

Project Book Store : Part Books

1. หน้าจอ

➤ รายการข้อมูลในตาราง

[Search Book By Title/](#) [Search Book By Name/](#) [Add new book/](#) [List of books](#)

Books List				
Book ISBN	Title	Publisher ID	Category ID	Price
0-02188448-8	HTML for the World Wide Web: Visual Quick Start Guide	00218	1	17.95
0-07-882198-3	Beyond HTML	00788	32	27.95
0-07-912100-4	HTML Publishing on the Internet	00791	32	45.00
0-07-912273-6	Webmaster's Guide to HTML	00791	32	34.95
0-07-970411-0	HTML Visual Quick Reference	07897	32	15.00
0-12623540-6	Advanced HTML & CGI Writer's Companion	01262	32	29.95
0-13-232331-1	The Invention of Religion	01356	1	40.00
0-13-242488-6	HTML For Fun and Profit: Gold Signature Edition	01356	1	40.00
0-13460296-X	The Web Page Recipe Book	01356	32	29.95
0-18-83577578	Netscape and HTML EXplorer	18835	32	39.99
0-47-114242-5	HTML Sourcebook, 2nd Ed.	04711	32	30.00
0-53451626-2	HTML Plus!	05345	32	36.00
0-78-970622-9	Using HTML	07897	32	24.99

➤ ฟอรมค้นหาจาก ID

[Search Book By Title/](#) [Search Book By Name/](#) [Add new book/](#) [List of books](#)

Books Search Form By Title

Enter Book Title :

➤ ผลการค้นหามาจาก ID

[Search Book By Title/](#) [Search Book By Name/](#) [Add new book/](#) [List of books](#)

Books Search Form By Title

Enter Book Title :

Books ISBN	0-78970812-4
Books Title	Green Illusions
Books Price	34.99

➤ ฟอรมค้นหาจากข้อความ

[Search Book By Title/](#) [Search Book By Name/](#) [Add new book/](#) [List of books](#)

Books Search Form By Book Name

Search by title:

➤ ผลการค้นหาจากข้อความ

[Search Book By Title/](#) [Search Book By Name/](#) [Add new book/](#) [List of books](#)

Books Search Form By Book Name

Search by title:

Book ISBN	Title	Publisher ID	Category ID	Price
0-78-970622-9	Using HTML	07897	32	24.99
0-78-970758-6	Using HTML Special Edition	07897	32	49.99

➤ ฟอรมเพิ่มข้อมูล

[Search Book By Title/](#) [Search Book By Name/](#) [Add new book/](#) [List of books](#)

Add New Book

Title:
 ISBN:
 Publisher Id :
 Category Id :
 Price :

➤ ผลการเพิ่มข้อมูล

[Search Book By Title/](#) [Search Book By Name/](#) [Add new book/](#) [List of books](#)

Add New Book

Saved Data Successful

Title:
 ISBN:
 Publisher Id :
 Category Id :
 Price :

2. โครงสร้างตาราง และ ตัวอย่างข้อมูลในตาราง

- โครงสร้างตาราง

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	Title	varchar(80)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext Distinct values
2	ISBN	char(13)	latin1_swedish_ci		No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values
3	Publisher_Id	char(5)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext Distinct values
4	CategoryID	varchar(5)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext Distinct values
5	Price	decimal(10,2)			Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext Distinct values

