

Condicional *switch*

Estructura del **switch**

```
switch expresion {  
    case condicion-1:  
        // instrucciones si se cumple condicion_1  
    case condicion-n:  
        // instrucciones si se cumple condicion_n  
    default:  
        // instrucciones si no se cumple ninguna condición  
}
```

La estructura de control **switch** nos permite evaluar múltiples casos condicionales y ejecutar instrucciones sobre la base de estos. Este es una mejor alternativa que utilizar **if/else if/else** anidados.

Switch se compone de tres partes principales, analicemos en detalle cada una.

Estructura del **switch**

```
1 switch expresion {  
    case condicion-1:  
        // instrucciones si se cumple condicion_  
    case condicion-n:  
        // instrucciones si se cumple condicion_n  
    default:  
        // instrucciones si no se cumple ninguna condición  
}
```

1

La primera parte se compone de la palabra reservada **switch** y una **expresión** la cual vamos a evaluar. Las instrucciones que se ejecutarán, se basarán en el valor de estas expresiones dentro de los casos a evaluar.

Estructura del **switch**

2

```
switch expresion {  
    case condicion-1:  
        // instrucciones si se cumple condicion_1  
    case condicion-n:  
        // instrucciones si se cumple condicion_n  
    default:  
        // instrucciones si no se cumple ninguna condición  
}
```



2

Luego tenemos las **condiciones**. Estas serán comparadas con el valor de las expresiones y, si encuentra una coincidencia, se ejecuta el bloque de instrucciones que contiene.

Estructura del **switch**

```
switch expresion {  
    case condicion-1:  
        // instrucciones si se cumple condicion_  
    case condicion-n:  
        // instrucciones si se cumple condicion_n  
3  default:  
        // instrucciones si no se cumple ninguna condición  
}
```



3

Finalmente, tenemos la palabra reservada **default** que nos permite definir un bloque de instrucciones en caso de que no se cumpla ninguno de los casos anteriores.



Este debe ser el último bloque de los posibles casos.

Switch sin condición

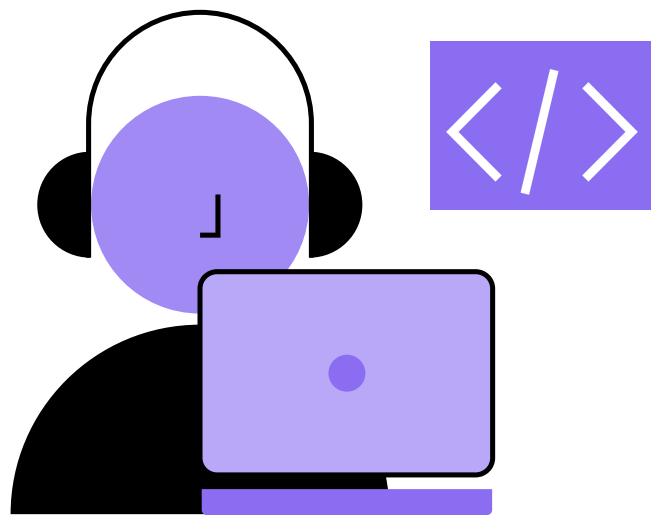
Podemos utilizar **switch** sin condición, agregando directamente la condición en el **case**:



```
var expresion
switch {
case expresion == condicion_1:
    // instrucciones si se cumple condicion_1
case expresion == condicion_2:
    // instrucciones si se cumple condicion_2
default:
    // instrucciones sin caso
}
```

Switch con múltiples casos

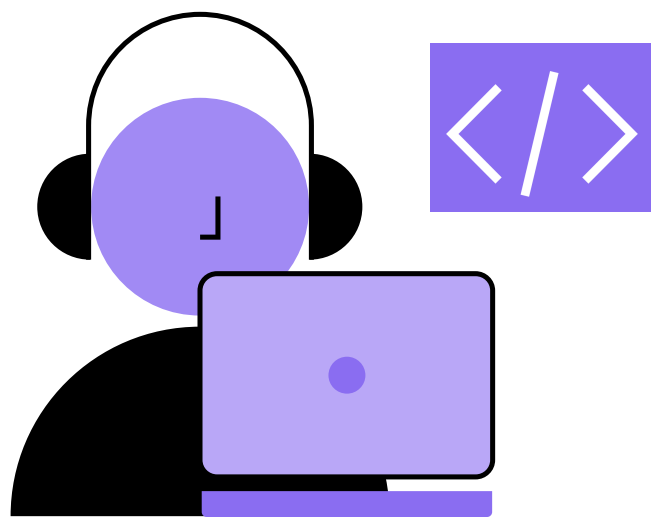
Los casos pueden tener múltiples valores separados por comas que indican los valores que cumple la condición:



```
switch expression {  
    case condicion_1, condicion_2, condicion_n:  
        // instrucciones si la condicion_1,  
        condicion_2,  
        // ..., condicion_n  
    default:  
        // instrucciones si no se cumple ninguna  
        condición  
}
```

