BRIGHT
PAGE NO.:
DATE: / /

Govind Jangid.
Rollno1-115

IT TOOLS PEP8 ASSIGNMENT

What is Pep?

- The PEP is an abbreviation from of phython 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 developing code style makes the code much reliable.
- PEP 8 is a document that provides various guidelines to write the readable in phython.
- PEP 8 describes how the developer can write the Beautiful code.
- It was officially written in 2001 by Guido van Russum, Barry warsaw and Nick Coghlan.

- CODE Layout:

- INDENTATION :-
- use 4 spaces per Identation level

- unlike the another programming language the indentation is used to define the code block in phython objects.
- The indentation part of the phython programming language and it determines the level of lines of code

. EG: x = 5
if x == 5
Print ("x is large than 5")

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

- TABS OR SPACES

- Spaces are the prefered indentation on method.
- Tabs should be used solely to remain consistent with code that is ~~at~~ already indented with tabs.
- phython disallows mixing tabs and spaces for indentations.

DATE: / /

- MAXIMUM LINE LENGTH.

- Limit all lines to a maximum of 19 characters.
- For flowing long blocks of text with fewer structural restriction (docstring or comments) the line length should be limited to 12 characters.
- Some teams strongly prefer a long line length
- For code maintained exclusively or primary by a team that can reach agreement on this length, limit upto 99 characters, provided that comment and docstring are still wrapped at 72 characters.
- The python standard library is conservation and require limiting lines to 79 characters.

- SHOULD ~~BE~~ A LINE BREA BEFORE OR AFTER A BINARY OPERATOR

- The lines before or after a binary operator is a tradition approach
- But its ~~after~~ affects the readability extensively because the operators scattered across the section.

- EG: Total - marks = (English - marks +
maths - marks +
Science - marks +
biology - marks +
Physics - marks)

Python allows us to Break line before or after a binary operator, as long as the convention is consistent locally.

- IMPORTING MODULE

- we should import the module in the separate line as follows:-

Eg:- from subprocess import popen, Pipe
The import statement should be written at the top of the file or just after any module comment.
Absolute imports are just after any module comment. Absolute import are the recommended because they are more readable and tend to be better behaved.

import mypkg.sibling

from mypkg or sibling
from mypkg import example

However, we can use the explicit relative import instead of absolutes, import especially dealing with complex package

- BLANK LINES
- Blank lines can be improved the readability of python code.
- If many lines of code bunched together the code will become harder to read
- we can remove this by using the many blank vertical line, and the read might need to Scroll more than necessary.
- Top level functions and classes with two lines! put extra vertical space around them so, that it can be understandable.

EG :- Class First class:
pass

class Second class
pass

def main_function():
return none