



# ACTIVIDAD EVALUABLE

César Ríos

Contenido

Pruebas de depuración..... 2

Código ..... 4

# Pruebas de depuración

## Página principal

gestor de biblioteca

C:\Users\CampusFP\Desktop\Actividades\AE\_CRP\_DWEC\index.html

Biblioteca de César

Agregar Libro

Título

Autor

Año

Género

ISBN

Añadir

Buscar Libro

Título o ISBN

Buscar

Eliminar Libro

ISBN

Eliminar

Listado de Libros

Listar

Título	Autor	Año	Género	ISBN
— Sin resultados —				

Estadísticas

Mostrar estadísticas

## Agregamos un libro y vemos que nos sale en la tabla

gestor de biblioteca

C:\Users\CampusFP\Desktop\Actividades\AE\_CRP\_DWEC\index.html

Biblioteca de César

Agregar Libro

Título

Autor

Año

Género

ISBN

Añadir

Buscar Libro

Título o ISBN

Buscar

Eliminar Libro

ISBN

Eliminar

Listado de Libros

Listar

Título	Autor	Año	Género	ISBN
La prueba depurada	Cesar Rios	2025	Ficcion	226262626262

Estadísticas

Mostrar estadísticas

Libro añadido

## Prueba ISBN

## Agregamos otro libro

Biblioteca de César

Agregar Libro

Prueba ISBN

Cesar Rios

2005

Drama

2623626262672

Añadir

Listado de Libros

Listar

Título	Autor	Año	Género	ISBN
La prueba depurada	Cesar Rios	2025	Ficcion	226262626262
Prueba ISBN	Cesar Rios	2005	Drama	2623626262672

Al poner su ISBN nos sale el libro “Prueba ISBN”

**Buscar Libro**

**Eliminar Libro**

**Listado de Libros**  

Título	Autor	Año	Género	ISBN
Prueba ISBN	Cesar Rios	2005	Drama	2623626262672

Prueba para eliminar

**Eliminar Libro**

**Listado de Libros**  

Título	Autor	Año	Género	ISBN
Prueba ISBN	Cesar Rios	2005	Drama	2623626262672

Como vemos se ha eliminado correctamente y al final de la página vemos “Libro Eliminado”

**Buscar Libro**

**Eliminar Libro**

**Listado de Libros**  

Título	Autor	Año	Género	ISBN
— Sin resultados —				

**Estadísticas**

Por último, mostramos las estadísticas de los libros que en este caso hay uno

**Listado de Libros**  

Título	Autor	Año	Género	ISBN
La prueba depurada	Cesar Rios	2025	Ficcion	226262626262

**Estadísticas**  

Total: 1  
Por género: {"Ficcion":1}  
Media antigüedad: 0.00 años

# Código

Index.html

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="utf-8" />
  <title>Gestor de Biblioteca</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <h2>Biblioteca de César</h2>

  <section class="card">
    <h3>Agregar Libro</h3>
    <input id="titulo" placeholder="Título">
    <input id="autor" placeholder="Autor">
    <input id="anio" type="number" placeholder="Año">
    <input id="genero" placeholder="Género">
    <input id="isbn" placeholder="ISBN">
    <button id="btnAgregar">Añadir</button>
  </section>

  <section class="card">
    <h3>Buscar Libro</h3>
    <input id="buscar" placeholder="Título o ISBN">
    <button id="btnBuscar">Buscar</button>
  </section>

  <section class="card">
    <h3>Eliminar Libro</h3>
    <input id="eliminar" placeholder="ISBN">
    <button id="btnEliminar">Eliminar</button>
  </section>

  <section class="card">
    <h3>Listado de Libros</h3>
    <button id="btnListar">Listar</button>
    <table id="tablaLibros" border="1"></table>
  </section>

  <section class="card">
    <h3>Estadísticas</h3>
    <button id="btnEstadisticas">Mostrar estadísticas</button>
    <pre id="stats"></pre>
  </section>

  <p id="mensaje" class="mensaje"></p>
```

```
<script defer src="app.js"></script>
</body>
</html>
```

## Styles.css

```
body {
  font-family: Arial;
  margin: 20px;
}

input {
  margin: 4px;
  padding: 5px;
}

button {
  margin: 4px;
  padding: 6px 10px;
  cursor: pointer;
}

table {
  margin-top: 10px;
  width: 100%;
  border-collapse: collapse;
}

th, td {
  padding: 5px;
}

.mensaje {
  font-weight: bold;
  margin-top: 10px;
}
```

## App.js

```
class Libro {
  constructor(titulo, autor, anioPublicacion, genero, ISBN) {
    this.titulo = titulo;
    this.autor = autor;
    this.anioPublicacion = anioPublicacion;
    this.genero = genero;
    this.ISBN = ISBN;
  }
}

class Biblioteca {
  constructor() {
    if (Biblioteca._instance) return Biblioteca._instance;
    this.libros = [];
    Biblioteca._instance = this;
  }

  agregarLibro(libro) {
    this.libros.push(libro);
    this.mostrarMensaje(" Libro añadido");
  }

  buscarLibro(termino) {
    if (!termino) return [];
    const t = termino.toLowerCase();
    return this.libros.filter(
      l => l.titulo.toLowerCase().includes(t) || l.ISBN === termino
    );
  }

  listarLibros() {
    return this.libros;
  }

  eliminarLibro(isbn) {
    const antes = this.libros.length;
    this.libros = this.libros.filter(l => l.ISBN !== isbn);
    return this.libros.length < antes;
  }

  obtenerEstadisticas() {
    const total = this.libros.length;
    const porGenero = this.libros.reduce((acc, l) => {
      acc[l.genero] = (acc[l.genero] || 0) + 1;
      return acc;
    }, {});
    const mediaAntig =
```

