**HTML (HyperText Markup Language)**

HTML es el lenguaje de marcado utilizado para estructurar el contenido en la web. Cada página web está construida en HTML y utiliza etiquetas para organizar los elementos. Algunas de las etiquetas esenciales son:

1. **Estructura básica de HTML:**

html

Copiar código

<!DOCTYPE html>

<html lang="es">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Título de la Página</title>

</head>

<body>

<h1>Título Principal</h1>

<p>Este es un párrafo.</p>

</body>

</html>

1. **Etiquetas Comunes:**
   * <h1>...<h6>: Títulos, donde <h1> es el de mayor importancia.
   * <p>: Párrafo de texto.
   * <a href="url">: Enlace a otra página.
   * <img src="url" alt="descripción">: Imagen con texto alternativo.
   * <div> y <span>: Contenedores para agrupar contenido y aplicar estilos.
2. **Atributos Importantes:**
   * class y id: Para aplicar estilos y seleccionar elementos en CSS o JavaScript.
   * href (en enlaces) y src (en imágenes): Especifican destinos y fuentes.
   * alt: Descripción alternativa para imágenes.
3. **Listas:**

html

Copiar código

<ul>

<li>Elemento de lista sin orden</li>

</ul>

<ol>

<li>Elemento de lista ordenada</li>

</ol>

1. **Formularios:**

html

Copiar código

<form action="url\_de\_envio" method="POST">

<input type="text" name="nombre" placeholder="Escribe tu nombre">

<button type="submit">Enviar</button>

</form>

**JavaScript**

JavaScript es un lenguaje de programación que se usa para añadir interactividad a las páginas web. Se puede integrar directamente en HTML o en archivos externos (.js).

1. **Insertar JavaScript en HTML:**

html

Copiar código

<script src="javaScript.js"></script>

1. **Sintaxis Básica:**
   * **Variables:**

javascript

Copiar código

var nombre = "Yeray";

let edad = 25;

const PI = 3.14;

* + **Tipos de datos:** string, number, boolean, array, object, null, undefined.
  + **Operadores:** +, -, \*, /, %, ++, --.

1. **Funciones:**

javascript

Copiar código

function saludar(nombre) {

console.log("Hola, " + nombre);

}

saludar("Yeray");

1. **Condicionales:**

javascript

Copiar código

let edad = 18;

if (edad >= 18) {

console.log("Eres mayor de edad");

} else {

console.log("Eres menor de edad");

}

1. **Bucles:**
   * **for**:

javascript

Copiar código

for (let i = 0; i < 5; i++) {

console.log(i);

}

* + **while**:

javascript

Copiar código

let contador = 0;

while (contador < 5) {

console.log(contador);

contador++;

}

1. **Manipulación del DOM:** JavaScript permite interactuar y modificar el DOM (Document Object Model) de la página.

javascript

Copiar código

document.getElementById("miElemento").textContent = "Nuevo contenido";

1. **Eventos:** Los eventos son acciones que ocurren en el navegador (clics, carga, etc.). Puedes agregar eventos así:

javascript

Copiar código

document.getElementById("miBoton").addEventListener("click", function() {

alert("¡Has hecho clic en el botón!");

});

1. **Arrays y Objetos:**
   * **Array**:

javascript

Copiar código

let frutas = ["Manzana", "Plátano", "Fresa"];

frutas.push("Naranja");

console.log(frutas);

* + **Objeto**:

javascript

Copiar código

let persona = {

nombre: "Yeray",

edad: 30,

saludar: function() {

console.log("Hola, soy " + this.nombre);

}

};

persona.saludar();

1. **Ámbito (Scope) y Closure:**
   * Variables definidas con let y const tienen un ámbito de bloque, mientras que var tiene ámbito de función.
   * Los **closures** permiten que una función tenga acceso a variables de su contexto exterior.
2. **Asincronía y Promesas:**
   * **setTimeout** y **setInterval** permiten ejecutar funciones de manera retrasada o repetida.
   * **Promesas** se usan para manejar operaciones asincrónicas, como peticiones a una API:

javascript

Copiar código

fetch("https://api.example.com/data")

.then(response => response.json())

.then(data => console.log(data))

.catch(error => console.error("Error:", error));

Espero que estos apuntes te resulten útiles. ¿Quieres que amplíe o detalle algún tema en particular?

4o

**Tú dijiste:**

incluye mas etiquetas de html y cosas de javascript

**ChatGPT Plus**

ChatGPT

Aquí tienes una versión ampliada, con más etiquetas de HTML y algunos conceptos adicionales de JavaScript.

