilation is complete.
- In the case of dynamic loading, the process of compilation and linking depends on the style that is provided with the concerned system. In order to provide dynamic loading of the configuration setup files and rebuilding the interpreter, the Python interpreter is used.