

Ejercicios básicos JavaScript 2

- 1) Crear una variable llamada **numero1** con el valor 4 y otra llamada **numero2** con el valor 5. Mostrar en la consola la suma de ambas variables:

```
console.log(uno_o_más_valores_separados_por_comas)
```

Las variables las podemos crear de varios modos:

```
let nombre = valor; // variables cuyo valor podemos modificar posteriormente
```

```
const nombre = valor; // variables cuyo valor no podemos modificar posteriormente
```

```
var nombre = valor; // variables cuyo valor podemos modificar posteriormente pero mejor no  
// usarlas hoy día
```

```
let resultado = 32 * 5 / 4; // Podemos asignarles cualquier expresión por compleja que sea
```

```
let nombre; // Podemos crearlas sin valor inicial (no las const)
```

- 2) Al programa anterior añadir una tercera variable llamada **suma** a la que asignaremos la suma de ambas variables. Mostrar en la consola la variable suma
- 3) Al programa anterior añadir una cuarta variable **producto** con la multiplicación de numero1 y numero2. Mostrar en la consola la variable producto
- 4) Crear una variable constante llamada **precio** con el valor 20.4 y otra llamada **cantidad** con el valor 5. Crear una variable constante llamada **total** con el producto de ambas. Mostrar en consola el texto *mayor que 100* si total es mayor que 100.

```
if(condición) {  
    instrucciones si se cumple la condición  
}
```

En la condición podemos usar == (igual valor), === (igual valor y tipo), != (distinto valor), !== (distinto valor o tipo), < (menor), <= (menor o igual), > (mayor), >= (mayor o igual).

Ejemplo:

"5" == 5	devuelve true
"5" === 5	devuelve false
" " == 0	devuelve true pues convierte " " a 0 para comparar
" " === 0	devuelve false

- 5) Modificar el programa anterior para que muestre *menor que 100* si total es menor que 100

```
if(condición) {  
    instrucciones si se cumple la condición  
} else {  
    instrucciones si no se cumple la condición  
}
```

- 6) Crear una variable constante **temperatura** a la que asignemos un valor entre -40 y 40 con la siguiente expresión: `Math.floor(Math.random() * 81 - 40)`. Si la temperatura es menor que cero escribir en la consola *bajo cero*, si es cero escribir *cero* y si es mayor que cero escribir *sobre cero*.

```
if(condición) {  
    instrucciones si se cumple la condición  
} else if(condición){  
    instrucciones si se cumple la condición
```

Ejercicios básicos JavaScript 2

```
} else {  
    instrucciones si no se cumple ninguna condición  
}
```

La fórmula para generar un número "aleatorio" entre un mínimo y un máximo, ambos incluidos, es:

```
Math.floor(Math.random() * (máximo - mínimo + 1) + mínimo);
```

- 7) Como el anterior, pero en lugar de mostrar directamente el resultado, guardarlo en una variable llamada **texto**, luego, después de todos los if, mostrarla en la consola

```
let texto; // Debemos crear la variable fuera de las llaves de sus if, ya que si no solo existiría dentro de esas llaves. En este caso la variable no tiene tipo al comienzo. Otra opción, ya que va a contener textos, es crearla con una cadena vacía: let texto = "";
```

- 8) Crear una variable llamada **día** con este valor: `new Date().getDay()`. Esto devuelve el día de la semana en número, siendo 0 el domingo, 1 el lunes, 6 e sábado. Mostrar por consola el día en letras. Para ello usar unos cuantos if anidados (6 if y un else final) o un switch. Para probar que funcione bien podemos ir probando con diferentes números en *día* en lugar de `new Date().getDay()`

```
switch(variable_o_expresión) {  
    case valor1:  
        instrucciones si la variable o expresión es igual al valor1  
        break;  
    case valor2:  
        instrucciones si la variable o expresión es igual al valor2  
        break;  
    más cases ...  
    default:  
        instrucciones si la variable o expresión no es igual a ningún valor. No es obligatorio usar el default  
}
```

- 9) Crear una variable llamada **nombre** a la que le asignéis un nombre cualquiera. Luego mostrar con `window.alert` el texto *Hola nombre*, siendo nombre el valor de la variable nombre. Para poder mostrar un texto entre comillas y una variable o expresión, debemos concatenarlas con el operador +. Otra opción es usar interpolación de cadenas.

```
window.alert(valor_o_expresión) // podemos omitir window.
```

Ejemplos de concatenaciones e interpolaciones:

```
const ciudad = "Vigo";  
console.log("Vives en " + ciudad);  
console.log(`Vives en ${ciudad}`);
```

- 10) Como el anterior, pero que muestreo *Hola nombre, bienvenido*.
- 11) Como en el ejercicio 4, crear una variable constante llamada **precio** con el valor 20.4 y otra llamada **cantidad** con el valor 5. Crear una variable constante llamada **total** con el producto de ambas. Luego, si el total es mayor que 100, mostrar un descuento del 10% y el total con ese descuento, si no, mostrar solo el total

Para calcular el descuento multiplicaremos el total por 0.1.

