

DOCENTE:

Carlos Alberto Lopez Castellanos

MATERIA:

Ejercicios básicos con JS - Primera Parte

PROGRAMACIÓN WEB

ALUMNO:

EDVAN DANIEL MERCADO LAMAS.

CIUDAD:

Mexicali, B.C.

FECHA:

7/03/2024

```
document.addEventListener("DOMContentLoaded", function() {...})
```

`document.addEventListener("DOMContentLoaded", function() {...})` es un método en JavaScript que se utiliza para ejecutar una función cuando el documento HTML ha sido completamente cargado y parseado, es decir, cuando todos los elementos del DOM (Document Object Model) están disponibles para ser manipulados por JavaScript. Esto es importante porque garantiza que el código JavaScript se ejecute en el momento adecuado, evitando posibles errores causados por intentar acceder a elementos del DOM que aún no se han cargado.

En el código proporcionado, la función anónima dentro de `addEventListener` se ejecutará cuando el evento `DOMContentLoaded` se dispare, lo que indica que el documento HTML ha sido cargado completamente. Dentro de esta función, se realiza la inicialización de la página, como la asignación de eventos a botones, la manipulación de elementos del DOM, o cualquier otra acción que sea necesaria una vez que la página esté lista para interactuar con el usuario.

Explicación:

- `document.getElementById`
- `document.createTextNode`
- `document.createElement`
- `appendChild()`

document.getElementById: Este método se utiliza para obtener una referencia a un elemento del documento HTML utilizando su ID. Aquí tienes un ejemplo:

```
<!DOCTYPE html>

<html>

<head>

    <title>getElementById Example</title>

</head>

<body>

    <div id="miDiv">Hola Mundo</div>

    <script>

        var div = document.getElementById("miDiv");

        console.log(div.innerText); // Imprime "Hola Mundo" en la consola

    </script>

</body>
```

```
</html>
```

En este ejemplo, `var div = document.getElementById("miDiv");` obtiene una referencia al elemento `div` con ID "miDiv", y luego `console.log(div.innerText);` imprime el texto contenido en ese `div`, que es "Hola Mundo".

document.createTextNode: Este método se utiliza para crear un nuevo nodo de texto. Aquí un ejemplo:

```
<!DOCTYPE html>

<html>

<head>

    <title>createTextNode Example</title>

</head>

<body>

    <div id="miDiv"></div>

    <script>

        var div = document.getElementById("miDiv");

        var texto = document.createTextNode("Este es un texto creado
dinámicamente");
```

```
div.appendChild(texto) ;

</script>

</body>

</html>
```

En este ejemplo, `var texto = document.createTextNode("Este es un texto creado dinámicamente");` crea un nuevo nodo de texto con el contenido especificado, y luego `div.appendChild(texto);` añade este nodo de texto como hijo del `div`.

document.createElement: Este método se utiliza para crear un nuevo elemento HTML. Aquí tienes un ejemplo:

```
<!DOCTYPE html>

<html>

<head>

    <title>createElement Example</title>

</head>

<body>

    <div id="miDiv"></div>

    <script>
```

```
var div = document.getElementById("miDiv");

var parrafo = document.createElement("p");

parrafo.textContent = "Este es un párrafo creado
dinámicamente";

div.appendChild(parrafo);

</script>

</body>

</html>
```

En este ejemplo, `var parrafo = document.createElement("p");` crea un nuevo elemento `<p>`, y luego `parrafo.textContent = "Este es un párrafo creado dinámicamente";` establece el texto del párrafo. Finalmente, `div.appendChild(parrafo);` añade el párrafo como hijo del `div`.

appendChild(): Este método se utiliza para añadir un nuevo nodo como hijo de un elemento existente. Aquí tienes un ejemplo:

```
<!DOCTYPE html>

<html>

<head>
```

```
<title>appendChild Example</title>

</head>

<body>

    <div id="miDiv"></div>

    <script>

        var div = document.getElementById("miDiv");

        var parrafo = document.createElement("p");

        var texto = document.createTextNode("Este es un texto creado
dinámicamente");

        parrafo.appendChild(texto);

        div.appendChild(parrafo);

    </script>

</body>

</html>
```

En este ejemplo, `parrafo.appendChild(texto);` añade el nodo de texto como hijo del párrafo, y luego `div.appendChild(parrafo);` añade el párrafo como hijo del div.

En conclusión, los métodos `document.getElementById`, `document.createTextNode`, `document.createElement` y `appendChild()` son herramientas fundamentales en JavaScript para manipular el contenido y la estructura de un documento HTML de manera dinámica. Estas funciones son fundamentales para el desarrollo web dinámico, ya que permiten a los desarrolladores crear y manipular elementos HTML en respuesta a eventos o acciones del usuario, lo que mejora la interactividad y la experiencia del usuario en la página web.