

Instrucción IF simple

- Estructura: Si la condición es cierta (true), se ejecutan las instrucciones que están dentro de las llaves.

```
if(condicion){  
    instrucción 1 a ejecutar;  
    instrucción 2 a ejecutar;  
    .....  
}
```

- En el caso que haya sólo una instrucción a ejecutar, las llaves son opcionales.

Instrucción IF compuesta

- Estructura: Se añade '**else**' que se ejecuta si la condición del if es falsa.

```
if(condicion){  
    instrucción 1 a ejecutar;  
    instrucción 2 a ejecutar;  
    .....  
}  
else{  
    instrucción 3 ejecutar;  
    instrucción 4 ejecutar;  
    ....  
}
```

- Si queremos evaluar varias condiciones podemos utilizar **else if**

```
if(x==1){  
    instrucciones ....  
}  
else if(x==2){  
    instrucciones....  
}  
else if(x==3){  
    instrucciones ....  
}  
else // cuando x no es ni 1 ni 2 ni 3  
{  
    ....  
}
```

Operador Condicional

Equivale a la estructura if/else de la siguiente forma:

```
condicion ? valor_si_true : valor_si_false
```

✓ Ejemplo:

```
let tipo = (edad >= 18) ? "mayor" : "impar"
```

Switch Case

- Evalúa una expresión o condición y según el valor, se ejecuta un bloque:

```
switch (expresion){  
    case valor1:  
        instrucciones ...  
        break;  
    case valor2:  
        instrucciones ...  
        break;  
    ....  
    default:  
        instrucciones...  
}
```

- La etiqueta **default** es opcional. Se ejecuta si el valor de la expresión no coincide con ninguno de los anteriores casos evaluados.
- **Break.** Provoca que termine la evaluación de casos. Si no se incluye break, sigue ejecutando los casos posteriores hasta que se encuentre un break que le obligue a salir.

Bucles:

En muchas situaciones es necesario repetir la misma operación sobre datos y para ello nos pueden ayudar los bucles.

Bucle con while

- Se evalúa la condición
- Si la expresión es verdadera se ejecuta las sentencias, si no, el programa sale de la sentencia while
- Tras ejecutar las sentencias, volvemos al paso1.
- Si queremos abandonar un bucle antes de que termine de cumplirse la condición usamos **break**;

✓ Ejemplo:

```
let i=1;
while(i<=100){
    console.log(i);
    i++;
}
```

Bucle con do / while

- El bucle se ejecuta al menos una vez.
- Ejecuta las instrucciones del do
- Evalúa la expresión lógica.
- Si es true, volvemos al paso1, si no, sale del bucle y continua el programa.

✓ Ejemplo:

```
let i=0;
do {
    i++;
    console.log(i);
} while(i < 1000);
```

Este bucle escribe por consola los números
Del 1 al 1000

Bucle for

- Utiliza índices

✓ Ejemplo:

```
for( i = 1; i<= 1000; i++){
    console.log(i);
}
```

Bucle for anidados

Dentro de un bucle for podemos tener anidado uno o más bucles for:

✓ Ejemplo:

```
for( i = 1; i<= 6; i++){
    document.write("Ficha: ");

    for (j = 1; j <= i; j++) {
        document.write("|" + i + ":" + j + "| ");
    }
    document.write("<BR>");
}
```

Ejercicios

1. Crear una página que pida una nota a través de un cuadro de entrada de texto (prompt) :
 1. Escribir en pantalla si es mayor o igual que 5, es decir, si ha aprobado.
 2. Escribir en pantalla si ha aprobado o ha suspendido.
2. Crear una página que pida **un número** entre 0 y 23:
 1. Si no está entre esos valores, volver a preguntar el número.
 2. Si es correcto:
 1. Escribir por pantalla “Buenos días” si el valor está entre 6 y 11
 2. Escribir por pantalla “Buenas tardes” si el valor está entre 12 y 20
 3. Escribir por pantalla “Buenas noches” si el valor está entre 21 y 6
3. Escribe un programa que pida por mensaje prompt un tipo de topping para comprar un helado. Los valores serán: oreo/kitkat/brownie/lacasitos
Se tendrá que mostrar por pantalla el precio del helado de forma que:
 1. El helado sin topping cuesta 1.90€.
 2. El topping de oreo cuesta 1€.
 3. El topping de kitkat cuesta 1.50€.
 4. El topping de brownie cuesta 0.75€.
 5. El topping de lacasitos cuesta 0.95€.
 6. En caso de no disponer del topping solicitado por el usuario el programa escribirá por pantalla “No tenemos este topping, lo sentimos” y a continuación informar del precio del helado sin ningun topping.
 7. Si tiene el topping, el programa escribe por consola el precio del helado con el topping seleccionado.
4. Escribe un programa que solicite por pantalla un tipo de motor: los valores permitidos son 1,2,3 y 4
 1. Si el tipo de motor es 0, mostrar un mensaje indicando “No hay establecido un valor definido para el tipo de bomba”.
 2. Si el tipo de motor es 1, mostrar un mensaje indicando “La bomba es una bomba de agua”.
 3. Si el tipo de motor es 2, mostrar un mensaje indicando “La bomba es una bomba de gasolina”.
 4. Si el tipo de motor es 3, mostrar un mensaje indicando “La bomba es una bomba de hormigón”.

