C# While Loop

To become expert programmer from novice, you need to completely cover the looping segments.

The loop is very important in every programming language that saves you from repetitive nature of work.

The **while loop** is one of the important looping constructs, that is being widely used in C sharp programming language.

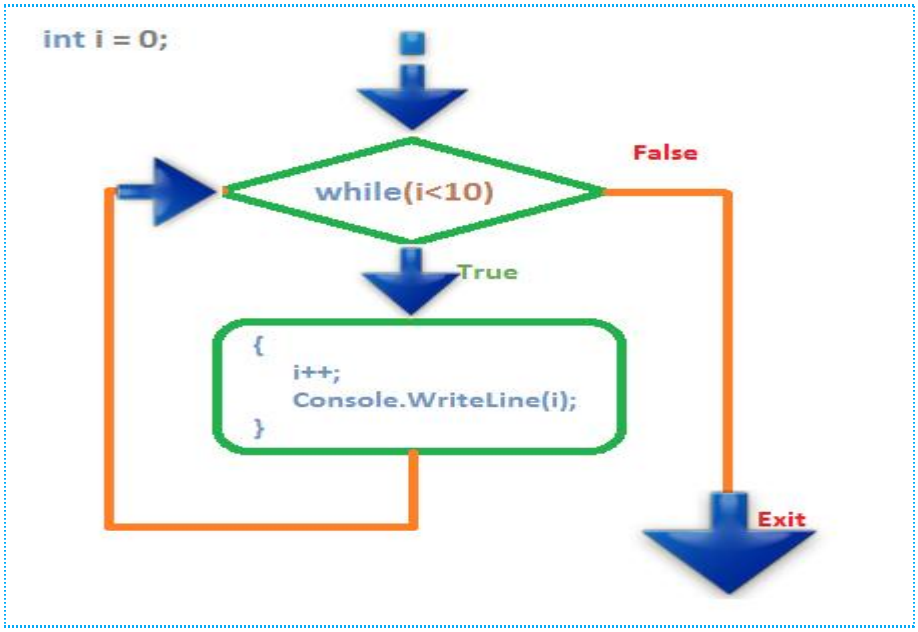
Below is the format of while statement loop in C#.

```
while (condition)
{
  loop body;
}
```

Condition is any expression that returns a **Boolean** result – **true or false**. It determines how long the loop body will be repeated and is called the loop condition.

The loop body is the code executed at each iteration of the loop whenever the input condition is **true**.

The behavior of while loops can be represented by the following scheme:



In the above example, three things are important.

- **Initialization**
- **Increment/Decrement**
- **Termination**

Initialization represents the starting position from there your loop will be started. In the above example **int i=0** refers to initialization of while loop.

i++ refers to **increment/decrement**

finally, **while(i<10)** refers to the **termination** of while loop.

- Each time the while loop executed, it checks for condition whether the value of **i** is less than 10 or not.
- If the value of **i** is less than 10, the loop will be executed and increment **i** by one and then print the value of **i**.
- When the value of **i** reaches 10, the **while(i<10)** condition becomes false and terminates the loop.

Sample Program

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace While_Loop
{
    class Program
    {
        static void Main(string[] args)
        {
            int num1, res, i;

            Console.WriteLine("Enter a number");
            num1 = Convert.ToInt32(Console.ReadLine());
```

```
.
.               i = 1; //Initialization
.               //Check whether condition matches or not
.               while (i <= 10)
.               {
.                   res = num1 * i;
.                   Console.WriteLine("{0} x {1} = {2}", num1, i, res);
.                   i++; //Increment by one
.               }
.               Console.ReadLine();
.           }
.       }
.   }
```

Output

```
Enter a number8
8 x 1 = 8
8 x 2 = 16
8 x 3 = 24
8 x 4 = 32
8 x 5 = 40
8 x 6 = 48
8 x 7 = 56
8 x 8 = 64
8 x 9 = 72
8 x 10 = 80
```

Sometime, you don't know the starting position of loop but want to execute some code in while loop. Or, you might want to run infinite loop that will be terminate with special character. In this situation, you can use **while(true)** as infinite loop.

Sample Program

```
.   using System;
.   using System.Collections.Generic;
```

```
.  using System.Linq;
.  using System.Text;
.
.  namespace while_true
.  {
.      class Program
.      {
.          static void Main(string[] args)
.          {
.              int num, sum = 0, res = 0;
.              Console.WriteLine("Enter number to add. Press x for Terminate the program");
.              while (true)
.              {
.                  num = Convert.ToInt32(Console.ReadLine());
.                  if (num == -1)
.                  {
.                      break;
.                  }
.                  sum = res;
.                  res += num;
.                  Console.WriteLine("\n{0} + {1} = {2}", sum, num, res);
.              }
.              Console.WriteLine("\n\nAborting... Press Enter.");
.              Console.ReadLine();
.          }
.      }
.  }
```

Output

```
Enter number to add. Press –1 for Terminate the program
60 + 6 = 6
56 + 5 = 11
4


11 + 4 = 15
3


15 + 3 = 18
2


18 + 2 = 20
1


20 + 1 = 21
–1


Aborting...Press Enter.
```