# **Python Continue Statement**

In this article, we will discuss **continue statements** in Python for altering the flow of loops.

## **Usage of Continue Statement**

[Loops](https://www.geeksforgeeks.org/loops-in-python/) in Python automates and repeats the tasks in an efficient manner. But sometimes, there may arise a condition where you want to exit the loop completely, skip an iteration or ignore that condition. These can be done by [loop control statements](https://www.geeksforgeeks.org/break-continue-and-pass-in-python/). Continue is a type of loop control statement that can alter the flow of the loop.

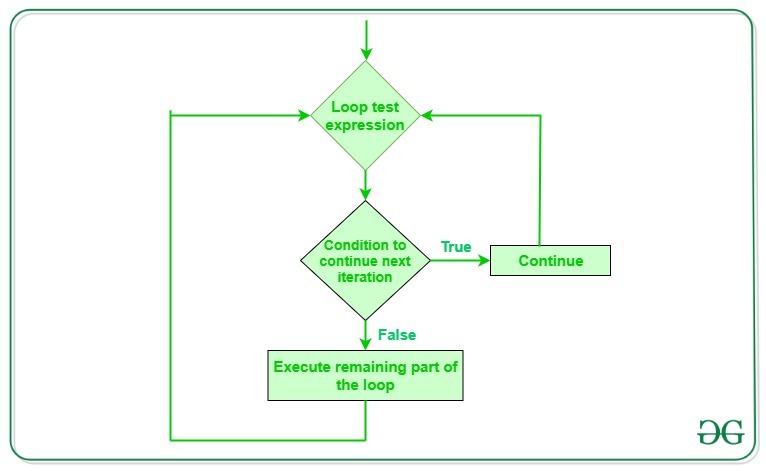
## **Continue statement**

**Continue statement** is a loop control statement that forces to execute the next iteration of the loop while skipping the rest of the code inside the loop for the current iteration only i.e. when the continue statement is executed in the loop, the code inside the loop following the continue statement will be skipped for the current iteration and the next iteration of the loop will begin.

**Syntax:**

continue

## **Flowchart of Continue Statement**



### **Example: Continue statement in Python**

Consider the situation when you need to write a program which prints the number from 1 to 10 and but not 6. It is specified that you have to do this using loop and only one loop is allowed to use. Here comes the usage of continue statement. What we can do here is we can run a loop from 1 to 10 and every time we have to compare the value of the iterator with 6. If it is equal to 6 we will use the continue statement to continue to the next iteration without printing anything otherwise we will print the value.

Below is the implementation of the above idea:

|  |
| --- |
| # Python program to  # demonstrate continue  # statement    # loop from 1 to 10  fori inrange(1, 11):        # If i is equals to 6,      # continue to next iteration      # without printing      ifi ==6:          continue      else:          # otherwise print the value          # of i          print(i, end=" ") |

**Output:**

1 2 3 4 5 7 8 9 10

**Note:** The continue statement can be used with any other loop also like [while loop](https://www.geeksforgeeks.org/python-while-loop/) in a similar way as it is used with [for loop](https://www.geeksforgeeks.org/python-for-loops/) above.

**Exercise Problem:**

Given a number n, print triangular pattern. We are allowed to use only one loop.

**Input:** 7

**Output:**

\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

\* \* \* \* \* \*

\* \* \* \* \* \* \*

Solution: [Print the pattern by using one loop | Set 2 (Using Continue Statement)](https://www.geeksforgeeks.org/print-pattern-using-one-loop-continue-statement/)