- ตัวอย่างข้อมูลในตาราง

				Title	ISBN	Publisher_Id	CategoryID	Price
<input type="checkbox"/>				HTML for the World Wide Web: Visual Quick Start Gu...	0-02188448-8	00218	1	17.95
<input type="checkbox"/>				Beyond HTML	0-07-882198-3	00788	32	27.95
<input type="checkbox"/>				HTML Publishing on the Internet	0-07-912100-4	00791	32	45.00
<input type="checkbox"/>				Webmaster's Guide to HTML	0-07-912273-6	00791	32	34.95
<input type="checkbox"/>				HTML Visual Quick Reference	0-07-970411-0	07897	32	15.00
<input type="checkbox"/>				Advanced HTML & CGI Writer's Companion	0-12623540-6	01262	32	29.95
<input type="checkbox"/>				The Invention of Religion	0-13-232331-1	01356	1	40.00
<input type="checkbox"/>				HTML For Fun and Profit: Gold Signature Edition	0-13-242488-6	01356	1	40.00
<input type="checkbox"/>				The Web Page Recipe Book	0-13460298-X	01356	32	29.95
<input type="checkbox"/>				Netscape and HTML EXplorer	0-18-83577578	18835	32	39.99
<input type="checkbox"/>				HTML Sourcebook, 2nd Ed.	0-47-114242-5	04711	32	30.00
<input type="checkbox"/>				HTML Plus!	0-53451626-2	05345	32	36.00
<input type="checkbox"/>				Using HTML	0-78-970622-9	07897	32	24.99
<input type="checkbox"/>				Using HTML Special Edition	0-78-970758-6	07897	32	49.99
<input type="checkbox"/>				10 Minute Guide to HTML	0-78970541-9	07897	32	15.00
<input type="checkbox"/>				Green Illusions	0-78970812-4	07897	1	34.99
<input type="checkbox"/>				HTML Quick Reference	0-78970867-1	07897	32	17.99
<input type="checkbox"/>				HTML In Action	1-55615948-X	15561	1	29.95
<input type="checkbox"/>				The Origin of Species	1-56-276352-0	15627	1	24.95
<input type="checkbox"/>				HTML: The Definitive Guide	1-56-592175-5	15659	32	29.95
<input type="checkbox"/>				HTML Publishing for Netscape, Windows Edition	1-56-604288-7	15680	32	39.95
<input type="checkbox"/>				Creating Cool Web Pages with HTML	1-56-884454-9	15688	32	19.99
<input type="checkbox"/>				More HTML For Dummies	1-56-88499-86	15688	32	22.99
<input type="checkbox"/>				Hybrid HTML Design	1-56205617-4	15620	32	35.00
<input type="checkbox"/>				How to Use HTML 3	1-56276390-3	15627	32	24.99

3. ไฟล์ model ของตาราง

ไฟล์ Books.java

: เป็นไฟล์ที่แสดงโครงสร้างของตาราง book ว่ามีตัวแปร และมี method ที่ใช้ในการติดต่ออย่างไร

```
package book;
import java.math.BigDecimal;
import java.util.HashSet;
import java.util.Set;

public class Books implements java.io.Serializable {
    private String isbn;
    private String title;
    private String publisherId;
    private String categoryId;
    private BigDecimal price;
    private Set<Booksauthors> booksauthorses = new HashSet<Booksauthors>(0);

    public Books() {
    }

    public Books(String isbn) {
        this.isbn = isbn;
    }

    public Books(String isbn, String title, String publisherId, String categoryId, BigDecimal price) {
        this.isbn = isbn;
        this.title = title;
        this.publisherId = publisherId;
        this.categoryId = categoryId;
        this.price = price;
        // this.booksauthorses = booksauthorses;
    }

    public String getIsbn() {
        return this.isbn;
    }

    public void setIsbn(String isbn) {
        this.isbn = isbn;
    }
}
```

```
public String getTitle() {
    return this.title;
}

public void setTitle(String title) {
    this.title = title;
}

public String getPublisherId() {
    return this.publisherId;
}

public void setPublisherId(String publisherId) {
    this.publisherId = publisherId;
}

public String getCategoryId() {
    return this.categoryId;
}

public void setCategoryId(String categoryId) {
    this.categoryId = categoryId;
}

public BigDecimal getPrice() {
    return this.price;
}

public void setPrice(BigDecimal price) {
    this.price = price;
}

public Set<Booksauthors> getBooksauthorses() {
    return this.booksauthorses;
}

public void setBooksauthorses(Set<Booksauthors> booksauthorses) {
    this.booksauthorses = booksauthorses;
}
}
```

