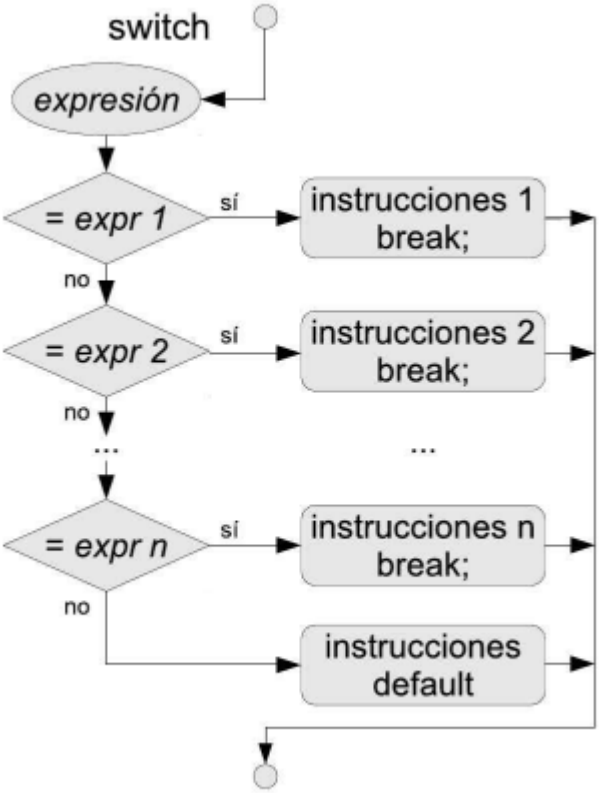
Sentencia *SWITCH*

Otra estructura de selección o alternativa para agregar a nuestro repertorio es la estructura ***switch***.

Syntax

```
switch(expression) {
  case x:
    // code block
    break;
  case y:
    // code block
    break;
  default:
    // code block
}
```



La expresión **switch** se evalúa una vez. El valor de la expresión se compara con los valores de cada caso. Si hay una coincidencia, se ejecuta el bloque de código asociado. Las palabras clave **break** y **default** **son opcionales**. Los casos (case) suelen ser literales que no varían, y a menudo se usan constantes o el tipo de dato enum. No se aceptan el tipo punto flotante(34.5)

Es una forma abreviada de codificar la estructura de selección múltiple que vimos en el capítulo anterior.

De forma que el siguiente código expresado con if-else múltiple:

```
    if (numJugadores == 1) {
        System.out.println("Single player");
    } else if (numJugadores == 2) {
        System.out.println("Two player");
    } else if (numJugadores == 3) {
        System.out.println("Multiplayer");
    } else {
        System.out.println("Not possible, too many players");
    }
```

Es equivalente a:

```
    switch (numJugadores) {
        case 1:
            System.out.println("Single player");
            break;
```

```
        case 2:
            System.out.println("Two player");
            break;
        case 3:
            System.out.println("Multiplayer");
            break;
        default:
            System.out.println("Not possible, too many players");
    }
```

BREAK

Cuando el programa Java alcanza la palabra break, se sale del bloque switch. Es decir, ya no se sigue ejecutando más código dentro del switch ni ningún case.

Un break puede ahorrar mucho tiempo de ejecución porque "ignora" la ejecución de todo el resto del código switch.

DEFAULT

La palabra default se usa para ejecutar código cuando no hay ninguna coincidencia con ningún caso (*case*). Siempre se pone al final, por tanto no necesita de la instrucción *break*.

Agrupar casos

En el **Switch** se va comparando cada *case* y una vez que encuentra una coincidencia ejecuta todo el código de **switch** hasta que se encuntra un **break** o termina.

```
int numJugadores=2;
switch (numJugadores) {
    case 1:
        System.out.println("Single player");
    case 2:
        System.out.println("Two player");
    case 3:
        System.out.println("Multiplayer");
    default:
        System.out.println("Not possible, too many players");
}
```

Se ejecuta el código marcado

```
Two player
Multiplayer
Not possible, too many players
```

Esta característica la podemos aprovechar para agrupar casos

```
String diaSemana="Miércoles";
switch (diaSemana) {
    case "Lunes":
    case "Martes":
    case "Miércoles":
    case "Jueves":
    case "Viernes":
        System.out.println("Es un día laborable.");
        break;
    case "Sábado":
    case "Domingo":
        System.out.println("Es un fin de semana.");
        break;
    default:
        System.out.println("Día no válido.");
        break;
}
```