

getElementById

Es una función de JavaScript que te permite acceder a un elemento HTML en el DOM (Document Object Model) utilizando su atributo id. Devuelve el objeto del elemento si encuentra uno con el ID especificado, o null si no existe.

En detalle:

- **Función:** getElementById es una función predefinida en el objeto document (que representa el documento HTML en el navegador).
- **Uso:** Se usa para obtener un elemento específico del DOM, basándose en su atributo id.
- **Sintaxis:** document.getElementById("el_id_del_elemento").
- **Retorno:** Devuelve un objeto que representa el elemento HTML con el ID especificado, o null si no se encuentra ningún elemento con ese ID.
- **Importancia:** Permite manipular el elemento directamente una vez que lo has obtenido.

Ejemplo:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Funciones</title>
</head>
<body>
  <p>Este texto no es modificado</p>
  <p id="miParrafo">Este es un párrafo.</p>
  <script>
    let parrafo = document.getElementById("miParrafo");
    parrafo.textContent = "Con este texto se modifica....."; // Modifica el contenido del párrafo
  </script>
</body>
</html>
```

innerHTML

Es una propiedad de JavaScript que permite obtener o establecer el contenido HTML de un elemento HTML. En otras palabras, permite leer y modificar el código HTML dentro de un elemento específico en una página web.

¿Cómo funciona?

- **Obtener el contenido:**

Cuando se utiliza para obtener el contenido de un elemento, innerHTML devuelve una cadena que contiene todo el marcado HTML dentro de ese elemento, incluyendo texto, etiquetas HTML y otros elementos anidados.

Intecap

- **Establecer el contenido:**

Cuando se utiliza para establecer el contenido, innerHTML reemplaza todo el contenido HTML existente dentro del elemento con la nueva cadena proporcionada.

Usos comunes:

- **Modificar la estructura de una página web dinámicamente:**

innerHTML se utiliza a menudo para actualizar la apariencia de una página web sin necesidad de recargarla completamente.

- **Insertar contenido dinámico:**

Permite agregar o eliminar elementos HTML, texto, etc., basados en interacciones del usuario o eventos.

- **Renderizar datos de forma dinámica:**

Se puede utilizar para mostrar información obtenida de una base de datos o de una API en la página web.

Consideraciones:

- innerHTML puede ser vulnerable a ataques de seguridad si se utiliza para insertar contenido generado por el usuario sin la debida validación.
- En algunos casos, textContent o innerText podrían ser alternativas más eficientes, especialmente si solo se necesita manipular el contenido textual del elemento.

Ejemplo innerHTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Funciones</title>
</head>
<body>
  <p id="texto">Alex Hernandez</p>

  <p>El contenido de "texto" es:</p>
  <p id="miElemento"><strong></strong></p>

  <script>
    let elemento = document.getElementById("texto").innerHTML;
    // Obtener el contenido HTML
    //const contenido = elemento.innerHTML; // Resultado: "Este es un <strong>texto en negrita</strong>."
    document.getElementById("miElemento").innerHTML = elemento;
  </script>
</body>
</html>
```

Ejemplo de cambio de texto en Html.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Funciones</title>
</head>
<body>
  <h1>Insertando texto en el Html</h1>
  <h2>La propiedad innerHTML</h2>

  <p id="demo" onclick="myFunction()">Clic para cambiar el conenido Html (innerHTML).</p>

  <script>
    function myFunction() {document.getElementById("demo").innerHTML = "Listo se cambio...";}
  </script>
</body>
</html>
```



Ejemplo: Diferencia entre innerText, innerHTML y textContent

```
<!DOCTYPE html>
<html>
<body>
  <h1>El elemento Object</h1>
  <h2>Diferencias entre innerText, innerHTML y textContent</h2>

  <p id="myP"> Este elemento tendrá un espaciado extra <span>un elemento span</span>.</p>

  <button onclick="getinnerHTML()">Get innerHTML</button>
  <button onclick="getinnerText()">Get innerText</button>
  <button onclick="gettextContent()">Get textContent</button>
  <pre id="demo"></pre>

  <script>
    function getinnerText() {
      let text = document.getElementById("myP").innerText;
      document.getElementById("demo").innerText = text;
    }

    function getinnerHTML() {
      let text = document.getElementById("myP").innerHTML;
      document.getElementById("demo").innerText = text;
    }

    function gettextContent() {
      let text = document.getElementById("myP").textContent;
      document.getElementById("demo").innerText = text;
    }
  </script>
</body>
</html>
```

Eventos de JavaScript

Los eventos HTML son **"cosas"** que les suceden a los elementos HTML.

Cuando se utiliza JavaScript en páginas HTML, JavaScript puede **"reaccionar"** a estos eventos.

Los eventos HTML son **"cosas"** que les suceden a los elementos HTML.

Cuando se utiliza JavaScript en páginas HTML, JavaScript puede **"reaccionar"** a estos eventos.

Eventos HTML comunes

A continuación, se muestra una lista de algunos eventos HTML comunes:

Evento	Descripción
onchange	Se ha cambiado un elemento HTML
onclick	El usuario hace clic en un elemento HTML
onmouseover	El usuario mueve el ratón sobre un elemento HTML
onmouseout	El usuario aleja el ratón de un elemento HTML
onkeydown	El usuario presiona una tecla del teclado
onload	El navegador ha terminado de cargar la página.

Ejemplo de extraer información de elementos de un formulario.

```
<!DOCTYPE html>
<html>
<head>
  <title>Mostrar datos de formulario</title>
</head>
<body>
  <form id="miFormulario">
    <label>Nombre:</label>
    <input type="text" id="nombre" name="nombre"><br><br>
    <label>Apellido:</label>
    <input type="text" id="apellido" name="apellido"><br><br>
    <button type="button" onclick="mostrarDatos()">Mostrar datos</button>
    <select name="mensajes" id="mensajes">
      <option value="Hola 😊">Hola 😊</option>
      <option value="Buena noche 😊">Buena noche 😊</option>
      <option value="Feliz noche 😊">Feliz noche 😊</option>
    </select>
  </form>

  <div id="datosMostrados"></div>

  <script>
    function mostrarDatos() {
      let nombre = document.getElementById("nombre").value;
      let apellido = document.getElementById("apellido").value;
      let mensajes = document.getElementById("mensajes").value;
      let datosMostrados = document.getElementById("datosMostrados");
      datosMostrados.innerHTML = "<hr>" + "Nombre: " + nombre + "<br>" + "Apellido: " + apellido + " Mensaje: " + mensajes + "<hr>";
    }
  </script>
</body>
</html>
```