

C# - Switch Statement

A **switch** statement allows a variable to be tested for equality against a list of values. Each value is called a case, and the variable being switched on is checked for each **switch case**.

Syntax

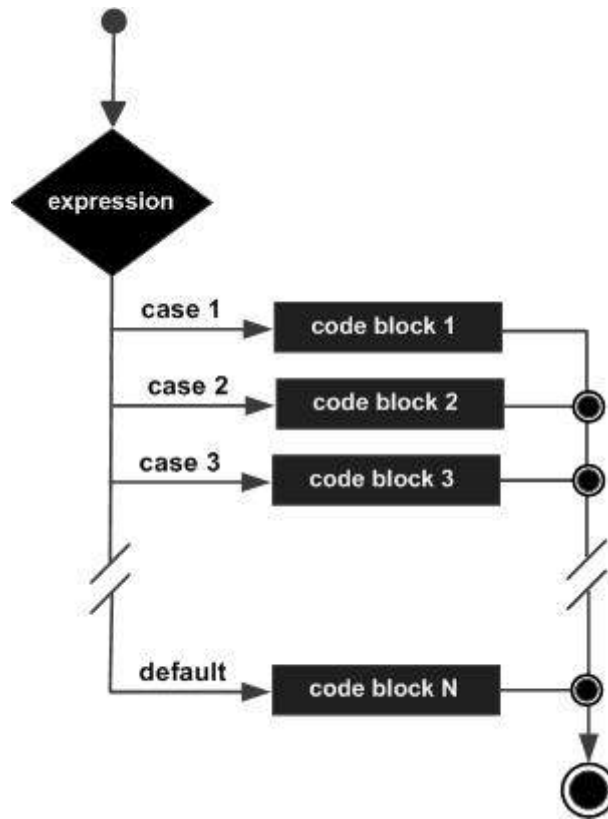
The syntax for a **switch** statement in C# is as follows –

```
switch(expression) {  
    case constant-expression1 :  
        statement(s);  
        break;  
    case constant-expression2 :  
    case constant-expression3 :  
        statement(s);  
        break;  
  
    /* you can have any number of case statements */  
    default : /* Optional */  
        statement(s);  
}
```

The following rules apply to a **switch** statement –

- The **expression** used in a **switch** statement must have an integral or enumerated type, or be of a class type in which the class has a single conversion function to an integral or enumerated type.
- You can have any number of case statements within a switch. Each case is followed by the value to be compared to and a colon.
- The **constant-expression** for a case must be the same data type as the variable in the switch, and it must be a constant or a literal.
- When the variable being switched on is equal to a case, the statements following that case will execute until a **break** statement is reached.
- When a **break** statement is reached, the switch terminates, and the flow of control jumps to the next line following the switch statement.
- Not every case needs to contain a **break**. If no **break** appears, then it will raise a compile time error.
- A **switch** statement can have an optional **default** case, which must appear at the end of the switch. The default case can be used for performing a task when none of the cases is true.

Flow Diagram



Example

```
using System;

namespace DecisionMaking {
    class Program {
        static void Main(string[] args) {
            /* local variable definition */
            char grade = 'B';

            switch (grade) {
                case 'A':
                    Console.WriteLine("Excellent!");
                    break;
                case 'B':
                case 'C':
                    Console.WriteLine("Well done");
                    break;
                case 'D':
                    Console.WriteLine("You passed");
                    break;
                case 'F':
                    Console.WriteLine("Better try again");
                    break;
                default:
                    Console.WriteLine("Invalid grade");
                    break;
            }
            Console.WriteLine("Your grade is {0}", grade);
            Console.ReadLine();
        }
    }
}
```

Live Demo

```
}  
  }  
}
```

When the above code is compiled and executed, it produces the following result –

```
Well done  
Your grade is B
```