

ESTRUCTURAS DE DECISIÓN

OPERADORES JAVA RELACIONALES

< Menor que
> Mayor que
<= Menor o igual que
>= Mayor o igual que
!= Distinto
== Igual

OPERADORES JAVA LÓGICOS

&& AND
|| OR
! NOT

IF

Con IF podemos realizar muchísimas comprobaciones utilizando las estructuras booleanas True y False.

```
if(condicion){  
    accion1;  
}  
//Esta condición solo es obligatoria cuando se necesitan comprobar varias condiciones  
else if(condicion2){  
    accion2;  
}  
else{  
    accion3;  
}
```

Ejemplo de programa que comprueba si es mayor o menor de edad

```
public static void main(String[] args) {  
    //creamos la variable con el valor 17  
    int valor1 = 17;  
  
    //si valor1 es mayor o igual que 18  
    if (valor1 >= 18) {  
        //entonces muestra este mensaje  
        System.out.println("Es mayor de edad");  
    }  
    else{  
        //si no lo es muestra este  
        System.out.println("Es menor de edad");  
    }  
}
```

Ejemplo de programa que comprueba si un número es par o impar

```

public static void main(String[] args) {
    //creamos la variable con el valor 4
    int paroi impar = 4;

    /*en esta variable almacenamos la operación del resto
    llamando a la variable anterior*/
    int resto = paroi impar % 2;

    //si el resto es 0 el número será par
    if (resto == 0) {
        System.out.println("El número " + paroi impar + " es par");
    }
    else{
        System.out.println("El número " + paroi impar + " es impar");
    }
}

```

SWITCH

Con SWITCH podemos crear menús para distintas operaciones que podemos vincular con otras estructuras como **IF**, **WHILE** e incluso el mismo **SWITCH**.

//switch recojerá una variable que debe tener el valor de los “case”

```

switch(valor1){
    //cada “case” debe terminar con un “break” sí o sí
    case 1:
        operacion1
        break;
    case 2:
        operacion2
        break;
    case 3:
        operacion3
        break;
    case 4:
        operacion4
        break;
    case 5:
        operacion5
        break;
    /*en el caso de que el valor introducido no corresponda a ningún “case” el switch entrará
    directamente en default*/
    default:
        operacion6
}

```

Ejemplo de un menú realizado con switch.

```
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);

    System.out.println("Introduce el valor de la sección del menú");
    int valor = sc.nextInt();

    switch (valor) {
        case 1:
            System.out.println("Has entrado en la sección 1");
            break;
        case 2:
            System.out.println("Has entrado en la sección 2");
            break;
        default:
            System.out.println("No has seleccionado ninguna "
                               + "de las opciones anteriores");
    }
}
```

BUCLES

WHILE

Con While podemos crear bucles infinitos siempre y cuando no se cumpla la condición de salida.

```
while( condición ){
    operación
    condicion++;
}
```

```
public static void main(String[] args) {
    //creamos la variable i con el valor 1
    int i = 1;
    //mientras i tenga un valor menor a 11 mostramos el siguiente mensaje
    while( i < 11 ){
        System.out.println("El valor de i es " + i);
        //una vez terminada la operación ampliamos el valor de i
        i++;
        /*ahora i tendrá el valor 2 y cada vez que llegue a este
        punto sumará 1 a su valor*/
    }
}
```

DO WHILE

El bucle “do while” utiliza la misma estructura que el “while” pero primero se realiza una acción que puede no llegar a entrar en el bucle, cosa que con el “while” normal si que pasa.

```
do{  
    operación  
}  
while ( condición );
```

```
public static void main(String[] args) {  
    Scanner sc = new Scanner(System.in);  
    //creamos las variables fuera del bucle  
    int i;  
    int x;  
    int suma;  
  
    do {  
        //pedimos los 2 números y realizamos la suma  
        System.out.println("Introduce el primer número");  
        i=sc.nextInt();  
        System.out.println("Introduce el segundo número");  
        x=sc.nextInt();  
        suma = i + x;  
        /*si la suma de esos números es menor a 20 entra en el bucle  
        y repite el proceso*/  
    } while ( suma < 20 );  
    //si la suma es mayor a 20 sales del bucle  
    System.out.println("Has pasado el bucle");  
}
```

FOR

El bucle for es de los más útiles debido a su simplicidad, se divide en 3 secciones:

```
for (variable; condición; incremento){  
    operación  
}
```

```
public static void main(String[] args) {  
    //no es necesario crear una variable antes del for  
    /*la primera sección hace referencia a la variable  
    que se va a utilizar para la salida del bucle, la  
    segunda sección crea la condición para la salida y  
    la tercera incrementa la variable hasta que se cumpla  
    la condición*/  
    for (int i = 0; i < 10; i++) {  
        //mostramos el valor de i hasta que sea mayor que 10  
        System.out.println("El valor de i es "+i);  
    }  
}
```

Ejemplo de unir las distintas estructuras.

```
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);

    int salida = 0;

    do{
        System.out.println("Introduce el valor de una de las secciones");
        System.out.println("1 - SUMA 2 NÚMEROS\n"
            + "2 - NÚMERO DE LETRAS DE UNA CADENA\n"
            + "3 - CARÁCTERES DE UNA CADENA\n"
            + "4 - SALIDA");
        int valor = sc.nextInt();
        String cadena;
        switch(valor){
            case 1:
                sc.nextLine();
                System.out.println("Introduce los 2 números a sumar");
                int i=sc.nextInt();
                int x=sc.nextInt();
                int suma=i+x;
                System.out.println("El resultado es "+suma);
                salida=0;
                break;
            case 2:
                sc.nextLine();
                System.out.println("Introduce la cadena");
                cadena=sc.nextLine();
                System.out.println("La cadena tiene "+cadena.length()+" caracteres");
                salida=0;
                break;
            case 3:
                sc.nextLine();
                System.out.println("Introduce la cadena");
                cadena=sc.nextLine();
                for (int j = 0; j < cadena.length(); j++) {
                    System.out.println("El carácter nº "+j+" es "+cadena.charAt(j));
                }
                salida=0;
                break;
            case 4:
                salida=1;
                System.out.println("Hasta luego");
                break;
            default:
                System.out.println("No has introducido un valor correcto");
        }
    }while( salida != 1);
}
```