C++ break and continue Statement

In C++, there are two statements break; and continue; specifically to alter the normal flow of a program.

Sometimes, it is desirable to skip the execution of a loop for a certain test condition or terminate it immediately without checking the condition.

For example: You want to loop through data of all aged people except people aged 65. Or, you want to find the first person aged 20.

In scenarios like these, continue; or a break; statement is used.

## C++ break Statement

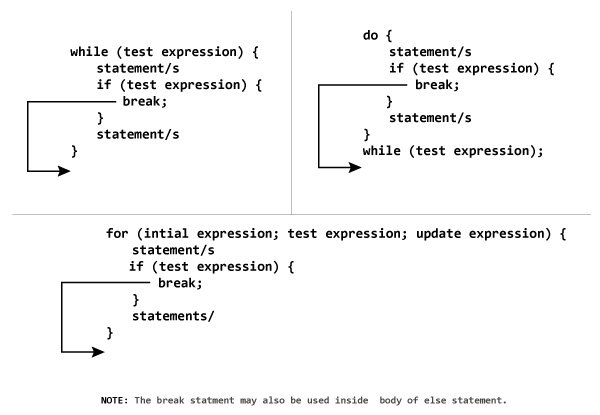
The break; statement terminates a loop (for, while and do..while loop) and a switch statement immediately when it appears.

### Syntax of break

break;

In real practice, break statement is usually used inside the body of conditional statement (if...else) inside the loop.

### How break statement works?



### Example 1: C++ break

**C++ program to add all number entered by user until user enters 0.**

// C++ Program to demonstrate working of break statement

#include <iostream>

using namespace std;

int main() {

float number, sum = 0.0;

// test expression is always true

while (true)

{

cout << "Enter a number: ";

cin >> number;

if (number != 0.0)

{

sum += number;

}

else

{

// terminates the loop if number equals 0.0

break;

}

}

cout << "Sum = " << sum;

return 0;

}

**Output**

Enter a number: 4

Enter a number: 3.4

Enter a number: 6.7

Enter a number: -4.5

Enter a number: 0

Sum = 9.6

In the above program, the test expression is always true.

The user is asked to enter a number which is stored in the variable number. If the user enters any number other than 0, the number is added to sum and stored to it.

Again, the user is asked to enter another number. When user enters 0, the test expression inside if statement is false and body of else is executed which terminates the loop.

Finally, the sum is displayed.

## C++ continue Statement

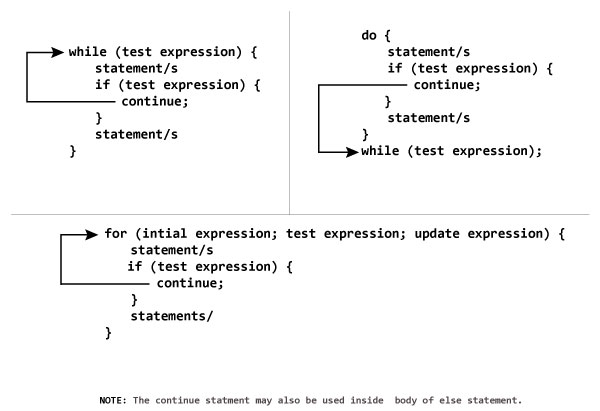
It is sometimes necessary to skip a certain test condition within a loop. In such case, continue; statement is used in C++ programming.

### Syntax of continue

continue;

In practice, continue; statement is usually used inside a conditional statement.

### Working of continue Statement



### Example 2: C++ continue

**C++ program to display integer from 1 to 10 except 6 and 9.**

#include <iostream>

using namespace std;

int main()

{

for (int i = 1; i <= 10; ++i)

{

if ( i == 6 || i == 9)

{

continue;

}

cout << i << "\t";

}

return 0;

}

**Output**

1 2 3 4 5 7 8 10

In above program, when i is 6 or 9, execution of statement cout << i << "\t"; is skipped inside the loop using continue; statement.