

WEEK- 7

Aim: Modular JavaScript and Working with JSON (Online Bookstore)

Tasks:

1. **Create a JSON data file or object** containing a list of books with the following properties:
 - title
 - author
 - price
 - availability (in stock or out of stock)
2. **Structure your code using JavaScript modules:**
 - A module for displaying book listings.
 - A module for managing the shopping cart (add, remove, calculate total).
 - A module for updating the UI dynamically.
3. **Use ES6 module syntax** (import / export) to separate and reuse functionality across scripts.
4. **Implement interactivity using event listeners:**
 - Add event listeners for "Add to Cart" buttons.
 - Handle removal of items from the cart.
 - Dynamically update the UI when the cart is modified.
5. **Display the cart contents and total price:**
 - Show a summary of books in the cart.
 - Allow users to remove items or proceed to a mock checkout view.

Project Structure:

Week 7/

|-----index.html

|--- data.js

|---books.js

|---cart.js

|-----ui.js

index.html

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <title>Online Bookstore</title>
```

```
  <style>
```

```
    body {
```

```
      font-family: Arial, sans-serif;
```

```
      background: #f4f4f4;
```

```
      margin: 0;
```

```
      padding: 20px;
```

```
    }
```

```
    .book-list, .cart {
```

```
      max-width: 600px;
```

```
      margin: auto;
```

```
background: white;

padding: 20px;

border-radius: 12px;

box-shadow: 0 4px 8px rgba(0,0,0,0.1);

margin-bottom: 20px;
}

.book {

display: flex;

justify-content: space-between;

padding: 10px 0;

border-bottom: 1px solid #ddd;
}

.book:last-child {

border-bottom: none;
}

button {

background: #007BFF;

color: white;

border: none;

border-radius: 6px;

padding: 6px 10px;

cursor: pointer;
}
```

```
button:hover {  
    background: #0056b3;  
}  
  
.cart-item {  
    display: flex;  
    justify-content: space-between;  
    margin: 5px 0;  
}
```

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

```
</style>
```

```
</head>
```

```
<body>
```

```
<h1 style="text-align:center;">📖 Online Bookstore</h1>
```

```
<div class="book-list" id="bookList"></div>
```

```
<div class="cart">
```

```
<h2>🛒 Shopping Cart</h2>
```

```
<div id="cartItems"></div>
```

```
<div class="cart-total" id="cartTotal">Total: ₹0</div>
```

```
</div>
```

```
<script type="module">
  import { displayBooks } from './books.js';
  import { updateCartUI } from './ui.js';
  import { cart } from './cart.js';
  // Display books on page load
  displayBooks();
  // Initialize empty cart display
  updateCartUI(cart);
</script>
</body>
</html>
```

data.js (JSON Data)

```
export const books = [
  { title: "JavaScript Essentials", author: "John Doe", price: 499, availability: "in stock"
  },
  { title: "Learn React", author: "Jane Smith", price: 699, availability: "in stock" },
  { title: "Node.js in Action", author: "Mark Lee", price: 599, availability: "out of
  stock" },
  { title: "Python for Everyone", author: "Ramya Krishna", price: 450, availability: "in
  stock" }
];
```

cart.js (Cart Logic)

```
export let cart = [];  
  
// Add book to cart  
export function addToCart(book) {  
  cart.push(book);  
}  
  
// Remove book by index  
export function removeFromCart(index) {  
  cart.splice(index, 1);  
}  
  
// Calculate total price  
export function calculateTotal() {  
  return cart.reduce((sum, book) => sum + book.price, 0);  
}
```

books.js (Display Book Listings)

```
import { books } from './data.js';  
import { addToCart, cart } from './cart.js';  
import { updateCartUI } from './ui.js';  
export function displayBooks() {  
  const bookList = document.getElementById("bookList");  
  bookList.innerHTML = ""; // Clear before rendering
```

```

books.forEach((book, index) => {
  const bookDiv = document.createElement("div");
  bookDiv.classList.add("book");
  bookDiv.innerHTML = `
    <div>
      <strong>${book.title}</strong> <br>
      <small>by ${book.author}</small> <br>
      <span>₹${book.price}</span>
    </div>
    <div>
      ${book.availability === "in stock"
        ? `<button data-index="${index}">Add to Cart</button>`
        : `<span style="color:red;">Out of Stock</span>`}
    </div>
  `;
  bookList.appendChild(bookDiv);
});

// Attach event listeners to "Add to Cart" buttons
bookList.querySelectorAll("button").forEach(btn => {
  btn.addEventListener("click", (e) => {
    const idx = e.target.getAttribute("data-index");
    addToCart(books[idx]);
  });
});

```

```
    updateCartUI(cart);  
  });  
});  
}
```

ui.js (UI Updates)

```
import { calculateTotal, removeFromCart, cart } from './cart.js';  
export function updateCartUI(cartData) {  
  const cartItemsDiv = document.getElementById("cartItems");  
  const cartTotalDiv = document.getElementById("cartTotal");  
  cartItemsDiv.innerHTML = "";  
  cartData.forEach((book, index) => {  
    const itemDiv = document.createElement("div");  
    itemDiv.classList.add("cart-item");  
    itemDiv.innerHTML = `  
      <span>${book.title} - ₹${book.price}</span>  
      <button data-index="${index}" style="background:red;">Remove</button>  
    `;  
    cartItemsDiv.appendChild(itemDiv);  
  });  
  // Attach remove event listeners  
  cartItemsDiv.querySelectorAll("button").forEach(btn => {  
    btn.addEventListener("click", (e) => {
```



```

const idx = e.target.getAttribute("data-index");

removeFromCart(idx);

updateCartUI(cart);

});

});

// Update total

cartTotalDiv.textContent = `Total: ₹${calculateTotal()}`;
}

```

Here,

- **JSON Object** (data.js) → stores book info.
- **ES6 Modules** → import & export used to separate logic.
- **Dynamic DOM updates** → Books displayed dynamically, cart updates live.
- **Event Listeners** → Add & remove books interactively.
- **Reusable Code** → UI updates & cart logic are modular.

Run index.html on Live Server — that's your **main entry point**

- It loads all other modules (data.js, books.js, cart.js, ui.js).
- It displays the book list and cart.
- It imports the JS modules using type="module".

Output:

