

**UNIVERSIDAD DE LAS FUERZAS ARMADAS "ESPE"**  
**CIENCIAS DE LA COMPUTACIÓN**  
**PROGRAMACION INTEGRATIVA DE COMPONENTES WEB**

**Nombre:** Andrés Revelo

**NRC:** 16496

**Fecha:** 20/ Junio/2024

**Aplicaciones Web JS con POO**

- **Introducción**

En este manual, exploraremos el desarrollo de una aplicación web utilizando el paradigma de Programación Orientada a Objetos (OOP) en JavaScript. La Programación Orientada a Objetos es un paradigma de programación que utiliza "objetos" – estructuras de datos que contienen datos (campos) y métodos (funciones) – y sus interacciones para diseñar aplicaciones y programas informáticos. Este enfoque permite una mejor organización del código, facilitando su mantenimiento, reutilización y escalabilidad.

- **Estructura del Proyecto**

La estructura de directorios y archivos para el proyecto es la siguiente:

tarea2/

├─ index.html

├─ css/

| └─ styles.css

├─ js/

| └─ app.js

└─ assets/

└─ images/

| « Documentos » Universidad ESPE » Sexto Semestre » Integrativa » tarea2 » |      |                       |  | ▼ ↺                 | Buscar en t... 🔍 |
|---|------|-----------------------|--|---------------------|------------------|
| Nombre  |      | Fecha de modificación |  | Tipo                |                  |
| do  | css  | 20/06/2024 2:02       |  | Carpeta de archivos |                  |
|   | html | 20/06/2024 2:02       |  | Carpeta de archivos |                  |
|   | js   | 20/06/2024 2:02       |  | Carpeta de archivos |                  |
| os  |      |                       |  |                     |                  |