ไฟล์ Books.hbm.xml

: เป็นไฟล์ที่เป็นตัวกลางในการ Mapping ข้อมูลจาก table ใน database มาอยู่ในรูปแบบไฟล์ java

```
<?xml version="1.0"?>
<!DOCTYPE hibernate-mapping PUBLIC "-//Hibernate/Hibernate Mapping DTD 3.0//EN"
"http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">
<!-- Generated Nov 6, 2016 5:57:08 PM by Hibernate Tools 4.3.1 -->
<hibernate-mapping>
  <class name="book.Books" table="books" catalog="bookstore" optimistic-lock="version">
    <id name="isbn" type="string">
      <column name="ISBN" length="13" />
      <generator class="assigned" />
    </id>
    <property name="title" type="string">
      <column name="Title" length="60" />
    </property>
    <property name="publisherId" type="string">
      <column name="Publisher_Id" length="5" />
    </property>
    <property name="categoryId" type="string">
      <column name="CategoryID" length="5" />
    </property>
    <property name="price" type="big_decimal">
      <column name="Price" precision="10" />
    </property>
    <set name="booksauthors" table="booksauthors" inverse="true" lazy="true" fetch="select">
      <key>
        <column name="ISBN" length="13" not-null="true" />
      </key>
      <one-to-many class="book.Booksauthors" />
    </set>
  </class>
</hibernate-mapping>
```

4. code ไฟล์ Hibernate.cfg.xml

: เป็นไฟล์ที่เป็นชุดของการตั้งค่าที่เกี่ยวข้องกับฐานข้อมูลและพารามิเตอร์อื่น ๆ ที่เกี่ยวข้อง ข้อมูลทั้งหมดจะถูกจัดเป็นไฟล์มาตรฐานคุณสมบัติที่เรียกว่า Javahibernate.properties หรือเป็น hibernate.cfg.xml ไฟล์ XML

```
<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE hibernate-configuration PUBLIC "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
"http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">

<hibernate-configuration>

    <session-factory>

        <property name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>

        <property name="hibernate.connection.driver_class">com.mysql.jdbc.Driver</property>

        <property
name="hibernate.connection.url">jdbc:mysql://localhost:3306/bookstore?zeroDateTimeBehavior=co
nvertToNull</property>

        <property name="hibernate.connection.username">root</property>

        <property name="hibernate.show_sql">true</property>

        <property name="hibernate.current_session_context_class">thread</property>

        <property
name="hibernate.query.factory_class">org.hibernate.hql.internal.classic.ClassicQueryTranslatorFactor
y</property>

        <mapping resource="book/Booksauthors.hbm.xml"/>

        <mapping resource="book/Books.hbm.xml"/>

        <mapping resource="book/Publishers.hbm.xml"/>

        <mapping resource="book/Authors.hbm.xml"/>

        <mapping resource="book/Categories.hbm.xml"/>

    </session-factory>

</hibernate-configuration>
```

