

Ejercicios JavaScript

Ejercicio 1

Al pulsar la edad debe aparecer un mensaje (window.alert) indicando que si la persona es mayor de edad (18 o más años) o no. También comprobar que la edad introducida sea correcta

Diagrama de un formulario web titulado "Mayor de Edad". El formulario contiene un campo de entrada etiquetado "Edad:" y un botón verde etiquetado "comprobar". Una flecha roja etiquetada "edad" apunta al campo de entrada, y otra flecha roja etiquetada "comprobar" apunta al botón.

Ejercicio 2

Como el anterior pero mostrando el mensaje en el cuadro de texto resultado

Diagrama de un formulario web titulado "Mayor de Edad". El formulario contiene un campo de entrada etiquetado "Edad:", un botón verde etiquetado "comprobar", y un campo de entrada adicional etiquetado "resultado". Tres flechas rojas indican los elementos: "edad" apunta al campo de entrada, "comprobar" apunta al botón, y "resultado" apunta al campo de entrada adicional.

Ejercicio 3

Cuando el usuario seleccione alguno de los radios, el texto debe cambiar su padding (style.padding) a 0, 5px o 10px según corresponda.

Diagrama de un formulario web titulado "Espaciado". El formulario contiene un grupo de radio buttons etiquetado "Padding" con opciones "0", "5px" y "10px". Debajo de los radios hay un campo de entrada con el texto "Lorem Ipsum". Seis flechas rojas indican los elementos: "espaciado1" apunta al título, "espaciado2" apunta al grupo de radio buttons, "espaciado3" apunta a la opción "10px", y "texto" apunta al campo de entrada.

Ejercicio 4

Al pasar el ratón por cada imagen (mouseover o mousemover) debe cambiarles la opacidad a 1 (style.opacity: 1)

Diagrama de un formulario web titulado "Opacidad". El formulario contiene tres imágenes de niños jugando en un estanque. Debajo de las imágenes hay un campo de entrada. Seis flechas rojas indican los elementos: "imagen1" apunta a la primera imagen, "imagen2" apunta a la segunda imagen, "imagen3" apunta a la tercera imagen, y una flecha roja sin etiqueta apunta al campo de entrada.

Ejercicios JavaScript

Ejercicio 5

Al pasar el ratón por cada imagen (mouseover o mousemove) debe aparecer en info el texto carrera, natación y pértiga respectivamente.



Ejercicio 6

Al seleccionar un deporte en la lista, debe mostrarse en el elemento imagen la imagen correspondiente al deporte (carrera.png, natacion.png y pertiga.png) mediante la propiedad src de la imagen.



Ejercicio 7

Al pulsar cada botón debe reproducirse el sonido correspondiente a cada animal (perro.wav, gato.wmv y mono.wav).



Para reproducir un sonido usaremos un código similar a este:

```
const audio = new Audio('ficheroSonido');
audio.play();
```

Ejercicio 8

Cada vez que se pulse el botón añadir, debe guardarse el nombre en un array. Al pulsar el botón listado debe mostrar todos esos nombres guardados.

Ejercicios JavaScript



Para ello:

- creamos un array global vacío
- para añadir elementos usaremos el método `push(valorDeseado)` del array
- para mostrar los nombres podemos recorrerlos con un `for`, ir creando elementos `div` o `p` con `document.createElement` y añadirlos a `listado` con su método `appendChild`

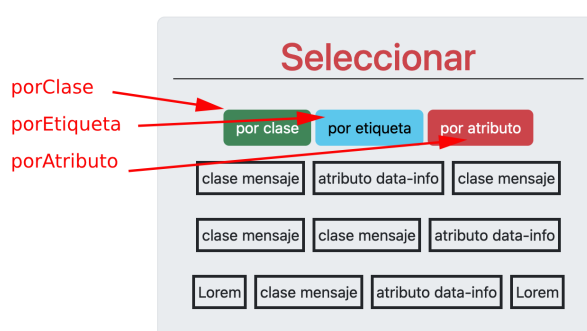
Ejercicio 9

Como el anterior pero añadiendo lo siguiente:

- un botón vaciar que elimine todos los elementos de array, asignando `[]` a la variable
- ordenar los datos antes de mostrarlos en el listado usando el método `sort()` del array
- antes de añadir un nombre comprobar que no se deje vacío (comparar con `""`)