Switch con declaración corta

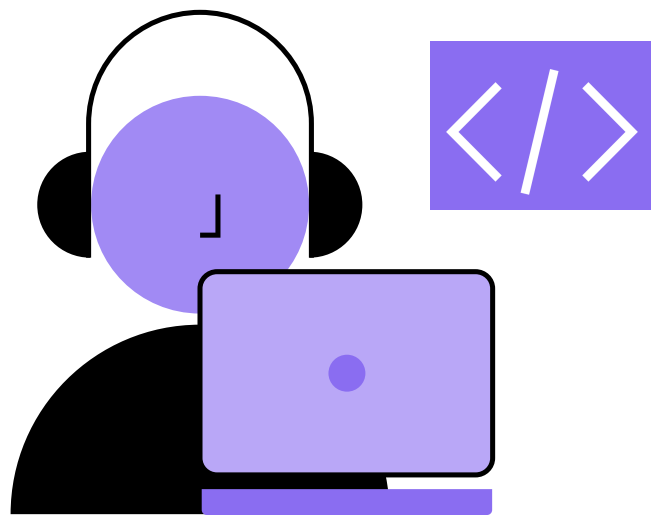
Tal como es posible hacerlo con **if**, **switch** también nos permite declarar una variable para usarla dentro de la sentencia, y luego evaluarla.



```
switch var expresion; expresion{  
    case condicion_1:  
        // instrucciones si cumple condicion_1  
    case condicion_2:  
        // instrucciones si cumple condicion_2  
    default:  
        // instrucciones sin caso  
}
```


Switch con fallthrough

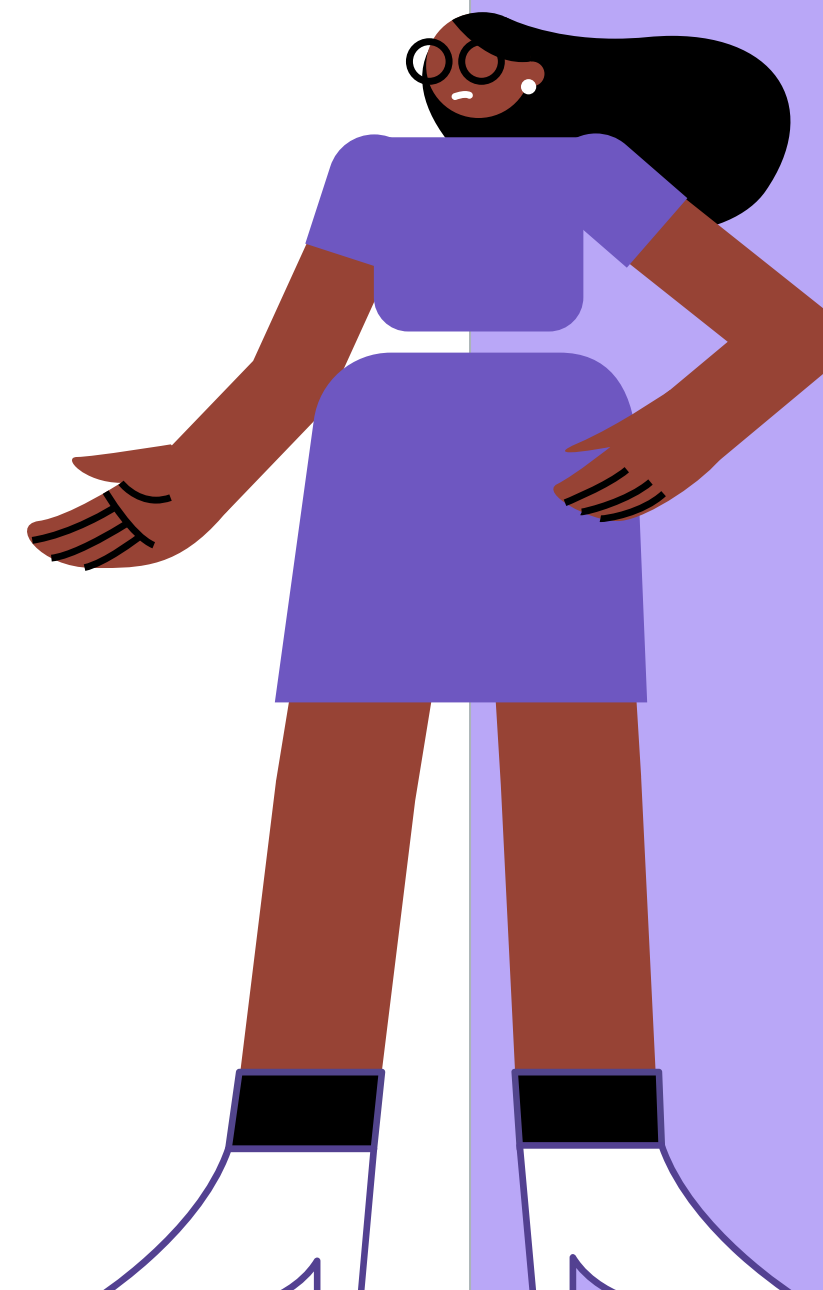
Dentro de los casos también podemos utilizar la palabra reservada **fallthrough**, que indica que se ejecute las instrucciones del caso siguiente:



```
switch {
  case condicion_1:
    // instrucciones si cumple condicion_1
  case condicion_2:
    // instrucciones si cumple condicion_2
    fallthrough
  case condicion_3:
    // instrucciones si cumple condicion_3 o condicion_2
  default:
    // instrucciones sin caso
}
```

Conclusiones

La estructura de control switch nos permite tomar distintos caminos en la ejecución de nuestro programa, al igual que **if/else**. No obstante, cuando las condiciones a evaluar son muchas, lo ideal es utilizar **switch**.



¡Muchas gracias!