5. code ไฟล์ Servlet ทั้งหมด

BookService.java

: เป็นไฟล์ที่เก็บ method ทั้งหมดที่ต้องการเรียกใช้จากหน้า controller ทั้งหมด

```
package book;

import java.io.IOException;
import java.io.PrintWriter;
import java.util.List;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import org.hibernate.Query;
import org.hibernate.Session;
import org.hibernate.Transaction;

public class BookService extends HttpServlet {

    public List getBooks() {
        List booksList = null;
        Session session = null;
        try {
            session = HibernateUtil.getSessionFactory().openSession();
            org.hibernate.Transaction tx = session.beginTransaction();
            Query q = session.createQuery("from Books as books");
            booksList = (List<Books>) q.list();
        } catch (Exception e) {
            e.printStackTrace();
        }
        return booksList;
    }

    public Books searchBooks(String search_books) {
        String message = null;
        Session session = null;
        Transaction tx = null;
```



```
Books books = null;
try {
    session = HibernateUtil.getSessionFactory().openSession();
    tx = session.getTransaction();
    tx.begin();
    Query query = session.createQuery("from Books as books where title = \" + search_books +
    \"");
    books = (Books) query.uniqueResult();
    tx.commit();
} catch (Exception e) {
    if (tx != null) {
        tx.rollback();
    }
    e.printStackTrace();
} finally {
    session.close();
}
return books;
}
```

```
public List searchBooksByName(String search_books) {
    List bookList = null;
    Session session = null;
    try {
        session = HibernateUtil.getSessionFactory().openSession();
        org.hibernate.Transaction tx = session.beginTransaction();
        Query query = session.createQuery("from Books as books where Title like :keyword ");
        query.setParameter("keyword", "%" + search_books + "%");
        bookList = (List<Books>) query.list();
    } catch (Exception e) {
        e.printStackTrace();
    }
    return bookList;
}
```

```

public String addBook(Books book) {
    String message = null;
    Session session = null;
    try {
        session = HibernateUtil.getSessionFactory().openSession();
        Transaction tx = session.getTransaction();
        tx.begin();
        session.save(book);
        tx.commit();
        message = "Saved Data Successful";
        return message;
    } catch (Exception e) {
        return (e.getMessage());
    } finally {
        session.close();
    }
}

/**
 * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
 * methods.
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    try (PrintWriter out = response.getWriter()) {
        /* TODO output your page here. You may use following sample code. */
        out.println("<!DOCTYPE html>");
        out.println("<html>");
        out.println("<head>");
    }
}

```

```

        out.println("<title>Servlet BookService</title>");
        out.println("</head>");
        out.println("<body>");
        out.println("<h1>Servlet BookService at " + request.getContextPath() + "</h1>");
        out.println("</body>");
        out.println("</html>");
    }
}

```

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

```

@Override
protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    processRequest(request, response);
}

```

```

@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    processRequest(request, response);
}

```

```

/**
 * Returns a short description of the servlet.
 * @return a String containing servlet description
 */

```

```

@Override
public String getServletInfo() {
    return "Short description";
}

```

// </editor-fold>

```

}

```

ListController.java

: ทำหน้าที่เป็นส่วน controller ที่ติดต่อกับส่วนของ View และ Model ในส่วนของการเรียกข้อมูลของรายละเอียดหนังสือทั้งหมด

```
package book;
import java.io.IOException;
import java.io.PrintWriter;
import java.util.List;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

public class ListController extends HttpServlet {
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {
            // Create a session object if it is already not created.
            HttpSession session = request.getSession(true);
            try {
                BookService acs = new BookService();
                List booksList = acs.getBooks();
                session.setAttribute("listResult", booksList);
            } catch (Exception e) {
                e.printStackTrace();
            }
            RequestDispatcher rd = request.getRequestDispatcher("booklist.jsp");
            rd.forward(request, response);
        }
    }
}

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left
// to edit the code.">
/**
 * Handles the HTTP <code>GET</code> method.
 * @param request servlet request
 * @param response servlet response
```

```
* @throws ServletException if a servlet-specific error occurs
* @throws IOException if an I/O error occurs
*/
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    processRequest(request, response);
}
/**
 * Handles the HTTP <code>POST</code> method.
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    processRequest(request, response);
}
/**
 * Returns a short description of the servlet.
 * @return a String containing servlet description
 */
@Override
public String getServletInfo() {
    return "Short description";
} // </editor-fold>
}
```