*Ilustración 1 Directorio de carpetas*

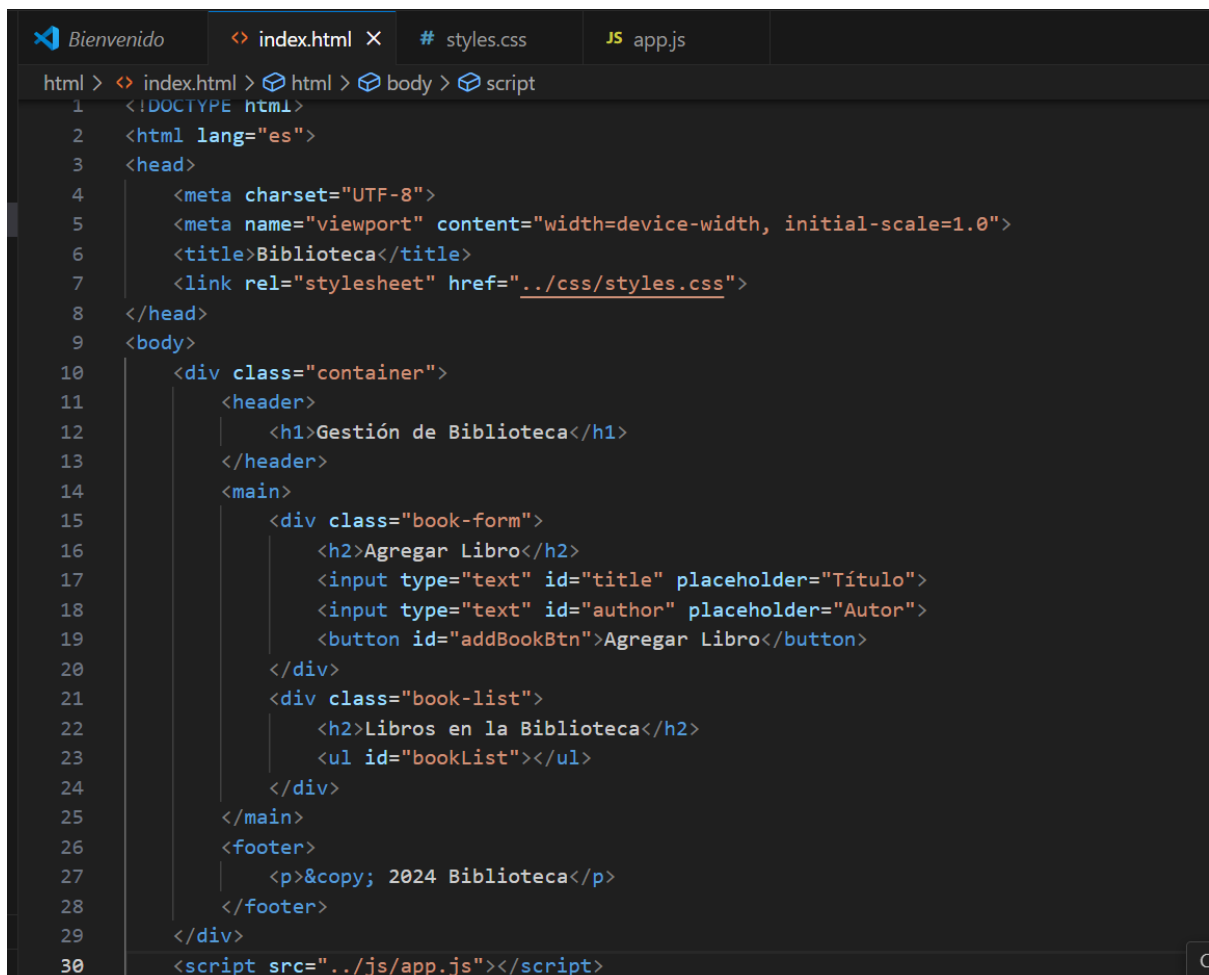
- Desarrollo

Para este ejercicio, propondré una aplicación web simple que gestione un sistema de biblioteca. La aplicación permitirá agregar, eliminar y mostrar libros en una biblioteca utilizando principios de OOP como clases, herencia, agregación y composición.

En las siguientes secciones, detallaremos cada paso del desarrollo de la aplicación, desde la estructura del proyecto hasta la implementación de las funcionalidades utilizando HTML, CSS y JavaScript. Al seguir este manual, podrás adquirir una comprensión práctica de cómo aplicar los principios de OOP en el desarrollo de aplicaciones web con JavaScript.

### -Código HTML

Desarrollamos nuestro código de HTML, esto para generar el aspecto visual de nuestra página, aquí definimos el contenido de igual forma. Este archivo define la estructura básica de la página, incluye el título, el encabezado, un botón interactivo y un pie de página. También enlaza los archivos CSS y JavaScript.

