Book Inventory Management project

**Overview**

The Book Inventory Management System is a web-based application designed to help users manage their book collection. Users can add, edit, filter, delete and export inventory data easily.

**System Features**

1. **Home Page**:
   * Provides quick navigation links to various features such as adding books, viewing inventory, and exporting data.
2. **Add Books**:
   * A form to input book details like title, author, genre, publication date, and ISBN.
   * Data is validated and stored in the database.
3. **View Inventory**:
   * Displays a table of all books in the inventory.
   * Includes filtering options based on title, author, genre, or publication date.
4. **Edit and Delete**:
   * Allows users to modify book details or remove books from the inventory.
5. **Export Data**:
   * Provides options to export the book inventory in CSV or JSON format.
6. **Navigation**:
   * A back button is provided on most pages to navigate back to the home page.

**Basic Operations**

1. Add Books:
   * Navigate to the "Add a New Book" link on the home page. Fill out the form and submit it.
2. View Inventory:
   * Click "View Inventory" to see all books. Use the filter options to refine your search
3. Edit and Delete:
   * In the inventory table, click "Edit" to modify details or "Delete" to remove a book.
4. Export Data:
   * Click "Export Data" on the home page to download the inventory in CSV or JSON format.

**Design Decisions**

1. Header and Footer Consistency:
   * A centralized header.php and footer.php ensure consistency across all pages.
   * This minimizes duplication and simplifies maintenance.
2. Filtering Functionality:
   * Filters are applied via GET requests, making the URL bookmarkable for specific searches.
3. Export Options:
   * Both CSV and JSON formats were implemented to accommodate user preferences for structured data.

**Challenges Faced**

1. Footer Positioning:
   * Initial implementation caused the footer to overlap content.
   * Solution: Adjusted CSS with a sticky footer design and min-height for the main content.
2. Form Validation:
   * Ensuring only valid data is submitted required both client-side (JavaScript) and server-side (PHP) validation.
3. Export Functionality:
   * Ensuring proper file download headers and format compatibility was tricky, especially with JSON.