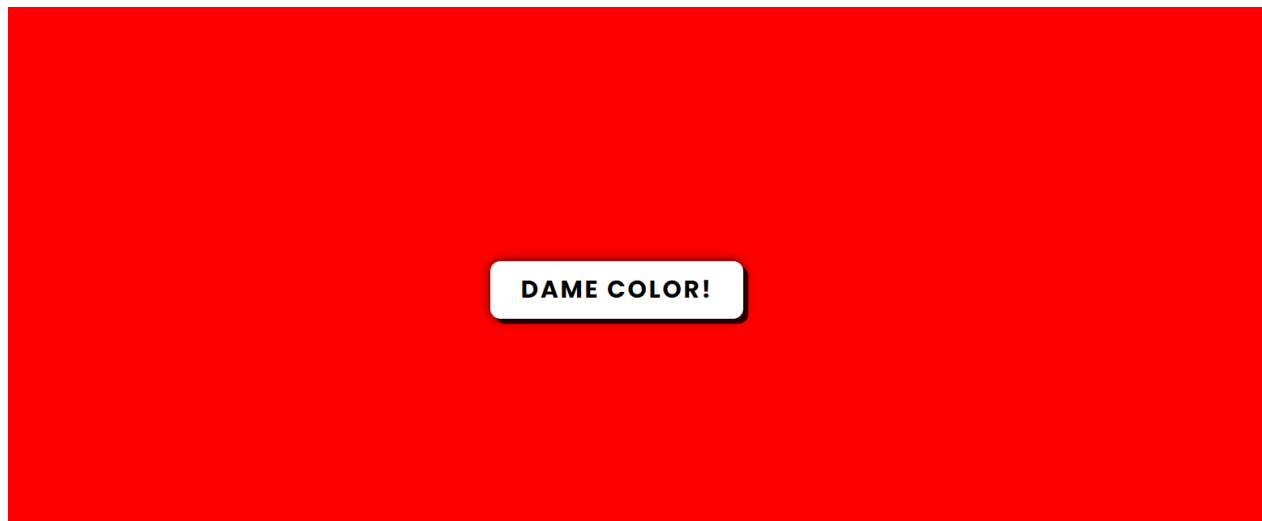
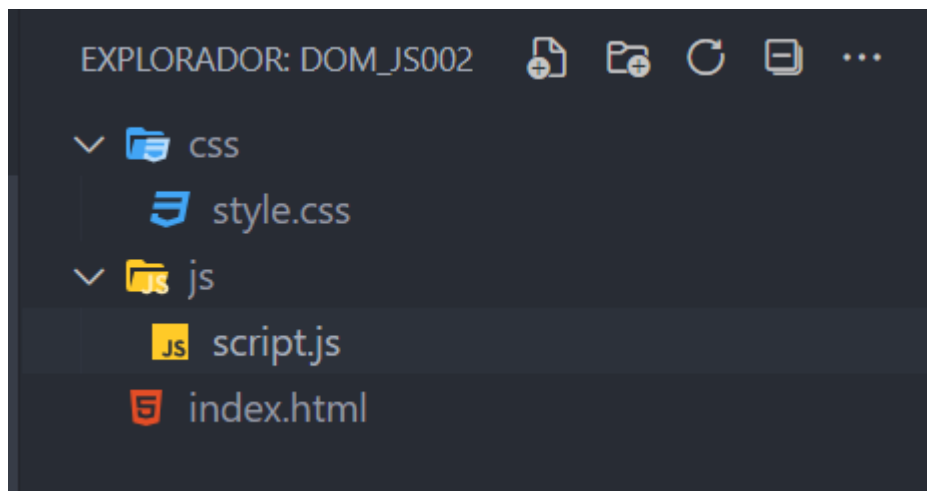


Color Manejo del DOM



Continuamos con el manejo del DOM en este ejercicio realizaremos un botón que nos permitirá manipular el color del fondo de nuestro DOM.

Estructura de nuestro proyecto:



@hdtoledo

Aquí el código HTML:

```
index.html > html > body
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/font-awesome@6.4.0/css/all.min.css" integrity="sha512-iecdLmaskl7CVkqkXNQ/ZH/XLlvWZOJyj7Yy7tcenmpD1ypASozpmT/E0iPtmFIB46ZmdtAc9eNBvH0H/ZpiBw==" crossorigin="anonymous" referrerpolicy="no-referrer" />
8   <link rel="stylesheet" href="css/style.css">
9   <title>Manejo del DOM 002</title>
10 </head>
11 <body>
12   <button class="btn">Dame Color!</button>
13   <script src="js/script.js"></script>
14 </body>
15 </html>
```

Resumen del código html inicial:

1. **!DOCTYPE html**: Esto define la versión de HTML que se está utilizando, en este caso, HTML5.
2. **<html lang="en">**: Aquí comienza el elemento HTML, con la especificación del lenguaje "en" (inglés).
3. **<head>**: Dentro de la cabeza del documento, se incluyen metadatos y enlaces a recursos externos, como archivos de estilo y scripts.
 - **<meta charset="UTF-8">**: Establece la codificación de caracteres del documento como UTF-8, que es una codificación común para texto en la web.
 - **<meta http-equiv="X-UA-Compatible" content="IE=edge">**: Especifica la compatibilidad con Internet Explorer.
 - **<meta name="viewport" content="width=device-width, initial-scale=1.0">**: Define las características de la ventana gráfica, asegurando un correcto escalado en dispositivos móviles.
 - **<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/font-awesome@6.4.0/css/all.min.css" integrity="sha512-iecdLmaskl7CVkqkXNQ/ZH/XLlvWZOJyj7Yy7tcenmpD1ypASozpmT/E0iPtmFIB46ZmdtAc9eNBvH0H/ZpiBw==" crossorigin="anonymous" referrerpolicy="no-referrer" />**: Enlaza la hoja de estilos de Font Awesome, una biblioteca de iconos.
 - **<link rel="stylesheet" href="css/style.css">**: Enlaza otra hoja de estilos personalizada ubicada en la carpeta "css".
 - **<title>Manejo del DOM 002</title>**: Establece el título de la página que se mostrará en la pestaña del navegador.
4. **<body>**: Aquí comienza el cuerpo del documento, que contendrá el contenido visible de la página.
 - **<button class="btn">Dame Color!</button>**: Se crea un botón con la clase "btn" y el texto "Dame Color!".



@hdtoledo

- **<script src="js/script.js"></script>**: Vincula un archivo JavaScript externo llamado "script.js" ubicado en la carpeta "js". Este archivo contendrá código para manipular el Document Object Model (DOM) cuando se haga clic en el botón.

Nuestro código CSS:

```
css > style.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 .btn{
16     outline: none;
17     border: none;
18     text-transform: uppercase;
19     letter-spacing: 2px;
20     font-size: 24px;
21     background-color: white;
22     padding: 10px 30px;
23     border-radius: 10px;
24     font-weight: bold;
25     color: black;
26     cursor: pointer;
27     box-shadow: 5px 5px 0px rgba(0, 0, 0, 0.8),
28                0 0 10px rgba(0, 0, 0, 0.8);
29 }
30
31 .btn:active{
32     outline: none;
33     position: relative;
34     top: 10px;
35     left: 2px;
36     box-shadow: none;
37 }
```

Resumen del CSS:

1. **@import**
url('https://fonts.googleapis.com/css2?family=Poppins:wght@400;500;700&display=swap');:
Importa la fuente "Poppins" desde Google Fonts. La fuente está disponible en tres pesos diferentes: 400 (normal), 500 (medio) y 700 (negrita).
2. ***{ font-family: "Poppins", sans-serif; }**: Aplica la fuente "Poppins" a todos los elementos de la página, utilizando "sans-serif" como fuente de respaldo si "Poppins" no está disponible.
3. **body{ height: 100vh; display: flex; justify-content: center; align-items: center; overflow-y: hidden; }**: Establece estilos para el cuerpo de la página.
 - **height: 100vh;**: Fija la altura del cuerpo al 100% de la altura de la ventana visible (viewport height).
 - **display: flex;**: Utiliza el modelo de diseño flexible (flexbox) para organizar los elementos hijos.



