

Python While Loop with Break Statement

Python While Loop with Break Statement

[Python While Loop](#) executes a set of statements in a loop based on a condition. But, in addition to the standard breaking of loop when this while condition evaluates to false, you can also break the while loop using builtin [Python break](#) statement.

break statement breaks only the enclosing while loop.

In this tutorial, we will learn how to break a while loop using break statement, with the help of example programs.

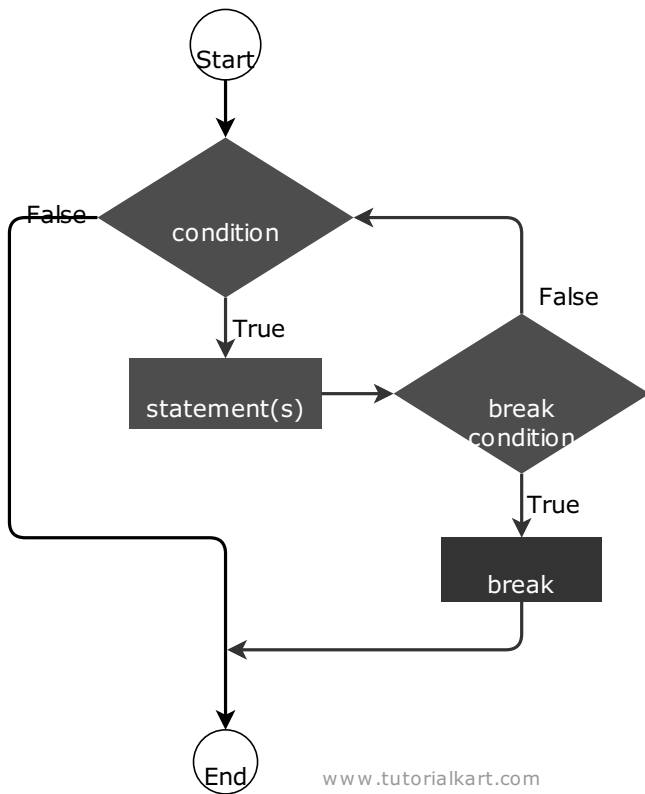
Syntax

Following is the syntax of while loop with a break statement in it.

```
#statement(s)
while condition :
    #statement(s)
    if break_condition :
        break
    #statement(s)
```

Usually break statement is written inside while loop to execute based on a condition. Otherwise the loop would break in the first iteration itself. Above syntax shows a [Python If](#) statement acting as conditional branching for break statement.

Following is the flow-diagram of while loop with break statement.



When the break condition is true, break statement executes and comes out of the loop.

Also, please note that the placement of break statement inside while loop is upto you. You can have statements before and after the break statement.

Example 1 – Break While Loop

In this example, we shall write a Python program with while loop to print numbers from 1 to 100. But, when we shall break the loop, after the number 7 is printed to the console.

Python Program

```

i = 1
while i <= 100 :
    print(i)
    if i == 7 :
        break
    i += 1
  
```

Output

```

1
2
3
4
5
6
7
  
```

Example 2 – Breaking Infinite While Loop

In this example, we shall write a Python program with an infinite while loop to print all natural numbers. But, when we shall break the loop, after some eight iterations.

Python Program

```
i = 1
while True :
    print(i)
    if i == 8 :
        break
    i += 1
```

Output

```
1
2
3
4
5
6
7
8
```

Please note that, if we do not write the break statement with the help of [Python IF](#) statement, the while loop is going to run forever until there is any interruption to the execution of the program.

Example 3 – Breaking Nested While Loop

In this example, we shall write a Python program with an nested while loop. We will break the inner while loop based on some condition.

Python Program

```
i = 1
while i < 6 :
    j = 1
    while j < 8 :
        print(i, end=" ")
        if j == 3 :
            break
        j += 1
    print()
    i += 1
```

Output

```
1 1 1
2 2 2
3 3 3
4 4 4
5 5 5
```

The nested while loop, without break statement would print numbers one to five, eight times each. But because of the break statement, each number is only printed thrice, instead of eight times.

Conclusion

In this [Python Tutorial](#), we learned how to break Python While Loop using break statement.

Python Programming

▸ [Python Tutorial](#)

▸ [Install Python](#)

▸ [Install Anaconda Python](#)

▸ [Python HelloWorld Program](#)

▸ [Python Variables](#)

▸ [Python Variable Data Type Conversion](#)

▸ [Python Comments](#)

Control Statements

▸ [Python If](#)

▸ [Python If Else](#)

▸ [Python While Loop](#)

▸ [Python For Loop](#)

Python String

▸ [Python String Methods](#)

▸ [Python String Length](#)

▸ [Python String Replace](#)

▸ [Python Split String](#)

▸ [Python Count Occurrences of Sub-String](#)

▸ [Python Sort List of Strings](#)

Functions

▸ [Python Functions](#)

Python Collections

▸ [Python List](#)

▸ [Python Dictionary](#)

Advanced

▸ [Python Multithreading](#)

Useful Resources

▸ [Python Interview Questions](#)