# **Statement, Indentation and Comment in Python**

#### **Statements**

Instructions written in the source code for execution are called statements. There are different types of statements in the Python programming language like Assignment statements, Conditional statements, Looping statements, etc. These all help the user to get the required output. For example, n = 50 is an assignment statement.  
**Multi-Line Statements:** Statements in Python can be extended to one or more lines using parentheses (), braces {}, square brackets [], semi-colon (;), continuation character slash (\). When the programmer needs to do long calculations and cannot fit his statements into one line, one can make use of these characters.   
**Example :**

Declared using Continuation Character (\):

s = 1 + 2 + 3 + \

4 + 5 + 6 + \

7 + 8 + 9

Declared using parentheses () :

n = (1 \* 2 \* 3 + 7 + 8 + 9)

Declared using square brackets [] :

footballer = ['MESSI',

'NEYMAR',

'SUAREZ']

Declared using braces {} :

x = {1 + 2 + 3 + 4 + 5 + 6 +

7 + 8 + 9}

Declared using semicolons(;) :

flag = 2; ropes = 3; pole = 4

#### **Indentation**

A block is a combination of all these statements. Block can be regarded as the grouping of statements for a specific purpose. Most of the programming languages like C, C++, Java use braces { } to define a block of code. One of the distinctive features of Python is its use of indentation to highlighting the blocks of code. Whitespace is used for indentation in Python. All statements with the same distance to the right belong to the same block of code. If a block has to be more deeply nested, it is simply indented further to the right. You can understand it better by looking at the following lines of code:

|  |
| --- |
| # Python program showing  # indentation    site ='gfg'    ifsite =='gfg':      print('Logging on to geeksforgeeks...')  else:      print('retype the URL.')  print('All set !') |

**Output:** 

Logging on to geeksforgeeks...

All set !

The lines print(‘Logging on to geeksforgeeks…’) and print(‘retype the URL.’) are two separate code blocks. The two blocks of code in our example if-statement are both indented four spaces. The final print(‘All set!’) is not indented, and so it does not belong to the else-block. 

|  |
| --- |
| j =1  while(j<=5):       print(j)       j =j +1 |

**Output:** 

1

2

3

4

5

To indicate a block of code in Python, you must indent each line of the block by the same whitespace. The two lines of code in the while loop are both indented four spaces. It is required for indicating what block of code a statement belongs to. For example, j=1 and while(j<=5): is not indented, and so it is not within the while block. So, Python code structures by indentation.

#### **Comments**

Python developers often make use of the comment system as, without the use of it, things can get real confusing, real fast. Comments are the useful information that the developers provide to make the reader understand the source code. It explains the logic or a part of it used in the code. Comments are usually helpful to someone maintaining or enhancing your code when you are no longer around to answer questions about it. These are often cited as a useful programming convention that does not take part in the output of the program but improves the readability of the whole program. There are two types of comments in Python:   
**Single line comments:** Python single line comment starts with hashtag symbol with no white spaces (#) and lasts till the end of the line. If the comment exceeds one line then put a hashtag on the next line and continue the comment. Python’s single-line comments are proved useful for supplying short explanations for variables, function declarations, and expressions. See the following code snippet demonstrating single line comment:  
**Code 1:** 

|  |
| --- |
| # This is a comment  # Print “GeeksforGeeks !” to console  print("GeeksforGeeks") |

**Code 2:** 

|  |
| --- |
| a, b =1, 3# Declaring two integers  sum=a +b # adding two integers  print(sum) # displaying the output |

**Multi-line string as acomment:** Python multi-line comment is a piece of text enclosed in a delimiter (“””) on each end of the comment. Again there should be no white space between delimiter (“””). They are useful when the comment text does not fit into one line; therefore need to span across lines. Multi-line comments or paragraphs serve as documentation for others reading your code. See the following code snippet demonstrating multi-line comment:  
**Code 1:** 

|  |
| --- |
| """  This would be a multiline comment in Python that  spans several lines and describes geeksforgeeks.  A Computer Science portal for geeks. It contains  well written, well thought  and well-explained computer science  and programming articles,  quizzes and more.  …  """  print("GeeksForGeeks") |

**Code 2:** 

|  |
| --- |
| '''This article on geeksforgeeks gives you a  perfect example of  multi-line comments'''    print("GeeksForGeeks") |