**HTML Avanzado**

**1. Más Etiquetas de HTML:**

* **<header> y <footer>**: Se usan para el encabezado y pie de una página o sección.

html

Copiar código

<header>

<h1>Encabezado Principal</h1>

</header>

<footer>

<p>&copy; 2024 Mi Sitio Web</p>

</footer>

* **<nav>**: Contenedor para enlaces de navegación.

html

Copiar código

<nav>

<a href="#inicio">Inicio</a>

<a href="#servicios">Servicios</a>

<a href="#contacto">Contacto</a>

</nav>

* **<section> y <article>**: <section> agrupa contenido temático y <article> se usa para contenidos independientes como publicaciones o comentarios.

html

Copiar código

<section>

<article>

<h2>Título del Artículo</h2>

<p>Contenido del artículo...</p>

</article>

</section>

* **<aside>**: Representa contenido relacionado, como una barra lateral.

html

Copiar código

<aside>

<h3>Noticias Relacionadas</h3>

<p>Información adicional...</p>

</aside>

* **<table>**: Para crear tablas.

html

Copiar código

<table>

<tr>

<th>Nombre</th>

<th>Edad</th>

</tr>

<tr>

<td>Yeray</td>

<td>30</td>

</tr>

</table>

* **<form> con elementos avanzados**:

html

Copiar código

<form>

<label for="nombre">Nombre:</label>

<input type="text" id="nombre" name="nombre">

<label for="edad">Edad:</label>

<input type="number" id="edad" name="edad">

<label for="pais">País:</label>

<select id="pais" name="pais">

<option value="es">España</option>

<option value="mx">México</option>

</select>

<button type="submit">Enviar</button>

</form>

**2. Atributos HTML útiles:**

* **data-\***: Para almacenar datos personalizados en HTML.

html

Copiar código

<div data-id="1234" data-role="usuario">Contenido</div>

* **required, placeholder y maxlength** en formularios:

html

Copiar código

<input type="text" placeholder="Escribe aquí" required maxlength="20">

**JavaScript Avanzado**

**1. Métodos avanzados para Arrays:**

* **map()**: Aplica una función a cada elemento y devuelve un nuevo array.

javascript

Copiar código

let numeros = [1, 2, 3, 4];

let dobles = numeros.map(num => num \* 2);

console.log(dobles); // [2, 4, 6, 8]

* **filter()**: Filtra los elementos según una condición.

javascript

Copiar código

let mayoresDeDos = numeros.filter(num => num > 2);

console.log(mayoresDeDos); // [3, 4]

* **reduce()**: Reduce un array a un único valor.

javascript

Copiar código

let suma = numeros.reduce((acumulador, num) => acumulador + num, 0);

console.log(suma); // 10

**2. Eventos más comunes:**

* **mouseover y mouseout**: Para efectos al pasar el ratón.

javascript

Copiar código

let boton = document.getElementById("miBoton");

boton.addEventListener("mouseover", function() {

boton.style.backgroundColor = "blue";

});

boton.addEventListener("mouseout", function() {

boton.style.backgroundColor = "initial";

});

* **keydown y keyup**: Captura eventos de teclado.

javascript

Copiar código

document.addEventListener("keydown", function(event) {

console.log("Tecla presionada:", event.key);

});

**3. Manipulación avanzada del DOM:**

* **Crear y añadir elementos**:

javascript

Copiar código

let nuevoParrafo = document.createElement("p");

nuevoParrafo.textContent = "Este es un nuevo párrafo.";

document.body.appendChild(nuevoParrafo);

* **Modificar clases y estilos**:

javascript

Copiar código

let elemento = document.getElementById("miElemento");

elemento.classList.add("nueva-clase");

elemento.style.color = "red";

**4. Funciones Flecha y Callbacks:**

* **Funciones Flecha**:

javascript

Copiar código

let sumar = (a, b) => a + b;

console.log(sumar(5, 3)); // 8

* **Callbacks**: Pasar funciones como argumentos.

javascript

Copiar código

function procesarCallback(nombre, callback) {

console.log("Procesando...");

callback(nombre);

}

procesarCallback("Yeray", nombre => {

console.log("Hola, " + nombre);

});

**5. Programación Asíncrona: Promesas y Async/Await**

* **Promesas**:

javascript

Copiar código

let promesa = new Promise((resolve, reject) => {

let exito = true;

if (exito) resolve("Éxito");

else reject("Error");

});

promesa.then(result => console.log(result)).catch(error => console.log(error));

* **Async/Await**: Simplifica el uso de Promesas.

javascript

Copiar código

async function obtenerDatos() {

try {

let respuesta = await fetch("https://api.example.com/data");

let datos = await respuesta.json();

console.log(datos);

} catch (error) {

console.error("Error:", error);

}

}

obtenerDatos();

**6. JSON (JavaScript Object Notation):**

JSON se usa para enviar y recibir datos entre el cliente y el servidor. Aquí te muestro cómo convertir un objeto JavaScript a JSON y viceversa:

* **Convertir objeto a JSON**:

javascript

Copiar código

let objeto = { nombre: "Yeray", edad: 30 };

let json = JSON.stringify(objeto);

console.log(json); // {"nombre":"Yeray","edad":30}

* **Convertir JSON a objeto**:

javascript

Copiar código

let textoJson = '{"nombre":"Yeray","edad":30}';

let objetoConvertido = JSON.parse(textoJson);

console.log(objetoConvertido.nombre); // "Yeray"

**7. Desestructuración de Objetos y Arrays:**

Simplifica el acceso a valores en objetos y arrays.

* **Objetos**:

javascript

Copiar código

let persona = { nombre: "Yeray", edad: 30 };

let { nombre, edad } = persona;

console.log(nombre); // "Yeray"

* **Arrays**:

javascript

Copiar código

let numeros = [10, 20, 30];

let [primero, segundo] = numeros;

console.log(primero); // 10

Estos conceptos adicionales te ayudarán a trabajar más eficientemente con HTML y JavaScript, cubriendo estructuras avanzadas y funcionalidades útiles para la creación de aplicaciones web. ¿Te gustaría que prepare algún ejemplo de implementación completa?