SearchController.java

: ทำหน้าที่เป็นส่วน controller ที่ติดต่อกับส่วนของ View และ Model ในส่วนของการค้นหาหนังสือโดยใช้
ชื่อหนังสือในการค้นหา

```
package book;
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

public class SearchController extends HttpServlet {
    /**
     * Processes requests for both HTTP GET and POST
     * methods.
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {
            HttpSession session = request.getSession(true);
            try {
                String search_book = request.getParameter("searchValue");
                BookService acs = new BookService();
                Books books_obj = acs.searchBooks(search_book);
                session.setAttribute("searchResult", books_obj);
                RequestDispatcher rd = request.getRequestDispatcher("books.jsp");
                rd.forward(request, response);
            } finally {
                out.close();
            }
        }
    }
}
```

```

    }
}

```

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

```

/**
 * Handles the HTTP <code>GET</code> method.
 *
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    processRequest(request, response);
}

```

```

/**
 * Handles the HTTP <code>POST</code> method.
 *
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    processRequest(request, response);
}

```

```

@Override
public String getServletInfo() {
    return "Short description";
}

```

```

}

```

SearchByNameController.java

: ทำหน้าที่เป็นส่วน controller ที่ติดต่อกับส่วนของ View และ Model ในส่วนของการค้นหาหนังสือโดยใช้
บางส่วนของชื่อหนังสือในการค้นหา

```
package book;
import java.io.IOException;
import java.io.PrintWriter;
import java.util.List;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

public class SearchByNameController extends HttpServlet {
    /**
     * Processes requests for both HTTP GET and POST
     * methods.
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {
            // Create a session object if it is already not created.
            HttpSession session = request.getSession(true);
            try {
                String searchValue = request.getParameter("searchValue");
                BookService acs = new BookService();
                List bookList = acs.searchBooksByName(searchValue);
                int num = bookList.size();
                session.setAttribute("bookListResult", bookList);
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    }
}
```



```

        RequestDispatcher rd = request.getRequestDispatcher("bookSearchList.jsp");
        rd.forward(request, response);
    }
}

```

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

```

/**
 * Handles the HTTP <code>GET</code> method.
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    processRequest(request, response);
}

```

```

/**
 * Handles the HTTP <code>POST</code> method.
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    processRequest(request, response);
}

```

```

@Override
public String getServletInfo() {
    return "Short description";
}

```

}

AddBookController.java

: ทำหน้าที่เป็นส่วน controller ที่ติดต่อกับส่วนของ View และ Model ในส่วนของการเพิ่มรายการหนังสือลงบน database

```
package book;
import java.io.IOException;
import java.io.PrintWriter;
import java.math.BigDecimal;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

public class AddBookController extends HttpServlet {
    /**
     * Processes requests for both HTTP GET and POST
     * methods.
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {
            out.println("add new book");
            HttpSession session = request.getSession(true);
            String isbn = request.getParameter("isbn");
            String title = request.getParameter("title");
            String publisherId = request.getParameter("publisherID");
            String categoryId = request.getParameter("categoryId");
            BigDecimal p = new BigDecimal(request.getParameter("price"));

            Books ac = new Books(title, isbn, publisherId, categoryId, p);
            BookService acs = new BookService();
            String resultMessage = acs.addBook(ac);
```

```

        RequestDispatcher rd = request.getRequestDispatcher("addbook.jsp?message=" +
resultMessage);
        request.setAttribute("searchResult", null);
        rd.forward(request, response);
    }
}

```

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

```

/**
 * Handles the HTTP <code>GET</code> method.
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    processRequest(request, response);
}

/**
 * Handles the HTTP <code>POST</code> method.
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    processRequest(request, response);
}

@Override
public String getServletInfo() {
    return "Short description";
}
}