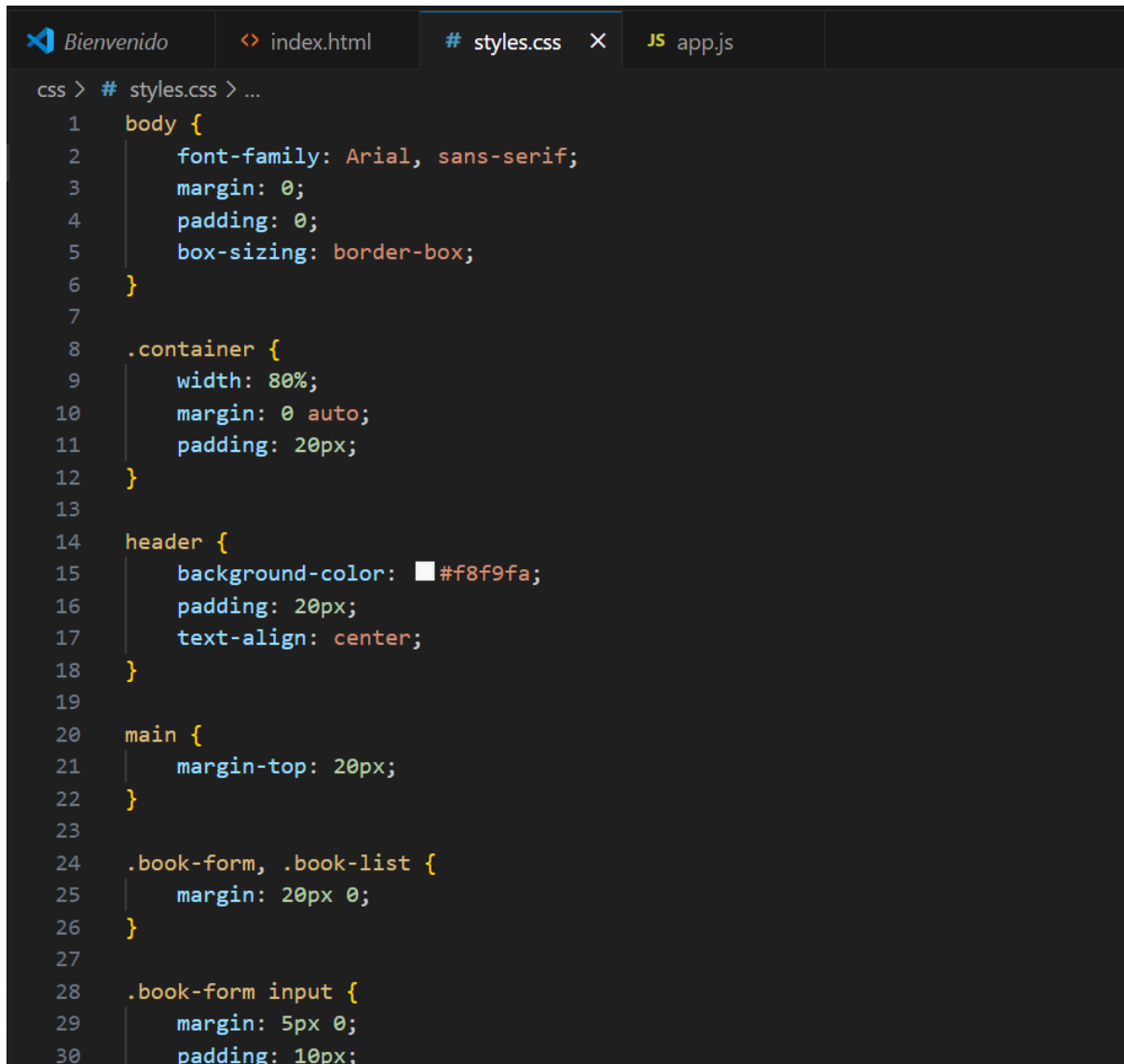
A screenshot of a code editor with a dark theme. The editor has four tabs at the top: 'Bienvenido', 'index.html X', '# styles.css', and 'JS app.js'. The 'index.html' tab is active, showing the following HTML code:

```
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Biblioteca</title>
7   <link rel="stylesheet" href="../css/styles.css">
8 </head>
9 <body>
10   <div class="container">
11     <header>
12       <h1>Gestión de Biblioteca</h1>
13     </header>
14     <main>
15       <div class="book-form">
16         <h2>Agregar Libro</h2>
17         <input type="text" id="title" placeholder="Título">
18         <input type="text" id="author" placeholder="Autor">
19         <button id="addBookBtn">Agregar Libro</button>
20       </div>
21       <div class="book-list">
22         <h2>Libros en la Biblioteca</h2>
23         <ul id="bookList"></ul>
24       </div>
25     </main>
26     <footer>
27       <p>&copy; 2024 Biblioteca</p>
28     </footer>
29   </div>
30   <script src="../js/app.js"></script>
```

Ilustración 2 Código HTML

## -Código CSS

Este archivo define estilos básicos para el cuerpo, encabezado, contenido principal y pie de página. Se utiliza CSS simple para mantener el diseño claro y accesible.

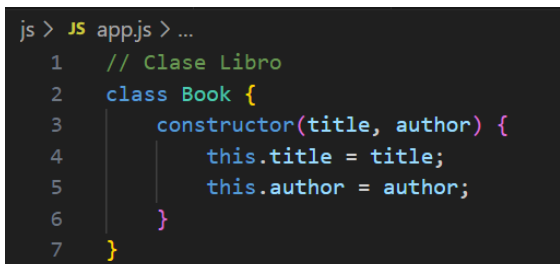
A screenshot of a code editor with a dark theme. The top bar shows four tabs: 'Bienvenido' (with a VS icon), 'index.html' (with a <> icon), '# styles.css' (selected, with a # icon), and 'JS app.js' (with a JS icon). The main editor area shows CSS code for 'styles.css'. The code is as follows:

```
css > # styles.css > ...
1  body {
2      font-family: Arial, sans-serif;
3      margin: 0;
4      padding: 0;
5      box-sizing: border-box;
6  }
7
8  .container {
9      width: 80%;
10     margin: 0 auto;
11     padding: 20px;
12 }
13
14 header {
15     background-color: #f8f9fa;
16     padding: 20px;
17     text-align: center;
18 }
19
20 main {
21     margin-top: 20px;
22 }
23
24 .book-form, .book-list {
25     margin: 20px 0;
26 }
27
28 .book-form input {
29     margin: 5px 0;
30     padding: 10px;
```