Ejercicios básicos JavaScript 2

12) Crear una variable **numero1** con el valor 3, **numero2** con el valor 5. Intercambiar el valor de ambas variables y mostrarlas. Para realizar el intercambio necesitamos una variable auxiliar y seguir este proceso:

- Asignamos el valor de numero1 a la tercera variable
- Asignamos el valor de numero2 a la variable numero1
- Asignamos el valor de la tercera variable a la variable numero2

13) Modificar el ejercicio 9 para que muestre el nombre en mayúsculas. Para ello podemos usar `nombreVariable.toUpperCase()` (`toLowerCase()` para minúsculas)

Ejercicios gráficos

Lo primero que debemos de hacer es ver que acciones (eventos) del usuario debemos tratar: click, change, keydown, mouseenter, mouseleave, ... Para ello usaremos alguno de los siguientes:

```
document.getElementById("idDelElemento").addEventListener("evento", funciónAejecutar);
```

O la siguiente que devuelve el primer elemento que coincida con el selector indicado

```
document.querySelector("selectorCSS").addEventListener("evento", funciónAejecutar);
```

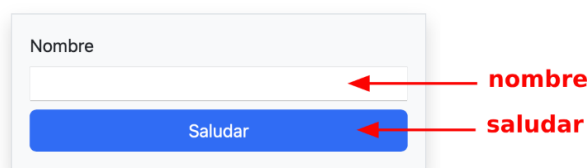
Luego crearemos la función o funciones con la siguiente sintaxis:

```
function nombre(evt) {  
}
```

Donde evt (podemos llamarlo de cualquier manera y es opcional) es un parámetro que contiene información del evento que se ha producido. Algunas propiedades que tiene son:

- `currentTarget`: elemento que ha provocado el evento y al que se suscribió con `addEventListener`
- `target`: elemento que ha provocado el evento y al que se suscribió con `addEventListener` o uno de sus hijos
- `type`: un texto con el nombre del evento (click, mouseenter,)

14) Al pulsar el botón se debe mostrar un `window.alert` que ponga Hola y el nombre escrito en el cuadro de texto



15) Parecido al anterior, pero ahora el saludo debe aparecer en el cuadro de texto saludo

Ejercicios básicos JavaScript 2

Nombre

Saludar

nombre

saludar

saludo

16) A partir de los voltios y los amperios, calcular la potencia usando la fórmula potencia = voltios * amperios

Voltaje (voltios)

Corriente (amperios)

calcular potencia

Potencia (vatios)

voltios

amperios

calcular

potencia

17) Parecido al ejercicio 15, pero ahora, si está marcado el radio español, debe decir Hola y el nombre; si está marcado el radio ingles, debe decir Hello y el nombre

Nombre

☒ Español ☐ Inglés

Saludar

nombre

espanhol

ingles

saludar

saludo

- Debemos comprobar la propiedad **checked** (true o false) de los radios para ver cual está marcado

18) Como el anterior pero ahora el idioma se coge de una lista desplegable

Nombre

Idioma

Español

Saludar

nombre

idioma

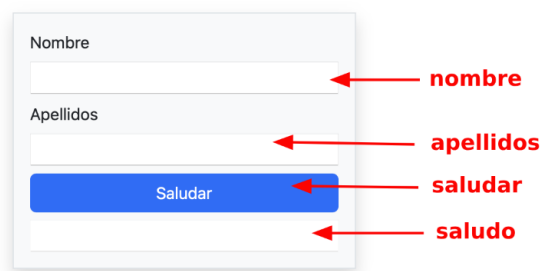
saludar

saludo

- Podemos usar la propiedad value del select para obtener el idioma elegido o selectedIndex (que devuelve 0 si está seleccionado el primero, 1 si el segundo, ...)

