Exp:08	Consider a Library Management System. Develop a JavaScript program that will validate the controls in the forms you have created for the application. Note: Your application must access a database using Servlet/JSP.

Aim

To design and implement a Library Management System using HTML, JSP, Servlet, JavaScript, and MySQL, where users can add, view, and delete books. The system includes form validation with JavaScript for user input, and communicates with a backend database using Servlets and JSP, offering a user-friendly, responsive interface.

Algorithm

1. **Start**
   - The user opens the home page (index.html) in a web browser.
2. **Display Options**
   - The home page displays three main buttons:
     - **Add Book**
     - **View Books**
     - **Delete Book**
   - Each button redirects to a specific HTML form or JSP page for the respective action.
3. **Add Book Form**
   - Loads bookform.html with the following fields:
     - Book ID
     - Book Name
     - Author
     - Price
   - Includes a submit button to send data to savebook.jsp.
4. **Apply JavaScript Validation**
   - On form submission, JavaScript validates the input:
     - Book ID must not be empty and must be unique.
     - Name and Author fields should contain only letters.
     - Price should be a valid decimal number.
   - If validation is successful, the form is submitted to the server.
5. **Save Data Using Servlet**
   - SaveBookServlet receives the data via JSP.
   - The servlet connects to the MySQL database.
   - It inserts the book record into the books table.
6. **View Books Page**
   - Loads viewbooks.jsp.
   - The JSP script fetches all book records from the database.
   - It displays the data in a structured and styled HTML table.
7. **Delete Book Page**
   - Loads deletebook.html.
   - The user is prompted to enter the Book ID they wish to delete.
   - The form sends the request to DeleteBookServlet.
   - The servlet deletes the corresponding record from the database.
   - A success or error message is shown based on the operation outcome.
8. **Apply CSS Styling**
   - Inline or internal CSS is used to enhance the user interface:
     - Input fields and buttons are styled for readability.
     - Forms are centered with background colors and shadows.

– Hover effects are added to interactive elements.
– Tables are styled for better visibility on the view page.

9. **End**

• The user can continue to add, view, or delete books using the menu.

CODE:

Index.html:

```
<!DOCTYPE html>
<html>
<head>
  <title>Library Home</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background: #f5f5f5;
      display: flex;
      flex-direction: column;
      align-items: center;
      margin-top: 80px;
    }
    h1 {
      color: #333;
    }
    .button-group {
      display: flex;
      gap: 20px;
      margin-top: 30px;
    }
    a button {
      padding: 15px 30px;
      font-size: 16px;
      background: #007BFF;
      color: white;
      border: none;
      border-radius: 8px;
      cursor: pointer;
      transition: background 0.3s;
    }
    a button:hover {
      background: #0056b3;
```

```
        }
    </style>
</head>
<body>
    <h1>Library Book Management</h1>
    <div class="button-group">
        <a href="bookform.html"><button>Add Book</button></a>
        <a href="ViewBookServlet"><button>View Books</button></a>
        <a href="deletebook.html"><button>Delete Book</button></a>
    </div>
</body>
</html>
```

Bookform.html:

```
<!DOCTYPE html>
<html>
<head>
    <title>Add Book</title>
    <style>
        body {
            font-family: Arial;
            background: #e9ecef;
            display: flex;
            flex-direction: column;
            align-items: center;
            margin-top: 60px;
        }
        form {
            background: #fff;
            padding: 30px;
            border-radius: 10px;
            box-shadow: 0 0 10px #ccc;
        }
        input {
            display: block;
            margin-bottom: 15px;
            width: 250px;
            padding: 10px;
        }
        input[type="submit"] {
            background: #28a745;
```

```
            color: white;
            border: none;
            cursor: pointer;
        }
        input[type="submit"]:hover {
            background: #218838;
        }
    </style>
</head>
<body>
    <h2>Add a New Book</h2>
    <form action="savebook.jsp" method="post">
        <input type="text" name="book_id" placeholder="Book ID" required />
        <input type="text" name="name" placeholder="Book Name" required />
        <input type="text" name="author" placeholder="Author" required />
        <input type="text" name="price" placeholder="Price" required />
        <input type="submit" value="Save Book" />
    </form>
</body>
</html>
```