Ejercicio 10

- Al pulsar el botón `porClase` se deben poner todos los elementos con la clase `mensaje` en verde
- Al pulsar el botón `porEtiqueta` se deben poner todos los `p` con `en` en celeste
- Al pulsar el botón `porAtributo` se deben poner todos los elementos con el atributo `data-info` en rojo

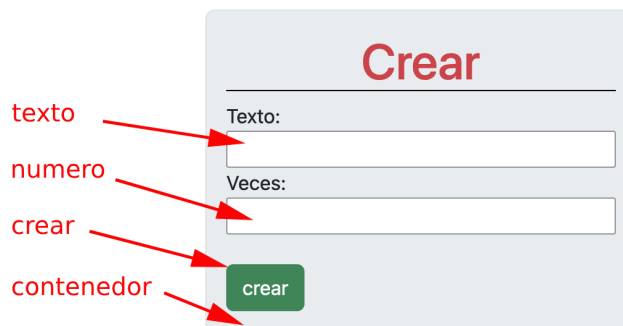


Para ello usaremos `getElementById`, `getElementsByClassName`, `getElementsByTagName`, `querySelector` o `querySelectorAll` según el caso y recorrerlos con un `for of`. Luego cambiamos el color con `style.color = "nombreColorInglés"`.

Ejercicio 11

Escribe un texto y un número. Al pulsar el botón crear añadir tantas etiquetas `p` en el `div` contenedor como veces se hayan introducido en número. Estos `p` tendrán el texto introducido en el cuadro de texto llamado `texto`.

Ejercicios JavaScript



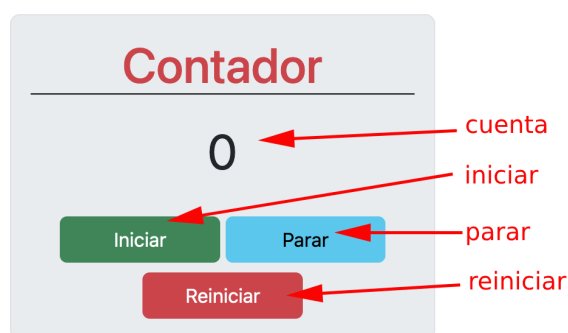
Ejercicio 11

Al pulsar el botón + el contador se debe incrementar en 1; al pulsar el botón - se debe disminuir en 1. Para ello es útil tener una variable global contador que vayamos incrementando o decrementando.



Ejercicio 12

Al pulsar el botón Iniciar se debe ir incrementando el contador cada segundo. Al pulsar el botón Parar se debe para la cuenta. Al pulsar el botón Reiniciar se debe poner a cero la cuenta.



Para ello:

- creamos una variable global llamada reloj, u otro nombre, de la forma let reloj;
- también un contador global que comience en cero
- en el botón Iniciar debemos poner en marcha el contador usando la función setInterval que ejecutará una función continuamente según el tiempo que le indiquemos:

```
reloj = setInterval(nombreFunción, milisegundos); // En nuestro caso 1000 milisegundos
```

- en la función incrementaremos el contador en 1 y lo mostraremos
- en el botón Parar detendremos el reloj con:

Ejercicios JavaScript

```
clearInterval(reloj);
```

- en el botón Reiniciar simplemente pondremos el contador a cero

Ejercicio 13

Dada una base y un exponente, calcular su potencia, es decir, la base elevada al exponente. Por ejemplo 2 elevado a 8 es $2 * 2 * 2 * 2 * 2 * 2 * 2 * 2$ (multiplicar 8 veces 2). Para ello necesitamos un acumulador que comience en 1 y un bucle que vaya hasta el exponente.

Diagrama de una interfaz de usuario para calcular la potencia. El título es "Potencia". Hay dos campos de entrada: "Base:" y "Exponente:". Un botón verde con el texto "calcular" está entre los campos. Debajo del botón hay un campo de salida etiquetado como "Resultado:". Seis flechas rojas indican las partes de la interfaz: "base" apunta al campo "Base:", "exponente" apunta al campo "Exponente:", "calcular" apunta al botón, y "resultado" apunta al campo "Resultado:".