

[Python Enhancement Proposals \(PEPs\)](#)

PEP-8 and PEP-257 are essential Python Enhancement Proposals (PEPs) that provide guidelines for writing Python code and docstrings. They help maintain consistency, readability, and standardization in Python projects.

[PEP-8: Style Guide for Python Code](#)

1. Code Layout

1.1. Indentation

- Use 4 spaces per indentation level.
- Avoid using tabs for indentation.

```
python

def function_example():
    if True:
        print("Indentation with 4 spaces")
```

1.2. Line Length

- Limit lines to 79 characters.
- For long lines, use parentheses, backslashes, or implied line continuations.

```
python

# Using parentheses
long_variable = (
    "This is a long string that is broken into "
    "multiple lines using parentheses."
)
```

1.3. Blank Lines

- Surround top-level functions and class definitions with two blank lines.
- Use one blank line to separate methods within a class.

```
python

class MyClass:
    def method_one(self):
        pass

    def method_two(self):
        pass
```

1.4. Imports

- Imports should:
 - Be on separate lines.
 - Follow this order:
 1. Standard library imports
 2. Related third-party imports
 3. Local application/library imports

```
python

import os
import sys

from third_party import module
from my_project import my_module
```

2. Naming Conventions

2.1. Variables and Functions

- Use snake_case for variables and functions.

python

```
def calculate_area(radius):  
    area = 3.14 * radius * radius  
    return area
```

2.2. Classes

- Use CamelCase for class names.

python

```
class MyClass:  
    pass
```

2.3. Constants

- Use UPPER_CASE for constants.

python

```
PI = 3.14
```

3. String Quotes

- Use single (') or double (") quotes consistently.

python

```
name = "John"
```

4. Whitespace

- Avoid extra spaces around operators and brackets.

Good:

```
python

a = b + c
```

Bad:

```
python

a = b  +  c
```

5. Comments

- Use comments sparingly and keep them concise.

5.1. Block Comments

- Explain the logic or purpose of the code.

```
python

# Calculate the area of a circle
radius = 5
area = 3.14 * radius ** 2
```

5.2. Inline Comments

- Use inline comments sparingly and add at least two spaces before the `#`.

```
python

x = x + 1  # Increment x
```

6. Avoid Trailing Whitespace

Remove any unnecessary whitespace at the end of a line.

PEP-257: Docstring Conventions

1. General Guidelines

- Use triple double quotes (""") for docstrings.
- All public modules, functions, classes, and methods should have docstrings.

```
python

def example_function():
    """This is a one-line docstring."""
    pass
```

2. Module Docstrings

- Describe the module's purpose and contents.

```
python

"""This module provides utility functions for mathematical operations."""
```

3. Function and Method Docstrings

- Include a brief description, arguments, return value, and exceptions (if applicable).

```
python

def add(a, b):
    """
    Add two numbers.

    Args:
        a (int): The first number.
        b (int): The second number.

    Returns:
        int: The sum of the two numbers.
    """
    return a + b
```

4. Class Docstrings

- Describe the purpose of the class and its methods.

```
python

class Calculator:
    """
    A simple calculator class.

    Methods:
        add(a, b): Returns the sum of two numbers.
        subtract(a, b): Returns the difference between two numbers.
    """
    def add(self, a, b):
        """Add two numbers."""
        return a + b

    def subtract(self, a, b):
        """Subtract two numbers."""
        return a - b
```

5. Multi-Line Docstrings

- Use a one-line summary followed by a blank line and detailed description.

```
python

def complex_function():
    """
    Perform a complex calculation.

    This function implements a multi-step process to calculate
    a complex mathematical result. It takes into account several
    factors, including environmental conditions and input variables.
    """
    pass
```

References

- <https://peps.python.org/pep-0008/>
- <https://peps.python.org/pep-0257/>