Sanskar Singh
17/2/22. FYIT-86.

PEP 8 in python

* The PEP is an abbreviation form of Python Enterprise Proposal. writing code with proper logic is a key facto of programming, but many other important factors can affect the code's quality. The developer's coding style makes the code much reliable, and every developer should keep in mind that python strictly follows the way of order and format of the string.

Adaptive a nice coding style makes the code more readable. The code becomes easy for end-user.

PEP-8 is a document that provide various guidelines to write the readable in Python. PEP 8 describ how the developer can write beautiful code. It was officially written in 2001 by Guido van Rossum, Barry Warsaw, and Nick Coghlan. The main aim of PEP is to enhance the readability and consistency of code.

Sanskar Singh

FYIT-86.

* why PEP 8 is important?

PEP 8 enhances the readability of python code, but why is readability so important? Let's understand this concept.

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 code. The code should reflect the meaning of each line. That's why readability is so much important.

We will describe few important guidelines for writing effective code that can be read by others as well.

Sanskar Singh
F417-86.

PAGE NO.:
DATE: / /

* 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. Let's see the following naming convention.

Example-

Single lowercase letter:

```
a=10
```

Single uppercase letter:

```
A=10
```

Sanskar Singh
FYIT - 86

Lowercase:

var = 10

lower_case_with_underscores

number_of_apple = 5

Uppercase:

VAR = 6

UPPERCASE_WITH_UNDERSCORES

NUM_OF_CARS = 20

Camelcase

NumberOfBooks = 100

* Name

Sanskar. Singh
PAGE NO.:
DATE: / /

FYIT-86.

* Name Style:

1) Function:
we should use lowercase words or seperates words by the underscore.

eg:
myfunction,
my_function

2) Variable:
we should use a lowercase letter, words, or seperate words to enhance the readability.

eg:
a,
var,
variable_name

3) class:
The first letter of class name shoul be capitalized; use camel case. Do not seperate words with the underscore.

eg:
MyClass,
Form,
Model

Sanskar Singh
FYIT-86.

4) Method:

we should use a lowercase letter, words, or seperate words to enhance the readability.

eg:

class_method,
method.

5) Constant:

we should use a short, uppercase letter, words or seperate words to enhance the readability.

eg:

MY CONSTANT,
CONSTANT,
MY_CONSTANT

6) Module:

We should use a lowercase letter, words, or seperate words to enhance the readability:

eg:

Module_name.py,
module.py

Sanskar · Singh
FYIT-86.

PAGE NO:
DATE: / /

7) Package:
We should use a lowercase letter, words, or seperate words to enhance the readability. Do not seperate words with the underscor

eg:
package,
mypackage.

Above are common naming conventions that are useful to beautify the Python code. For additional imporovement, we should choose the name carefully

* Code Layout.

The code layout defines how much the code is readable. In this section, we will learn how to use whitespace to improve code readability.

* Intentation:

Unlike other programming languages, the indentation is used to define the code block in Python.

Sanskar Singh

FYIT-86.

The indentations is used to define the code block in Python. The indentations are important part of the python programming language and it determines the level of lines of code. Generally, we use the 4 space for indentation. Let's understand the following example:

eg:

```
x = 5
if x == 5:
    print("x is larger than 5")
```

In the above example, the indented print statement will get executed if the condition of if statement is true. This indentation defines the code block and tells us what statements execute when a function is called or condition trigger.

* Tabs vs. Space.

We can also use the tabs to provide the spaces to indicate the indentation, but whitespaces are the most preferable.

Sanskar Singh
F41T-86.

* Indentation following line Bre

It is essential to use indentati when using line continuations t keep the line to fewer than 79 charachters. It provides the flexibility to determining the between two lines of code and a single line of code that extends two lines. Let's understand the following example:

example:

```
# Correct way


# Alligned with opening delimite
obj= func_name(argument_one,
     argument_two, argument_
     three, argument_four
```

Sanskar Singh
FYIT-86.

we can use the following structure:

```
# first line doesn't has any argument

# we add 4 spaces from the second line to discriminate arguments from the rest.

def function_name (

    argument_one, argument_two,
    argument_three, argument_four):

print(argument_two)


# 4 space indentation to add a level.

foo = long_function_name(
  var_one, var_two
   var_three, var_four)
```

Sanskar Singh
FYIT-86.

BRIGHT
PAGE NO.:
DATE: / /

* Comments.

Comments are integral part of any programming language. These are the best way to explain the code. when we documented our code with the proper comments anyone can able to 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 docstrings to 72 characte

* Conclusion.

We have discussed the PEP 8 guidelines to make the code remov ambiguity and enhance readabili These guidelines improve the code especially when sharing the code with potential employees or collaborators.

Sanskar Singh
F417-86.

* 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. Let's see the following naming convention.

Example -

Sanskar Singh

FYIT-86

Lowercase:

```
Var = 10
```

Lower_case_with_underscores

```
number_of_apple = 5
```

Uppercase:

```
VAR = 6
```

UPPERCASE_WITH_UNDERSCORES

```
NUM_OF_CARS = 20
```

Camelcase

```
NumberOfBooks = 100
```

* Name