@hdtoledo

- **justify-content: center;;** Centra los elementos horizontalmente en el contenedor.
 - **align-items: center;;** Centra los elementos verticalmente en el contenedor.
 - **overflow-y: hidden;;** Oculta la barra de desplazamiento vertical en el eje Y.
4. **.btn{ ... }**: Define estilos para un elemento con la clase "btn", que probablemente se refiere al botón de la página.
- **outline: none; border: none;;** Elimina el contorno y el borde predeterminados del botón.
 - **text-transform: uppercase;;** Convierte el texto del botón a mayúsculas.
 - **letter-spacing: 2px;;** Agrega un espaciado de 2 píxeles entre las letras.
 - **font-size: 24px;;** Establece el tamaño de la fuente del botón en 24 píxeles.
 - **background-color: white;;** Establece el color de fondo del botón en blanco.
 - **padding: 10px 30px;;** Define el relleno del botón.
 - **border-radius: 10px;;** Agrega bordes redondeados al botón.
 - **font-weight: bold;;** Hace que el texto del botón sea negrita.
 - **color: black;;** Establece el color del texto del botón en negro.
 - **cursor: pointer;;** Cambia el cursor a una mano cuando pasa sobre el botón, indicando que es interactivo.
 - **box-shadow: ...:** Agrega una sombra al botón para darle un efecto tridimensional.
5. **.btn:active{ ... }**: Define estilos específicos cuando el botón está en estado activo (cuando se está presionando).
- **outline: none;;** Elimina el contorno cuando el botón está activo.
 - **position: relative; top: 10px; left: 2px;;** Desplaza ligeramente el botón hacia abajo y hacia la derecha cuando está activo, para dar un efecto de presión.
 - **box-shadow: none;;** Elimina la sombra cuando el botón está activo.



@hdtoledo

Nuestro código JS:

```
js > script.js > ...
1  const btnColor = document.querySelector(".btn");
2  const bodyColor = document.querySelector("body");
3
4  const colores = ["green", "blue", "red", "orange", "yellow", "violet"];
5  let count = 0;
6
7  bodyColor.style.background = "pink";
8  btnColor.addEventListener("click", cambiarColor);
9
10 function cambiarColor(){
11     console.log("Si cambia de color");
12     const color = parseInt(Math.random()*colores.length);
13     bodyColor.style.background = colores[color];
14     count++;
15 }
```

Resumen del código JS:

1. **const btnColor = document.querySelector(".btn");**: Selecciona el primer elemento con la clase "btn" en el documento y lo asigna a la variable **btnColor**.
2. **const bodyColor = document.querySelector("body");**: Selecciona el primer elemento **<body>** en el documento y lo asigna a la variable **bodyColor**.
3. **const colores = ["green", "blue", "red", "orange", "yellow", "violet"];**: Crea un array llamado **colores** que contiene diferentes colores.
4. **let count = 0;**: Declara una variable **count** e inicializa su valor en 0. Esta variable se utilizará para contar cuántas veces se ha hecho clic en el botón.
5. **bodyColor.style.background = "pink";**: Establece el color de fondo inicial del cuerpo de la página en rosa.
6. **btnColor.addEventListener("click", cambiarColor);**: Agrega un evento de clic al botón con la clase "btn" para ejecutar la función **cambiarColor** cuando se hace clic en el botón.
7. **function cambiarColor() { ... }**: Define la función **cambiarColor** que se ejecuta cuando se hace clic en el botón.
 - **console.log("Si cambia de color");**: Imprime en la consola el mensaje "Si cambia de color" cada vez que se hace clic en el botón. Esto es útil para propósitos de depuración.
 - **const color = parseInt(Math.random() * colores.length);**: Genera un índice aleatorio basado en la longitud del array **colores** y lo convierte a un número entero utilizando **parseInt**. El índice se guarda en la variable **color**.
 - **bodyColor.style.background = colores[color];**: Cambia el color de fondo del cuerpo de la página al color correspondiente al índice generado aleatoriamente.



@hdtoledo

- **count++;** Incrementa el valor de la variable **count** en 1 cada vez que se hace clic en el botón. Esto puede ser utilizado para llevar un registro de cuántas veces se ha cambiado el color.



@hdtoledo