

Introducción a C# .NET

Módulo 2

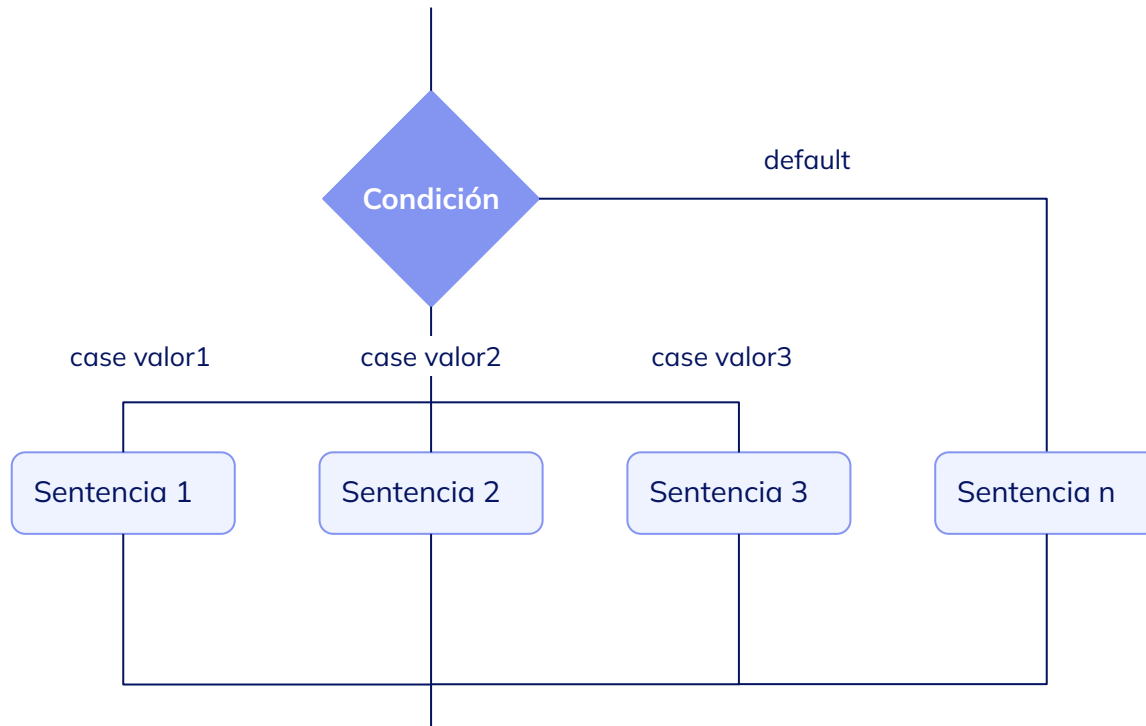
Instrucción switch

Instrucción switch

La instrucción **switch** permite ejecutar unos u otros bloques de instrucciones **según el valor de una expresión**. Su estructura es:

```
switch(<expresión>)  
{  
    case valor1: Sentencias1; <siguiente accion>;  
    case valor2: Sentencias2; <siguiente accion>;  
    case valor3: Sentencias3; <siguiente accion>;  
    default: SentenciasN;  
}
```





El significado de esta instrucción es el siguiente: se evalúa <expresión>. Si su valor es <valor1> se ejecuta el <bloque1>, si es <valor2> se ejecuta <bloque2>, y así para el resto de los posibles valores especificados. Si no es igual a ninguno de esos valores y se incluye la rama **default**, se ejecuta el <bloqueDefault>; pero si no se incluye se pasa directamente a ejecutar la instrucción siguiente al **switch**.

Los valores indicados en cada rama del **switch** han de ser expresiones constantes que produzcan valores de algún tipo básico entero, o de una enumeración, o de tipo **char** o de tipo **string**. Además, no puede haber más de una rama con el mismo valor.

En realidad, aunque todas las ramas de un **switch** son opcionales siempre se ha de incluir al menos una. Además, la rama **default** es opcional, pero es recomendable que se incluya para facilitar la legibilidad del código y también a la robustez del programa por si el valor de <expresión>, no se tuvo en cuenta al escribir el programa.



El elemento marcado como `<siguienteAcción>` colocado tras cada bloque de instrucciones indica qué es lo que ha de hacerse tras ejecutar las instrucciones del bloque que lo preceden.

Puede ser uno de estos tres tipos de instrucciones:

- **goto case** `<valor>;`
- **goto default;**
- **break;**

Si es un **goto case** indica que se ha de seguir ejecutando el bloque de instrucciones asociado en el **switch** a la rama del `<valor>` indicado.

Si es un **goto default** indica que se ha de seguir ejecutando el bloque de instrucciones de la rama **default**, si es un **break** indica que se ha de seguir ejecutando la instrucción siguiente al **switch**.

Nota:

Si bien los estamos explicando, **no es una buena práctica** de programación usar la instrucción de salto **goto**, dado que puede generar errores de lógica de ejecución difíciles de encontrar (desestructura el flujo normal de ejecución de un programa).

El siguiente ejemplo muestra cómo se utiliza **switch**:

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

namespace Switch
{
    class Estado Civil
    {
        static void Main(string[] args)
        {
            string Estado = "";
            Console.WriteLine("Ingrese su estado civil (Soltero / Casado / Divorciado): ");
            Estado = Console.ReadLine();
```



...

```
if (Estado.Length > 0)
{
    switch (Estado)
    {
        case "Soltero":
            Console.WriteLine("Usted es SOLTERO.");
            break;
        case "Casado":
            Console.WriteLine("Usted es CASADO.");
            break;
        case "Divorciado":
            Console.WriteLine("Usted es DIVORCIADO.");
            break;
    }
}
```

...

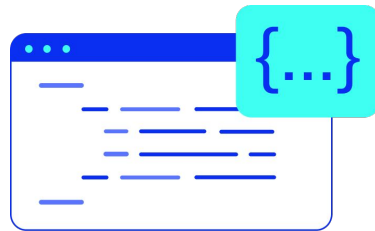
...

```
        default:
            Console.WriteLine("No es un estado civil permitido !");
            break;
    }
else
    {
        Console.WriteLine("No ingresó ninguna palabra !!!");
    }
Console.ReadKey();
}
```

El programa que vimos reconoce 3 posibles valores para el estado civil.

La rama **default** se incluye para mostrar que la cadena de caracteres ingresada no es un valor permitido. Como se observa, sólo se ejecuta el **switch** si la cadena de caracteres ingresada tiene al menos 1 carácter.

—



**¡Sigamos
trabajando!**