19) Parecido al 15, pero ahora debe poner el nombre y los apellidos al saludar

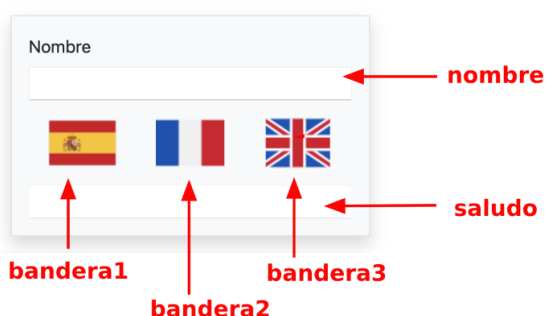
Ejercicios básicos JavaScript 2



20) Como el anterior pero ahora debe saludar de la forma Hola apellidos, nombre. Para ello recordad que tenemos dos opciones ya que la coma es un texto que debe ir entre comillas

- Concatenar las variables, hola y la coma con el operador +
- Usar interpolación de cadenas: todo entre `` y poniendo las variables y expresiones dentro de las `` de la forma `\${nombreVariable}`

21) Parecido al 15, pero ahora con tres idiomas. Además, el saludo debe producirse cuando se pulsa cada una de las imágenes:



- Podemos crear un listener para cada imagen, cada una con su función
- Podemos crear un listener para cada imagen, todas con la misma función y usar `evt.currentTarget.id` en ella para saber que bandera se pulsó
- Podemos crear un solo listener en el div y usar `evt.target.id` en la función para saber que bandera se pulsó (no `currentTarget` pues siempre se refiere al elemento que hizo el `addEventListener`)

Si creamos el listener en el padre nos ahorramos crear un listener en cada hijo; si son muchos nos ahorramos bastante código (aunque veremos posteriormente como crear listeners usando bucles). En este caso hay varias opciones en la función:

- Si el padre solo tiene los hijos para los que queramos los listeners sin huecos entre los hijos y el padre, basta con usar `evt.target` en el hijo
- Si el padre solo tiene los hijos para los que queramos los listeners con huecos entre los hijos y el padre, debemos diferenciar si la función se ejecuta porque ha pasado el evento en los hijos o en el padre. Para ello tenemos varios modos; algunos son:
 - o Comprobar si el target es diferente del `currentTarget`, pues este último siempre será el padre:

```
if(evt.target !== evt.currentTarget) {  
    Es un hijo
```

Ejercicios básicos JavaScript 2

```
}
```

- o Si son diferentes etiquetas, preguntar por ellas:

```
if(evt.target.nodeType == "IMG") {  
    // Si los hijos son img y el padre otra cosa  
}
```

- o Si los hijos que nos interesan tienen alguna clase que no tienen los otros:

```
if(evt.target.classList.contains("nombreClase")) {  
    // Los hijos que tengan esa clase  
}
```

- Si hay otros hijos en el padre además de los que nos interesan podemos preguntar, al igual que antes, por `nodeType`, `contains`,... para diferenciarlos de los otros

22) Al pulsar en cada color el fondo de cuadro de texto debe cambiarse a ese color. En este caso aplicaremos el color de la forma `elemento.style.backgroundColor = "rgb(valor1, valor2, valor3)"`. También podemos asignar el `style.backgroundColor` del `div` pulsado. Como en otros ejercicios, hacer una función listener para cada color o una para todos.

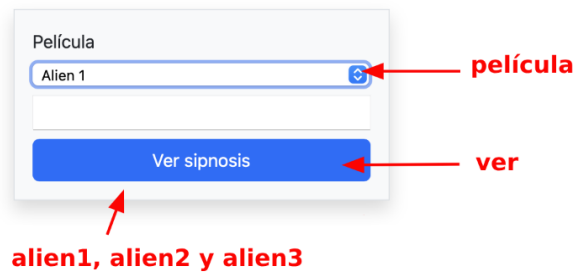


23) Como el ejercicio anterior pero que se ponga el color cuando el puntero del ratón entra en cada uno de los colores (`mouseenter`)

24) Al elegir una película en el select (evento `change`) debemos mostrar la sinopsis de la película. Para ello tenemos ocultos tres `div`, `alien1`, `alien2` y `alien3`, con lo que debemos mostrar el que corresponda y ocultar los otros. Una manera sería:

- Ocultar los `divs` no deseados añadiéndoles la clase `d-none` (`elemento.classList.add("d-none")`). O también ocultar todos que luego mostraremos el deseado.
- Mostrar el `div` deseado quitándole la clase `d-none` (`elemento.classList.remove("d-none")`)

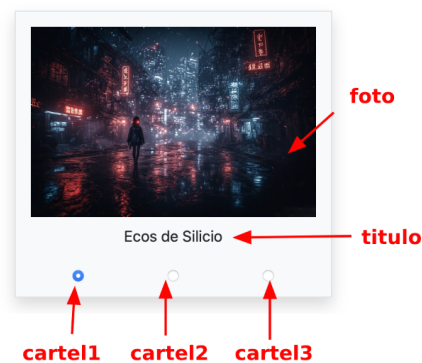
Ejercicios básicos JavaScript 2



25) Al marcar cada radio, evento click, debemos cambiar la foto por cartel1.jpg, cartel2.jpg y cartel3.jpg respectivamente.