```

        total === 0
        ? 0
        : this.libros
            .map(l => new Date().getFullYear() - Number(l.anioPublicacion
|| 0))
            .reduce((a, b) => a + b, 0) / total;
        return { total, porGenero, mediaAntig };
    }

    mostrarMensaje(msg) {
        const el = document.getElementById("mensaje");
        if (el) el.textContent = msg;
    }
}

const biblioteca = new Biblioteca();

document.addEventListener("DOMContentLoaded", () => {
    const tituloInput = document.getElementById("titulo");
    const autorInput = document.getElementById("autor");
    const anioInput = document.getElementById("anio");
    const generoInput = document.getElementById("genero");
    const isbnInput = document.getElementById("isbn");
    const btnAgregar = document.getElementById("btnAgregar");

    const buscarInput = document.getElementById("buscar");
    const btnBuscar = document.getElementById("btnBuscar");

    const eliminarInput = document.getElementById("eliminar");
    const btnEliminar = document.getElementById("btnEliminar");

    const btnListar = document.getElementById("btnListar");
    const tablaLibros = document.getElementById("tablaLibros");

    const btnEstadisticas = document.getElementById("btnEstadisticas");
    const statsElement = document.getElementById("stats");

    function mostrarTabla(libros) {
        tablaLibros.innerHTML = `
            <tr>
                <th>Título</th><th>Autor</th><th>Año</th><th>Género</th><th>ISBN<
/ th>
            </tr>
        `;
        if (!libros || libros.length === 0) {
            tablaLibros.innerHTML += `<tr><td colspan="5" style="text-align:center">— Sin resultados —</td></tr>`;
            return;
        }
    }

```



```

    }
    const filas = libros.map(l => {
        return `<tr>
            <td>${escapeHtml(l.titulo)}</td>
            <td>${escapeHtml(l.autor)}</td>
            <td>${escapeHtml(l.anioPublicacion)}</td>
            <td>${escapeHtml(l.genero)}</td>
            <td>${escapeHtml(l.ISBN)}</td>
        </tr>`;
    }).join("");
    tablaLibros.innerHTML += filas;
}

function escapeHtml(str) {
    if (str === null || str === undefined) return "";
    return String(str)
        .replaceAll("&", "&amp;")
        .replaceAll("<", "&lt;")
        .replaceAll(">", "&gt;")
        .replaceAll("'", "&quot;")
        .replaceAll('"', "&#039;");
}

btnAgregar.addEventListener("click", () => {
    try {
        const titulo = tituloInput.value.trim();
        const autor = autorInput.value.trim();
        const anio = parseInt(anioInput.value, 10);
        const genero = generoInput.value.trim();
        const isbn = isbnInput.value.trim();

        if (!titulo || !autor || !anio || !genero || !isbn) {
            return biblioteca.mostrarMensaje(" Completa todos los campos");
        }

        biblioteca.agregarLibro(new Libro(titulo, autor, anio, genero,
isbn));
        tituloInput.value = autorInput.value = generoInput.value =
isbnInput.value = "";
        anioInput.value = "";
        mostrarTabla(biblioteca.listarLibros());
        console.log("Libro añadido:", { titulo, autor, anio, genero, isbn
});
    } catch (err) {
        console.error("Error en agregar:", err);
        biblioteca.mostrarMensaje(" Error al añadir (ver consola)");
    }
});

```

```

btnBuscar.addEventListener("click", () => {
  try {
    const termino = buscarInput.value.trim();
    const resultados = biblioteca.buscarLibro(termino);
    mostrarTabla(resultados);
    console.log("Buscar:", termino, " => ", resultados);
  } catch (err) {
    console.error("Error en buscar:", err);
  }
});

btnEliminar.addEventListener("click", () => {
  try {
    const isbn = eliminarInput.value.trim();
    if (!isbn) return biblioteca.mostrarMensaje(" Introduce un ISBN a
eliminar");
    const ok = biblioteca.eliminarLibro(isbn);
    biblioteca.mostrarMensaje(ok ? " Libro eliminado" : " ISBN no
encontrado");
    mostrarTabla(biblioteca.listarLibros());
    console.log("Eliminar ISBN:", isbn, "=>", ok);
  } catch (err) {
    console.error("Error en eliminar:", err);
  }
});

btnListar.addEventListener("click", () => {
  try {
    mostrarTabla(biblioteca.listarLibros());
    console.log("Listar:", biblioteca.listarLibros());
  } catch (err) {
    console.error("Error en listar:", err);
  }
});

btnEstadisticas.addEventListener("click", () => {
  try {
    const stats = biblioteca.obtenerEstadisticas();
    statsElement.textContent = ` Total: ${stats.total}\n Por género:
${JSON.stringify(stats.porGenero)}\n Media antigüedad:
${stats.mediaAntig.toFixed(2)} años`;
    console.log("Estadísticas:", stats);
  } catch (err) {
    console.error("Error en estadísticas:", err);
  }
});

mostrarTabla([]);
});

```