```

6. code ไฟล์ JSP ทั้งหมด

booklist.jsp

: เป็นไฟล์ที่ใช้สำหรับใช้แสดงผลรายการหนังสือทั้งหมด

```
<%@page import="java.util.List"%>
<%@page import="book.Books"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JSP Page</title>
  </head>
  <div style="background-color: #D7BDE2; width:100%;">
    <br>
    <a href="SearchController">Search Book By Title</a>
    <a href="SearchByNameController">Search Book By Name</a>
    <a href="addbook.jsp">Add new book</a>
    <a href="ListController">List of books</a>
    <br><br>
  </div>
  <center>
    <body>
      <div style="background-color: #F5EEF8 ; width:100%;">
        <br>
        <h2 align = 'center'>Books List</h2>
      <%

        try {
          List<Books> booksList = (List<Books>) session.getAttribute("listResult");
          if (booksList != null) {
            // out.println("not null");
          }
        }

      %>
      <table bgcolor = '#FFFF99' border="1">
```

```

<tbody>
  <tr>
    <td bgcolor = '#FFFFD7'>Book ISBN</td>
    <td bgcolor = '#FFFFD7'>Title</td>
    <td bgcolor = '#FFFFD7'>Publisher ID</td>
    <td bgcolor = '#FFFFD7'>Category ID</td>
    <td bgcolor = '#FFFFD7'>Price</td>
  </tr>
  <%
      for (Books acc : booksList) {
  %>

  <tr>
    <td bgcolor = '#FFFFFFD'><% out.println(acc.getIsbn()); %></td>
    <td bgcolor = '#FFFFFFD'><% out.println(acc.getTitle()); %></td>
    <td bgcolor = '#FFFFFFD'><% out.println(acc.getPublisherId()); %></td>
    <td bgcolor = '#FFFFFFD'><% out.println(acc.getCategoryId()); %></td>
    <td bgcolor = '#FFFFFFD'><% out.println(acc.getPrice()); %></td>
  </tr>
  <%} %>
</tbody>
</table>
<%
    } else {
        out.println("return null");
    }
    } catch (Exception e) {
        e.printStackTrace();
        out.println("error");
    }
  %>
</body>
</center>
</html>

```

books.jsp

: เป็นไฟล์ที่ไว้สำหรับใช้แสดงผลรายการหนังสือ ที่ได้จากการค้นหาโดยใช้ชื่อของหนังสือ (by book title)

```
<%@page import="book.Books"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JSP Page</title>
  </head>
  <div style="background-color: #D8F0F8; width:100%;">
    <br>
    <a href="SearchController">Search Book By Title</a>
    <a href="SearchByNameController">Search Book By Name</a>
    <a href="addbook.jsp">Add new book</a>
    <a href="ListController">List of books</a>
    <br><br>
  </div>

  <center>
    <body>
      <div style="background-color: #EFF9FC; width:100%;">
        <br>
        <h2 align = 'center'>Books Search Form By Title</h2>
        <form name="searchForm" action="SearchController">
          Enter Book Title : <input type="text" name="searchValue" value="" >
          <input type="submit" value="Find Books Details" name="searchButton" />
        </form>

        <%
          try {
            // Create a session object if it is already not created.
            Books acc = (Books) session.getAttribute("searchResult");
```

```
        if (acc != null) {
    %>
    <br>
    <table bgcolor = '#FDFEFF' border="1">
      <tbody>
        <tr>
          <td bgcolor = '#FFF7B7'>Books ISBN</td>
          <td bgcolor = '#FFFEF7'><% out.println(acc.getIsbn()); %></td>
        </tr>

        <tr>
          <td bgcolor = '#FFF7B7'>Books Title</td>
          <td bgcolor = '#FFFEF7'><% out.println(acc.getTitle()); %></td>
        </tr>

        <tr>
          <td bgcolor = '#FFF7B7'>Books Price</td>
          <td bgcolor = '#FFFEF7'><% out.println(acc.getPrice()); %></td>
        </tr>

      </tbody>
    </table>

    <%
      }
    } catch (Exception e) {
      out.println(e.getMessage());
    }
    %>
  <br><br>
  </body>
</center>
</html>
```

