

Control Flow Statements in C#

In C#, control flow statements are used to manage the flow of execution in a program based on certain conditions. They help make decisions, loop through code, and handle various execution paths. Here's an overview of some of the most common control flow statements: if, if-else, if-else-if, break, continue, goto, and switch-case.

1. if Statement

The if statement executes a block of code only if a specified condition is true.

Syntax:

```
if (condition)
{
    // Code to execute if the condition is true
}
```

Example:

```
int number = 10;

if (number > 5)
{
    Console.WriteLine("Number is greater than 5.");
}
```

2. if-else Statement

The if-else statement executes one block of code if a condition is true, and another block of code if the condition is false.

Syntax:

```
if (condition)
{
    // Code to execute if the condition is true
}
else
{
    // Code to execute if the condition is false
}
```

```
}
```

Example:

```
int number = 3;

if (number > 5)
{
    Console.WriteLine("Number is greater than 5.");
}
else
{
    Console.WriteLine("Number is not greater than 5.");
}
```

3. if-else-if Statement

The if-else-if statement allows you to test multiple conditions sequentially. It checks the first condition, and if it is false, it checks the next condition, and so on.

Syntax:

```
if (condition1)
{
    // Code to execute if condition1 is true
}
else if (condition2)
{
    // Code to execute if condition2 is true
}
else
{
    // Code to execute if none of the above conditions are true
}
```

Example:

```
int number = 7;

if (number > 10)
{
    Console.WriteLine("Number is greater than 10.");
}
```

```
}  
else if (number == 7)  
{  
    Console.WriteLine("Number is 7.");  
}  
else  
{  
    Console.WriteLine("Number is less than 10 and not equal to 7.");  
}
```

4. break Statement

The break statement is used to exit from a loop or a switch-case statement prematurely.

Syntax (in loops or switch):

```
break;
```

Example in a Loop:

```
for (int i = 0; i < 10; i++)  
{  
    if (i == 5)  
    {  
        break; // Exits the loop when i is 5  
    }  
    Console.WriteLine(i);  
}
```

5. continue Statement

The continue statement skips the rest of the current iteration of a loop and proceeds to the next iteration.

Syntax (in loops):

```
continue;
```

Example:

```
for (int i = 0; i < 10; i++)  
{
```

```
if (i % 2 == 0)
{
    continue; // Skips the rest of the loop body when i is even
}
Console.WriteLine(i); // Only prints odd numbers
}
```

6. goto Statement

The goto statement transfers control to a labeled statement within the same method. It is generally discouraged because it can make code harder to understand and maintain.

Syntax:

```
goto label;
```

```
label:
```

```
// Code to execute after the goto statement
```

Example:

```
int number = 1;

start:
Console.WriteLine(number);
number++;

if (number <= 5)
{
    goto start; // Loops back to the start label
}
```

7. switch-case Statement

The switch-case statement allows you to execute one block of code out of many options based on the value of an expression. It is often used when dealing with multiple possible values of a single variable.

Syntax:

```
switch (expression)
{
```

```
case value1:
    // Code to execute if expression == value1
    break;
case value2:
    // Code to execute if expression == value2
    break;
default:
    // Code to execute if no case matches
    break;
}
```

Example:

```
int day = 3;
switch (day)
{
    case 1:
        Console.WriteLine("Monday");
        break;
    case 2:
        Console.WriteLine("Tuesday");
        break;
    case 3:
        Console.WriteLine("Wednesday");
        break;
    default:
        Console.WriteLine("Invalid day");
        break;
}
```

Summary

- **if Statement:** Executes code if a condition is true.
- **if-else Statement:** Executes one block of code if a condition is true and another if it is false.
- **if-else-if Statement:** Tests multiple conditions sequentially.
- **break Statement:** Exits loops or switch-case statements prematurely.
- **continue Statement:** Skips to the next iteration of a loop.
- **goto Statement:** Transfers control to a labeled statement (use sparingly).
- **switch-case Statement:** Selects a block of code to execute based on the value of an expression.

