

Manejo del DOM



Continuamos con el manejo del DOM en este ejercicio realizaremos un contador.

Aquí el código HTML:

```
<div class="container">
  <h1>Contador</h1>
  <p class="contar">0</p>
  <div class="contadores">
    <button class="btn dis">Disminuir</button>
    <button class="btn inc">Incrementar</button>
  </div>
  <button class="reset">Reset</button>
</div>
<script src="js/script.js"></script>
```



@hdtolledo

Nuestro CSS:

```
1  @import url('https://fonts.googleapis.com/css2?family=Poppins:wght@400;500;700&display=swap');
2
3  *{
4      font-family: "Poppins", sans-serif;
5  }
6
7  body{
8      height: 100vh;
9      display: flex;
10     justify-content: center;
11     align-items: center;
12     overflow-y: hidden;
13 }
14
15 .container{
16     background-color: white;
17     color: black;
18     width: 400px;
19     min-height: 300px;
20     border-radius: 10px;
21     box-shadow: inset 5px 5px 10px
22                 rgba(0, 0, 0, 0.3),
23                 inset -5px -5px 10px
24                 rgba(0, 0, 0, 0.3);
25     display: flex;
26     align-items: center;
27     flex-direction: column;
28     justify-content: center;
29 }
30
31 h1{
32     margin-bottom: 10px;
33     text-transform: uppercase;
34     color: blueviolet;
35 }
```



@hdtoledo

```

.contar{
  font-size: 70px;
}

.contadores{
  display: flex;
  justify-content: center;
  align-items: center;
  gap: 140px;
}

.btn{
  color: white;
  padding: 10px 20px;
  outline: none;
  border: 1px solid black;
  background-color: blueviolet;
  box-shadow: 4px 4px 0 black;
  cursor: pointer;
}

.btn:active{
  position: relative;
  top: 4px;
  left: 4px;
  box-shadow: none;
}

.reset{
  color: white;
  background-color: black;
  padding: 3px 10px;
  outline: none;
  border: none;
  cursor: pointer;
  margin-bottom: 10px;
}

```

```

▼ .reset:active{
  transform: scale(0.95);
}

```



@hdtoledo

Y por último nuestro JS:

```
const btnIncre = document.querySelector(".inc");
const btnDismi = document.querySelector(".dis");
const btnReset = document.querySelector(".reset");
const botones = document.querySelector(".btn");
const contar = document.querySelector(".contar");

let counts = 0;

btnIncre.addEventListener("click", incrementar);
btnDismi.addEventListener("click", disminuir);
btnReset.addEventListener("click", reset);
```

En primer lugar, se definen cinco constantes utilizando la función **document.querySelector()** para seleccionar elementos HTML con las clases **.inc**, **.dis**, **.reset**, **.btn**, y **.contar**.

Después, se define una variable **counts** inicializada en 0, que será utilizada para llevar la cuenta de los incrementos y disminuciones del contador.

Luego, se agregan tres eventos de escucha a los botones **.inc**, **.dis**, y **.reset** mediante el método **addEventListener()**. Los eventos son **"click"** y las funciones que se ejecutarán al hacer clic son **incrementar()**, **disminuir()**, y **reset()**, respectivamente.

Finalmente, se definen las tres funciones **incrementar()**, **disminuir()**, y **reset()**, que serán ejecutadas al hacer clic en los botones correspondientes. La función **incrementar()** incrementa la variable **counts** en 1 y actualiza el contenido del elemento HTML con la clase **.contar** con el nuevo valor. La función **disminuir()** decrementa la variable **counts** en 1 y actualiza el contenido del elemento HTML con la clase **.contar** con el nuevo valor. La función **reset()** reinicia la variable **counts** a 0 y actualiza el contenido del elemento HTML con la clase **.contar** con el valor 0.



@hdtoledo

```

function incrementar(){
    counts++;
    contar.innerHTML = counts;
    if(counts>0){
        contar.style.color = "green";
    }
    if(counts == 0){
        contar.style.color = "black";
    }
}

```

- La primera línea de la función "incrementar" incrementa la variable "counts" en 1 cada vez que se hace clic en el botón.
- La segunda línea actualiza el contenido HTML del elemento "contar" con el valor actual de la variable "counts".
- En la tercera línea, si el valor de "counts" es mayor que cero, el color del texto en el elemento "contar" cambia a verde. Esto se utiliza para indicar que el número es positivo.
- En la cuarta línea, si el valor de "counts" es igual a cero, el color del texto en el elemento "contar" cambia a negro. Esto se utiliza para indicar que el número es neutro o cero.

En resumen, esta función es responsable de incrementar una variable cada vez que se hace clic en el botón, actualizar un elemento HTML con el valor actual de esa variable y cambiar el color del texto en ese elemento en función del valor actual de la variable.

```

function disminuir(){
    counts--;
    contar.innerHTML = counts;
    if(counts<0){
        contar.style.color = "red";
    }
    if(counts == 0){
        contar.style.color = "black";
    }
}

```

Este código es una función llamada **disminuir** que decrementa en 1 la variable **counts**, que se utiliza para llevar un conteo. Luego, se actualiza el contenido HTML del elemento **contar** con el nuevo valor de **counts**.

Además, la función incluye dos estructuras condicionales **if**. La primera condición verifica si **counts** es menor que cero, y si es así, cambia el color del texto del elemento **contar** a rojo. La segunda condición verifica si **counts** es igual a cero, y si es así, cambia el color del texto del elemento **contar** a negro. En resumen, esta función es utilizada para decrementar el contador y cambiar su color según su valor.



@hdtoledo

```
function reset(){  
  counts = 0;  
  contar.innerHTML = counts;  
  contar.style.color = "black";  
}
```

La función **reset** se utiliza para restablecer el contador a cero y mostrarlo en la página HTML.

Primero, se establece la variable **counts** en 0, lo que significa que el contador se restablecerá a cero. Luego, se actualiza el contenido de la etiqueta HTML que muestra el contador estableciendo su propiedad **innerHTML** en el valor actual de la variable **counts**, que es 0 en este caso.

Finalmente, se establece el color del texto en negro, utilizando la propiedad **style.color** de la etiqueta HTML que muestra el contador, para que sea visible en la página.



@hdtolledo