Deletebooks.html:

```
<!DOCTYPE html>
<html>
<head>
    <title>Delete Book</title>
    <style>
        body {
            font-family: Arial;
            background: #f8f9fa;
            display: flex;
            flex-direction: column;
            align-items: center;
            margin-top: 60px;
        }
        form {
            background: #fff;
            padding: 25px;
            border-radius: 8px;
            box-shadow: 0 0 8px #ccc;
        }
```

```
        input {
            padding: 10px;
            margin-bottom: 15px;
            width: 240px;
        }
        input[type="submit"] {
            background: #dc3545;
            color: white;
            border: none;
        }
        input[type="submit"]:hover {
            background: #c82333;
        }
    </style>
</head>
<body>
    <h2>Delete Book</h2>
    <form action="DeleteBookServlet" method="post">
        <input type="text" name="book_id" placeholder="Enter Book ID to delete" required />
        <input type="submit" value="Delete Book" />
    </form>
</body>
</html>
```

SaveBookServlet.java:

```
import java.io.*;
import java.sql.*;
import jakarta.servlet.*;
import jakarta.servlet.http.*;

public class SaveBookServlet extends HttpServlet {
    public void doPost(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {

        String book_id = request.getParameter("book_id");
        String name = request.getParameter("name");
        String author = request.getParameter("author");
        String price = request.getParameter("price");

        try {
```

```
            Class.forName("com.mysql.cj.jdbc.Driver");
            Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/librarydb", "root",
"Guna@2005");

            PreparedStatement ps = con.prepareStatement("INSERT INTO books VALUES (?, ?,
?, ?)");
            ps.setString(1, book_id);
            ps.setString(2, name);
            ps.setString(3, author);
            ps.setString(4, price);

            ps.executeUpdate();

            response.setContentType("text/html");
            PrintWriter out = response.getWriter();
            out.println("<h3 style='color:green;'>Book added successfully!</h3>");
            out.println("<a href='index.html'>Back to Home</a>");
            con.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

ViewBookServlet.java:

```
import java.io.*;
import java.sql.*;
import jakarta.servlet.*;
import jakarta.servlet.http.*;

public class ViewBookServlet extends HttpServlet {
    public void doGet(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {

        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/librarydb", "root",
```

```
"Guna@2005");
            Statement stmt = con.createStatement();
            ResultSet rs = stmt.executeQuery("SELECT * FROM books");

            out.println("<h2>Books List</h2>");
            out.println("<table border='1' cellpadding='10'>");
            out.println("<tr><th>ID</th><th>Name</th><th>Author</th><th>Price</th></tr>");

            while (rs.next()) {
               out.println("<tr><td>" + rs.getString("book_id") + "</td><td>" +
                     rs.getString("name") + "</td><td>" +
                     rs.getString("author") + "</td><td>" +
                     rs.getString("price") + "</td></tr>");
            }
            out.println("</table>");
            out.println("<br><a href='index.html'>Back to Home</a>");
            con.close();
         } catch (Exception e) {
            e.printStackTrace();
         }
      }
}
```

DeleteBookServlet.java:

```
import java.io.*;
import java.sql.*;
import jakarta.servlet.*;
import jakarta.servlet.http.*;

public class ViewBookServlet extends HttpServlet {
   public void doGet(HttpServletRequest request, HttpServletResponse response)
         throws ServletException, IOException {

      response.setContentType("text/html");
      PrintWriter out = response.getWriter();