26) Como el anterior pero ahora debe aparecer en el elemento título, el título de la película: Ecos de Silicio, La Última Función y El Día que ocurrió

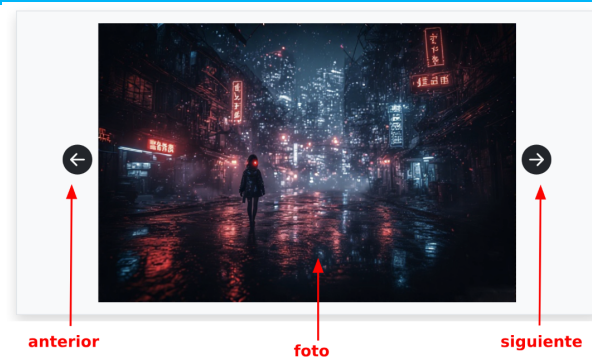


27) Como el ejercicio 15, pero mostrando un window.alert si no se ha introducido el nombre. Darle también el foco al cuadro de texto en este caso (con elemento.focus())

28) Como el ejercicio 16, pero mostrando window.alert si no se ha introducido el voltaje o la corriente.

29) En este caso se irá cambiando la foto cada vez que se pulse el botón siguiente o anterior. Si llegamos al final, que vuelva a mostrar la primera. Para ello puede convenir tener un contador con la foto actual. Para este ejercicio tenemos las fotos cartel1.jpg a cartel6.jpg

Ejercicios básicos JavaScript 2



30) Crear una página que muestre muchas de las banderas que tenemos en el fichero flagpedia.net.zip. Luego debe generarse aleatoriamente el nombre de uno de los países de esas banderas y el usuario debe pulsar en la bandera adecuada. Si acierta se sumará un contador de aciertos y si falla un de fallos. A continuación, mostrar otra bandera y repitiendo el proceso hasta que acierte todas las banderas o tenga 5 fallos.

31) Vamos a simular un medidor de temperaturas donde se irá incrementando o decrementando la temperatura de forma aleatoria. Si esta es 5 o menos, deberá aparecer la imagen baja.png y el color de la temperatura será azul; si es de 15 o más, deberá aparecer la imagen alta.png y el color de la temperatura será rojo. En los demás casos la imagen será normal.png y el color negro. Para ello podemos hacer lo siguiente:

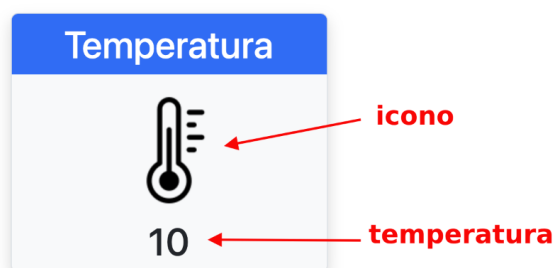
- Tener una variable global con la temperatura inicial de 10
- Usar setInterval para que se ejecute una función cada cierto tiempo:

```
setInterval(nombreFunción, milisegundos);
```

- En la función generar un número aleatorio entre 0 y 1:

```
Math.floor(Math.random() * 2);
```

- Si es 1 incrementamos la temperatura en uno, si es cero la bajamos en uno
- Hacemos unos if para ver si está por debajo de 5, por encima de 15 o el resto



32) Modificar el anterior para que cada vez que la temperatura esté por debajo de 0 grados o por encima de 20, ir eliminando una persona. Para ello usaremos:

```
// no podemos simplemente eliminar el último elemento del div personas pues los enter
// también son nodos, en concreto de texto, y eliminará un nodo texto y la siguiente vez
// una imagen
document.getElementById("personas")
    .removeChild(document.querySelector("#personas img:last-of-type"));
```


Ejercicios básicos JavaScript 2

Si ya no quedan habitantes debemos mostrar el div final:

- Para comprobar si no quedan personas tenemos:

```
document.getElementById("personas").children.length
```

- Para mostrar el div quitamos su clase d-none de Bootstrap:

```
document.getElementById("final").classList.remove("d-none");
```