5. Si el tipo de motor es 4, mostrar un mensaje indicando “La bomba es una bomba de pasta alimenticia”.
 6. Si no se cumple ninguno de los valores anteriores mostrar el mensaje “No existe un valor valido para tipo de bomba”.
-
5. Escribir un programa que muestre por pantalla los número pares entre 0 y 100
 6. Escribir un programa que muestre por pantalla los número del 100 al 0
 7. Escribir un programa que muestre por pantalla los 50 número impares a partir de un número dado por el usuario.
 8. Crear una página que pida al usuario un número a través de un cuadro de lectura. Mostrar con una ventana alert si el valor introducido es un número o no.
 9. Crear una aplicación web que pida al usuario un número entero positivo. Si no es un número o no es positivo, volveremos a pedir al usuario el número. La aplicación escribirá un triángulo con tantos asteriscos como indique el usuario. Entre cada asterisco utilizar un espacio.

Ej: para el número 3

```
*  
* *  
* * *
```

10. Sacar por pantalla utilizando bucles:

```
54321  
4321  
321  
21  
1
```

11. Crear una aplicación web en la que se pida el nombre, los apellidos, el salario (número con decimales) y la edad de una persona(un número entero). Suponemos que los datos están correctamente.

La página indicará el nombre y apellidos, edad y el salario según lo siguiente:

- si el salario es mayor de 2000 euros, no cambiará
- si el salario está entre 1000 y 2000:
 1. si además la edad es mayor de 45 años, se sube un 3%.
 2. si además la edad es menor de 45 o igual, se sube un 10%
- si el salario es menor de 1000
 1. los menores de 30 años cobrarán exactamente 1100 euros
 2. si la edad es de 30 a 45 años, el salario sube un 3%
 3. A los mayores de 45 sube un 15%

12. Dada una cadena mostrarla al revés por pantalla

Ej: "Hoy es viernes" mostrar: "senreiv se yoH"

13. Calcular y mostrar la media de 3 números leídos por el teclado. Si el promedio es mayor o igual a 7 mostrar por pantalla 'Promocionado'
14. Realizar la carga del lado de un cuadrado, mostrar por pantalla el perímetro del mismo (El perímetro de un cuadrado se calcula multiplicando el valor del lado por cuatro)
15. Pedir la base y la altura y mostrar una figura con asteriscos en los lados de un rectángulo.

Eje: base 6 y altura 3

```
* * * * *
*           *
* * * * *
```

16. Realizar esta figura rombo para un número de asteriscos que se pedirá por pantalla.
El ejemplo es para 5 asteriscos.

```

                *
            *   *
        *   *   *
    *   *   *   *
*   *   *   *   *
```

17. Solicitar un número del 1 al 10 y mostrar la tabla de multiplicar de dicho número
18. Pedir dos números al usuario y mostrar todos los números que se encuentran entre ellos.
19. Se ingresan por teclado tres números, si al menos uno de los valores ingresados es menor a 10, imprimir en la página la leyenda 'Alguno de los números es menor a diez'
20. De un operario se conoce su sueldo y los años de antigüedad. Se pide confeccionar un programa que lea los datos de entrada e informe:
- Si el sueldo es inferior a 500 y su antigüedad es igual o superior a 10 años, otorgarle un aumento del 20 %, mostrar el sueldo a pagar.
 - Si el sueldo es inferior a 500 pero su antigüedad es menor a 10 años, otorgarle un aumento de 5 %.
 - Si el sueldo es mayor o igual a 500 mostrar el sueldo en la página sin cambios.