

## Comments

- **Comments in Python** are the lines in the code that are ignored by the interpreter during the execution of the program.
- Comments enhance the readability of the code and help the programmers to understand the code very carefully.

### ❖ Types of Comments

There are three types of comments in Python:

- **Single line Comments**
- **Multiline Comments**
- **Docstring Comments**

### ❖ Single-Line Comments

- Single-line remarks in Python have shown to be effective for providing quick descriptions for parameters, function definitions, and expressions.
- A single-line comment of Python is the one that has a hashtag # at the beginning of it and continues until the finish of the line.
- If the comment continues to the next line, add a hashtag to the subsequent line and resume the conversation.

**Example:**

```
# This code is to show an example of a single-line comment  
print( 'This statement does not have a hashtag before it' )
```

**Output:**

```
This statement does not have a hashtag before it
```

## ❖ Multi-Line Comments

Python does not provide the facility for multi-line comments. However, there are indeed many ways to create multi-line comments.

### With Multiple Hashtags (#)

In Python, we may use hashtags (#) multiple times to construct multiple lines of comments. Every line with a (#) before it will be regarded as a single-line comment.

```
# it is a  
# comment  
# extending to multiple lines
```

### Using String Literals

Because Python overlooks string expressions that aren't allocated to a variable, we can utilize them as comments.

```
'it is a comment extending to multiple lines'
```

## ❖ Python Docstring

- The strings enclosed in triple quotes that come immediately after the defined function are called Python docstring.
- It's designed to link documentation developed for Python modules, methods, classes, and functions together.
- It's placed just beneath the function, module, or class to explain what they perform. The docstring is then readily accessible in Python using the `__doc__` attribute.

**Example:**

```
# Code to show how we use docstrings in Python

def add(x, y):
    """This function adds the values of x and y"""
    return x + y

# Displaying the docstring of the add function
print( add.__doc__ )
```

**Output:**

```
This function adds the values of x and y
```

**❖ Advantages of Using Comments**

- ◆ Readability of the Code
- ◆ Restrict code execution
- ◆ Provide an overview of the program or project metadata
- ◆ To add resources to the code