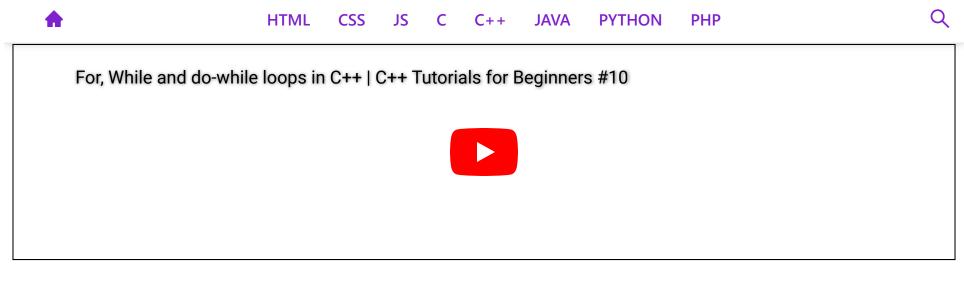
# **CodeWithHarry**



Overview Q&A Downloads Announcements

# For, While and do-while loops in C++ | C++ Tutorials for Beginners #10

For, While and Do-While Loops in C++

In this series of our C++ tutorials, we will visualize for loop, while loop, and do-while loop in C++ language in this lecture. In our last lesson, we discussed the control structures, If-else statements, and switch statements in C++.

### Loops in C++

Loops are block statements, which keeps on repeatedly executing until a specified condition is met. There are three types of loops in C++

• For loop in C++

- While loop in C++
- Do While in C++

# For Loop in C++

For loop help us to run some specific code repeatedly until the specified condition is met. An example program **for the loop** is shown in figure 1.

```
for (int i = 0; i < 4; i++)
{
    /* code */
    cout<<i<<endl;
}</pre>
```

# Figure 1: For Loop Program

As shown in figure 1, we created **for loop**, and inside its condition, there are three statements separated by a semicolon. The 1<sup>st</sup> statement is called "**initialization**", the 2<sup>nd</sup> statement is called "**condition**", and the 3<sup>rd</sup> statement is called "**updation**". After that, there is a loop body in which code is written, which needs to be repeated. Here is how our for loop will be executed:

- Initialize integer variable "i" with value "O"
- Check the condition if the value of the variable "i" is smaller than "4"
- If the condition is true go into loop body and execute the code
- Update the value of "i" by one
- Keep repeating this step until the condition gets false

The output for the following program is shown in figure 2.

1 2



Figure 2: For Loop Program Output

#### While Loop in C++

While loop helps us to run some specific code repeatedly until the specified condition is met. An example program of **while loop** is shown in figure 3.

```
int i=1;
while(i<=40){
    cout<<i<<endl;
    i++;
}</pre>
```

# Figure 3: While Loop Program

As shown in figure 3, we created **a while loop**, and inside its condition, there is one statement. The statement is called "**condition**". Here is how our while loop will be executed:

- Initialize integer variable "i" with value "1"
- Check the condition if the value of the variable "i" is smaller or equal to "40."
- If the condition is true to go into loop body and execute the code
- Update the value of "i" by one
- Keep repeating this step until the condition gets false.

# Do-While Loop in C++

The do-while loop helps us to run some specific code repeatedly until a specified condition is met. An example

https://www.codewithharry.com/videos/cpp-tutorials-in-hindi-10/

program of the **do-while loop** is shown in figure 1.

```
int i=1;
do{
    cout<<i<<endl;
    i++;
}while(i<=40);</pre>
```

Figure 4: Do-While Loop Program

As shown in figure 4, we created a **do-while loop**, and the syntax of the do-while loop is like write body with "**do**" keyword and at the end of body write "**while**" keyword with the condition. Here is how our do-while loop will be executed:

- Initialize integer variable "i" with value "1"
- Go into loop body and execute the code
- Check the condition if the value of the variable "i" is smaller or equal to "40"
- If the condition is true go into loop body and execute the code
- Keep repeating this step until the condition gets false

#### Code as described/written in the video

```
#include <iostream>
using namespace std;
int main()
```

```
/*Loops in C++:
There are three types of loops in C++:
    1. For loop
    2. While Loop
    do-While Loop
*/
/*For loop in C++*/
// int i=1;
// cout<<i;
// i++;
// Syntax for for loop
// for(initialization; condition; updation)
// {
       loop body(C++ code);
//
// }
// for (int i = 1; i <= 40; i++)
// {
//
     /* code */
//
       cout<<i<<endl;</pre>
// }
// Example of infinite for loop
```

```
// for (int i = 1; 34 <= 40; i++)
// {
// /* code */
// cout<<i<<endl;</pre>
// }
/*While loop in C++*/
// Syntax:
// while(condition)
// {
// C++ statements;
// }
   Printing 1 to 40 using while loop
// int i=1;
// while(i<=40){
//
       cout<<i<<endl;</pre>
//
       i++;
// }
// Example of infinite while loop
// int i = 1;
// while (true)
// {
       cout << i << endl;</pre>
//
//
       i++;
```

```
// }
/* do While loop in C++*/
    // Syntax:
    // do
    // {
           C++ statements;
    // }while(condition);
    Printing 1 to 40 using while loop
// int i=1;
// do{
       cout<<i<<endl;</pre>
       i++;
// }while(false);
return 0;
```

Previous

Cod

CodeWithHarry

Copyright © 2022 CodeWithHarry.com

Next

https://www.codewithharry.com/videos/cpp-tutorials-in-hindi-10/