      try {
         Class.forName("com.mysql.cj.jdbc.Driver");
         Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/librarydb", "root",
"Guna@2005");
```

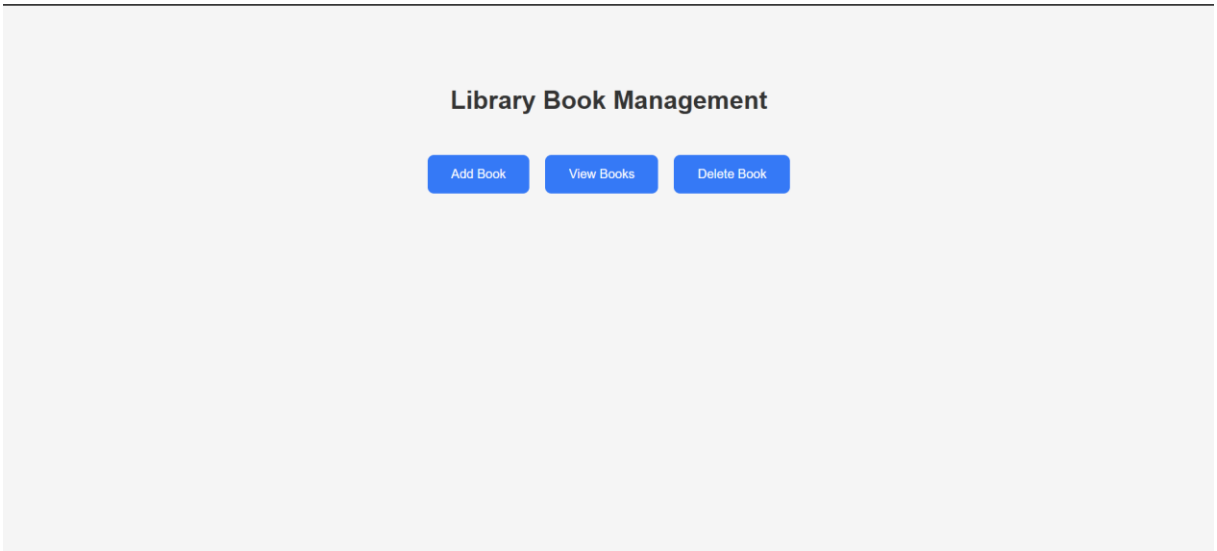
```
            Statement stmt = con.createStatement();
            ResultSet rs = stmt.executeQuery("SELECT * FROM books");

            out.println("<h2>Books List</h2>");
            out.println("<table border='1' cellpadding='10'>");
            out.println("<tr><th>ID</th><th>Name</th><th>Author</th><th>Price</th></tr>");

            while (rs.next()) {
                out.println("<tr><td>" + rs.getString("book_id") + "</td><td>" +
                        rs.getString("name") + "</td><td>" +
                        rs.getString("author") + "</td><td>" +
                        rs.getString("price") + "</td></tr>");
            }
            out.println("</table>");
            out.println("<br><a href='index.html'>Back to Home</a>");
            con.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

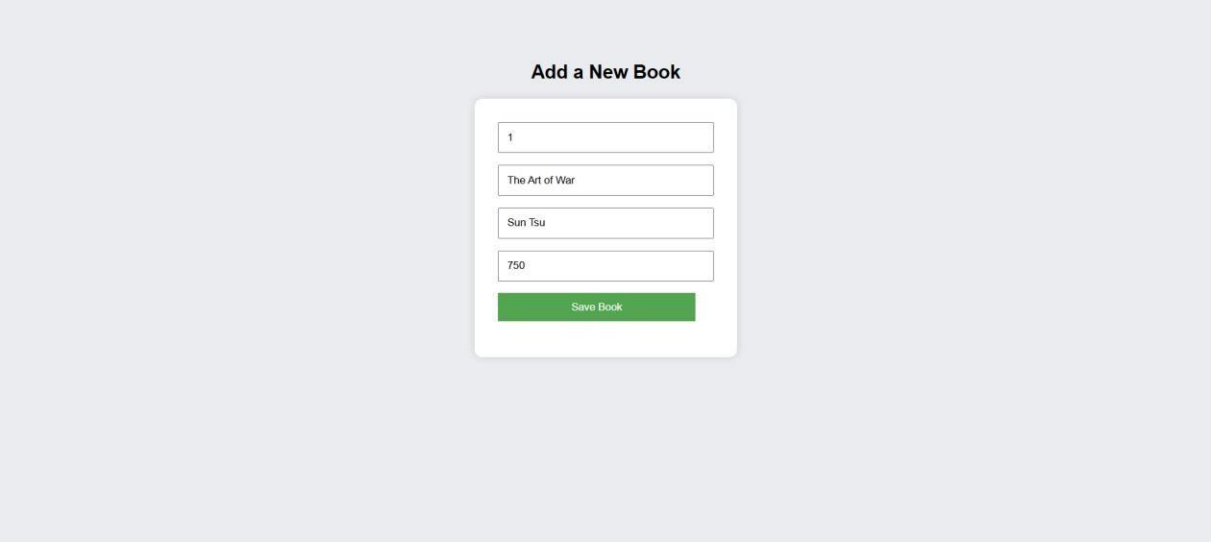
Web.xml:

```
<web-app xmlns="http://xmlns.jcp.org/xml/ns/javaee" version="3.1">
    <servlet>
        <servlet-name>SaveBookServlet</servlet-name>
        <servlet-class>SaveBookServlet</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>SaveBookServlet</servlet-name>
        <url-pattern>/SaveBookServlet</url-pattern>
    </servlet-mapping>

    <servlet>
        <servlet-name>ViewBookServlet</servlet-name>
        <servlet-class>ViewBookServlet</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>ViewBookServlet</servlet-name>
        <url-pattern>/ViewBookServlet</url-pattern>
```

```
    </servlet-mapping>

    <servlet>
        <servlet-name>DeleteBookServlet</servlet-name>
        <servlet-class>DeleteBookServlet</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>DeleteBookServlet</servlet-name>
        <url-pattern>/DeleteBookServlet</url-pattern>
    </servlet-mapping>
</web-app>
```

