PRACTICAL 7: Implementing coding practices in Python using PEP8.

- What is PEP-8?

- The PEP is an abbreviation form of Python Enterprise Proposal. Writing code with proper logic is a key factor 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. PEP 8 is a document that provides various guidelines to write the readable in Python. PEP 8 describes how the developer can write beautiful code. The main aim of PEP is to enhance the readability and consistency of code.

Here are the most important points extracted for every Python developer read it as follows:

1)Use 4-space indentation and no tabs: The 4-space rule is not always mandatory and can be overruled for continuation line.

2)Use docstrings: There Are both single and multi-line docstrings that can be used in python.

3)Wrap lines so that they don't exceed 79 characters**:** The Python standard library is conservative and requires limiting lines to 79 characters. The lines can be wrapped using parenthesis, brackets, and braces. They should be used in preference to backslashes.

4)Use of trailing commas: This is not mandatory except while making a tuple.

5)Naming conventions: There Are few conventions that should be followed in order to make the program less complex and more readable. At the same time, the naming conventions in Python are a bit of mess, but here are few conventions that can be followed easily. There is an overriding principle that follows that the names which are visible to the user as public parts of API should follow conventions that reflect usage rather than implementation. In addition to these few leading or trailing underscores are also considered.
Single_trailing_underscore: used to avoid conflicts with Python keyword.
_double_leading_underscore: when naming a class attribute, invokes name mangling.

6)Characters that shouldn't be used for identifiers:'1' (lowercase letter el), 'O' (uppercase letter Oh), or 'I' (uppercase letter eye) as single character variable names as there are similar to the numerals one and zero.

IT TOOLS PRACTICAL 7 ROHIT MOURYA FYCS-56

7)Name your classes and functions consistently: The convention is to use Camel case for classes and lower_case_with_underscores for functions and methods. Always use self as the name for the first method argument.

8)Don’t use non-ASCII characters in identifiers if there is only the slightest chance people speaking a different language will read or maintain the code.

9)Use python’s default UTF-8 or ASCII encodings and not any fancy encodings, if it is meant for international environment.

10**)** Use spaces around operators and after commas, but not directly inside bracketing.