

SWITCH

ESTRUCTURAS DE CONTROL

SWITCH

- Nos permite **evaluar varias posibilidades** sin tener que pasar por todas las anteriores.
- Desventaja: las ***condiciones son siempre de igualdad***.
- Funciona con los **tipos primitivos** y **String**, entre otros.
- Dos versiones
 - switch como sentencia.
 - switch como expresión.

SWITCH COMO SENTENCIA

- Es la forma tradicional.
- Para cada caso podemos ejecutar una o varias sentencias.
- Al terminar un caso añadimos *break*.
- Si no lo hacemos, continúa ejecutando el siguiente caso.
- *default* por si no coincide con ningún caso (similar a *else*)

Sintaxis

```
switch(expresion) {  
    case v1: accion1;  
            accion2;  
            break;  
    case v2: accion3;  
            break;  
    // más casos  
    default: accionN;  
}
```

SWITCH COMO EXPRESIÓN

- Desde Java 14
- Variante del anterior.
- En lugar de ejecutar una serie de acciones por cada caso, se realiza una asignación.
- Se pueden agrupar varios casos:
case v1, v2, ... → resultado

Sintaxis

```
variable = switch(expresión) {  
    case valor1 -> resultado1;  
    case valor2 -> resultado2;  
    // ...  
    case valorn -> resultadoN;  
    default -> resultadoPorDefecto;  
};
```