bookSearchList.jsp

: เป็นไฟล์ที่ไว้สำหรับใช้แสดงผลรายการหนังสือ ที่ได้จากการค้นหาหนังสือโดยใช้บางส่วนของชื่อหนังสือ
(by some part of book title)

```
<%@page import="java.util.List"%>
<%@page import="book.Books"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JSP Page</title>
  </head>
  <div style="background-color: #FFCECE; width:100%;">
    <br>
    <a href="SearchController">Search Book By Title</a>
    <a href="SearchByNameController">Search Book By Name</a>
    <a href="addbook.jsp">Add new book</a>
    <a href="ListController">List of books</a>
    <br><br>
  </div>
  <center>
    <body>
      <div style="background-color: #FFF2F2; width:100%;">
        <br><h1 align = 'center'>Books Search Form By Book Name</h1>
        <form name="formSearchBook" action="SearchByNameController" >
          Search by title: <input type="text" name="searchValue" value="" />
          <input type="submit" value="Search" name="SearchBook" />
        </form><br>
        <%
          try {
            List<Books> booksList = (List<Books>) session.getAttribute("bookListResult");
            if (booksList != null) {
              if (booksList.size() != 0) {
```



```

%>
<table bgcolor = '#FDFEFF' border="1">
  <tbody>
    <tr>
      <td bgcolor = '#FFFFD7'>Book ISBN</td>
      <td bgcolor = '#FFFFD7'>Title</td>
      <td bgcolor = '#FFFFD7'>Publisher ID</td>
      <td bgcolor = '#FFFFD7'>Category ID</td>
      <td bgcolor = '#FFFFD7'>Price</td>
    </tr>
    <%
      for (Books acc : booksList) {
    %>
    <tr>
      <td bgcolor = '#FFFEFB'><% out.println(acc.getIsbn()); %></td>
      <td bgcolor = '#FFFEFB'><% out.println(acc.getTitle()); %></td>
      <td bgcolor = '#FFFEFB'><% out.println(acc.getPublisherId()); %></td>
      <td bgcolor = '#FFFEFB'><% out.println(acc.getCategoryId()); %></td>
      <td bgcolor = '#FFFEFB'><% out.println(acc.getPrice()); %></td>
    </tr>
    <% } %>
  </tbody>
</table>
<%
  }
  } else { out.println("return null"); }
} catch (Exception e) {
  e.printStackTrace();
  out.println("error");
}
%>
<br><br>
</body>
</center>
</html>

```

addBook.jsp

: เป็นไฟล์ที่ไว้สำหรับใช้แสดงการเพิ่มรายการหนังสือ

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>JSP Page</title>
    </head>
    <div style="background-color: #FFF06A; width:100%;">
        <br>
        <a href="SearchController">Search Book By Title</a>
        <a href="SearchByNameController">Search Book By Name</a>
        <a href="addbook.jsp">Add new book</a>
        <a href="ListController">List of books</a>
        <br><br>
    </div>
    <center>
        <body>
            <div style="background-color: #FFFBDF; width:100%;">
                <br>
                <h1 align = 'center'>Add New Book</h1>
                <%
                    String message = request.getParameter("message");
                    if (message != null) {
                        out.println(message);
                    }%>
                <form name="formAddBook" action="AddBookController">
                    <table border = '0' style = 'width: 300px' align = 'center'>
                        <tbody>
                            <tr>
                                <td> &nbsp;Title: </td>
                                <td><input type="text" name="title" value="" /></td>
```

```
</tr>
<tr>
  <td>  ISBN: </td>
  <td><input type="text" name="isbn" value="" /></td>
</tr>
<tr>
  <td>  Publisher Id : </td>
  <td><input type="text" name="publisherID" value="" /></td>
</tr>
<tr>
  <td>  Category Id :</td>
  <td><input type="text" name="categoryID" value="" /></td>
</tr>
<tr>
  <td>  Price :</td>
  <td><input type="text" name="price" value="" /></td>
</tr>
</tbody>
</table>
<br>
<input type="submit" value="Submit" name="AddBook" /><br>
</form>
<br><br>
</body>
</center>
</html>
```