*Ilustración 3 Código CSS*

## -Interacción con JavaScript

Definimos nuestra clase libro, para poder agregar tanto el título como su autor

A screenshot of a code editor with a dark theme. The top bar shows two tabs: 'JS app.js' (selected, with a JS icon). The main editor area shows JavaScript code for 'app.js'. The code is as follows:

```
js > JS app.js > ...
1  // Clase Libro
2  class Book {
3      constructor(title, author) {
4          this.title = title;
5          this.author = author;
6      }
7  }
```

*Ilustración 4 Código JavaScript*

Definimos nuestra clase biblioteca con sus respectivos constructores

```
s > JS app.js > ...
9 // Clase Biblioteca
10 class Library {
11   constructor() {
12     this.books = [];
13   }
14
15   addBook(book) {
16     this.books.push(book);
17     this.displayBooks();
18   }
19
20   removeBook(title) {
21     this.books = this.books.filter(book => book.title !== title);
22     this.displayBooks();
23   }
24 }
```

*Ilustración 5 Clase Biblioteca*

Generamos nuestra función `displayBooks` esto para actualizar dinámicamente una lista de libros en una página web. Cada libro se muestra como un elemento de lista con su título y autor, y se proporciona un botón de eliminación para cada libro. Cuando se hace clic en el botón de eliminación, se llama a la función `this.removeBook(book.title)` para eliminar el libro correspondiente. Es una forma común de mostrar y manipular datos en una interfaz web.

```
25   displayBooks() {
26     const bookList = document.getElementById('bookList');
27     bookList.innerHTML = '';
28     this.books.forEach(book => {
29       const li = document.createElement('li');
30       li.textContent = `${book.title} by ${book.author}`;
31       const deleteBtn = document.createElement('button');
32       deleteBtn.textContent = 'Eliminar';
33       deleteBtn.addEventListener('click', () => {
34         this.removeBook(book.title);
35       });
36       li.appendChild(deleteBtn);
37       bookList.appendChild(li);
38     });
39   }
40 }
```

*Ilustración 6 displayBooks*

Por último generamos el evento para agregar libros

```
42 // Evento para agregar libros
43 document.addEventListener('DOMContentLoaded', () => {
44     const library = new Library();
45
46     document.getElementById('addBookBtn').addEventListener('click', () => {
47         const title = document.getElementById('title').value;
48         const author = document.getElementById('author').value;
49
50         if (title && author) {
51             const book = new Book(title, author);
52             library.addBook(book);
53             document.getElementById('title').value = '';
54             document.getElementById('author').value = '';
55         }
56     });
57 });
58
```

Ilustración 7 Evento agregar libros

-Resultado final

Agregamos un titulo con su autor

## Gestión de Biblioteca

### Agregar Libro

Agregar Libro

### Libros en la Biblioteca

Mariposa Traicionera by Julio Verne

Eliminar

© 2024 Biblioteca

Ilustración 8 Función agregar libro

Podemos ver que se agrego correctamente el libro

### Gestión de Biblioteca

#### Agregar Libro

Título

Autor

Agregar Libro

#### Libros en la Biblioteca

|                                     |          |
|-------------------------------------|----------|
| Mariposa Traicionera by Julio Verne | Eliminar |
| Reminiscencias by Julio Jaramillo   | Eliminar |

© 2024 Biblioteca

Ilustración 9 Libro Agregado

Eliminamos el libro agregado

### Gestión de Biblioteca

#### Agregar Libro

Título

Autor

Agregar Libro

#### Libros en la Biblioteca

|                                     |          |
|-------------------------------------|----------|
| Mariposa Traicionera by Julio Verne | Eliminar |
|-------------------------------------|----------|

© 2024 Biblioteca

Ilustración 10 Función eliminar

- **Conclusión**

Esta aplicación web básica demuestra la manipulación del DOM con JavaScript, el uso de CSS para el diseño y la estructura lógica del código mediante la separación por capas. Puedes expandir esta aplicación agregando más funcionalidad y mejorando el diseño.