OUTPUT :

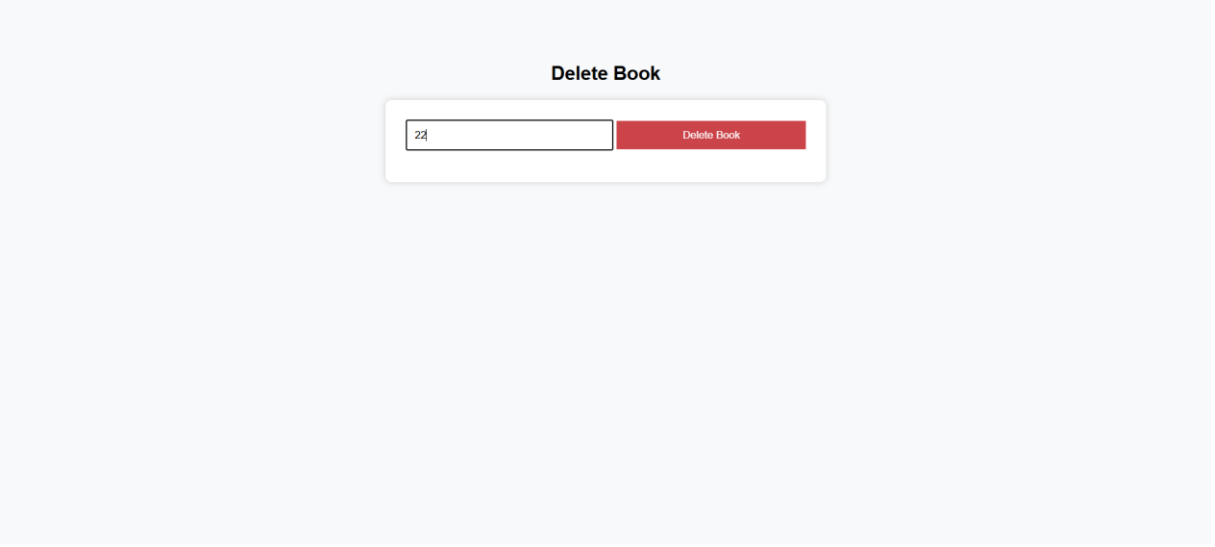




Books List

ID	Name	Author	Price
1	The Art of War	Sun Tsu	750.00
22	Sherlock Holmes	Unknown	1000.00
32	Harry Potter	JK Rowling	500.00

Back to Home



Books List

ID	Name	Author	Price
1	The Art of War	Sun Tsu	750.00
32	Harry Potter	JK Rowling	500.00

Back to Home

RESULT :

Thus the give program is executed successful