Implementing coding practical in python using pep8 .

PEP 8, sometimes spelled PEP8 or PEP-8, is a document that provides guidelines and best practices on how to write Python code. It was written in 2001 by Guido van Rossum, Barry Warsaw, and Nick Coghlan. The primary focus of PEP 8 is to improve the readability and consistency of Python code.

PEP stands for Python Enhancement Proposal, and there are several of them. A PEP is a document that describes new features proposed for Python and documents aspects of Python, like design and style, for the community.

This tutorial outlines the key guidelines laid out in PEP 8. It's aimed at beginner to intermediate programmers, and as such I have not covered some of the most advanced topics. You can learn about these by reading the full PEP 8 documentation.

- Write Python code that conforms to PEP 8
- Understand the reasoning behind the guidelines laid out in PEP 8
- Set up your development environment so that you can start writing PEP 8 compliant Python code

- Naming Styles

type	**Naming Convention**	**Examples**
Function	Use a lowercase word or words. Separate words by underscores to improve readability.	myfunction, my_functio
Variable	We should use a lowercase letter, words, or separate words to en	a, var, readability. variable_ name
Class	The first letter of class name should be capitalized; use camel case. Do not separate words with the underscore.	MyClass, Form, Model
Method	We should use a lowercase letter, words, or separate words to enhance readability.	class_method, method
Constant	We should use a short, uppercase letter, words, or separate words to enhance the readability.	MYCONSTANT, CONSTANT, MY_CONSTANT
Module	We should use a lowercase letter, words, or separate words to enhance the readability.	Module_name.py,

		module.py
Package	We should use a lowercase letter, words, or separate words to enhance the readability. Do not separate words with the underscore.	package, mypackage,