



# Bài 06

## Client Với Restful Web Services

- ❖ Tạo ứng dụng web client để sử dụng các chức năng của restful web services:
  - Load dữ liệu
  - Insert
  - Update
  - Delete
  - Show detail

eclipse-workspace - Webservice\_Client/pom.xml - Eclipse IDE

File Edit Source Navigate Search Project Run Window Help

Project Explorer

- Servers
- WebService\_Client
  - Deployment Descriptor: Webservice\_Client
  - JAX-WS Web Services
  - Java Resources
  - Deployed Resources
  - build
  - target
  - WebContent
    - META-INF
    - WEB-INF
      - index.jsp
      - insertBook.jsp
      - listBooks.jsp
  - pom.xml
- WebService\_Session05\_DB

WebService\_Client/pom.xml

```
1<?xml version="1.0" encoding="UTF-8" ?>
2<project xmlns="http://maven.apache.org/POM/4.0.0"
3  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
5  <modelVersion>4.0.0</modelVersion>
6  <groupId>WebService_Client</groupId>
7  <artifactId>WebService_Client</artifactId>
8  <version>0.0.1-SNAPSHOT</version>
9  <packaging>war</packaging>
10 <dependencies>
11   <!-- https://mvnrepository.com/artifact/com.sun.jersey/jersey-client -->
12   <dependency>
13     <groupId>com.sun.jersey</groupId>
14     <artifactId>jersey-client</artifactId>
15     <version>1.19.4</version>
16   </dependency>
17   <!-- https://mvnrepository.com/artifact/com.google.code.gson/gson -->
18   <dependency>
19     <groupId>com.google.code.gson</groupId>
20     <artifactId>gson</artifactId>
21     <version>2.8.6</version>
22   </dependency>
23   <!-- https://mvnrepository.com/artifact/jstl/jstl -->
24   <dependency>
25     <groupId>jstl</groupId>
26     <artifactId>jstl</artifactId>
27     <version>1.2</version>
28   </dependency>
29 </dependencies>
30 <build>
31   <sourceDirectory>src</sourceDirectory>
32   <plugins>
33     <plugin>
34       <artifactId>maven-compiler-plugin</artifactId>
35       <version>3.8.1</version>
36       <configuration>
37         <source>1.8</source>
38         <target>1.8</target>
39       </configuration>
40     </plugin>
41   </plugins>
42 </build>
43 </project>
```

Overview Dependencies Dependency Hierarchy Effective POM pom.xml

Writable Smart Insert 1:1:0

# Tạo trang index.jsp để gọi tới Servlet

eclipse-workspace - Webservice\_Client/WebContent/index.jsp - Eclipse IDE

File Edit Navigate Search Project Run Window Help

The screenshot shows the Eclipse IDE interface. On the left, the Project Explorer displays the project structure for 'Webservice\_Client'. The 'WebContent' folder is expanded, showing 'index.jsp' selected. The main editor window displays the content of 'index.jsp', which is a JSP page that forwards to 'LoadBooks'.

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"%>
2   pageEncoding="UTF-8"%>
3   <!DOCTYPE html>
4   <html>
5   <head>
6     <meta charset="UTF-8">
7     <title>Insert title here</title>
8   </head>
9   <body>
10     <jsp:forward page="LoadBooks"></jsp:forward>
11   </body>
12 </html>
```

# Tạo Servlet gọi chức năng getListBook

```
LoadBooks.java
27  * @see HttpServlet#HttpServlet()
28  */
29  public LoadBooks() {
30      super();
31      // TODO Auto-generated constructor stub
32  }
33
34  /**
35   * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
36   */
37  protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
38      // TODO Auto-generated method stub
39      Client client = Client.create();
40      WebResource resource = client.resource("http://localhost:8080/Webservice_Session05_DB/rest/book-service/listBooks");
41      String data = resource.get(String.class);
42
43      Gson son = new Gson();
44      GenericType<List<Book>> listBookType = new GenericType<List<Book>>() {};
45      List<Book> listBook = son.fromJson(data, listBookType.getType());
46
47      //xu ly success
48      Object success = request.getAttribute("success");
49      if(success!=null) {
50          request.setAttribute("success", success);
51      }
52      request.setAttribute("list", listBook);
53      request.getRequestDispatcher("listBooks.jsp").forward(request, response);
54  }
55
56  /**
57   * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
58   */
59  protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
60      // TODO Auto-generated method stub
61      doGet(request, response);
62  }
63
64  }
65
```

# Tạo trang jsp hiển thị chức năng load dữ liệu

```
LoadBooks.java
27  * @see HttpServlet#HttpServlet()
28  */
29  public LoadBooks() {
30      super();
31      // TODO Auto-generated constructor stub
32  }
33
34  /**
35   * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
36   */
37  protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
38      // TODO Auto-generated method stub
39      Client client = Client.create();
40      WebResource resource = client.resource("http://localhost:8080/Webservice_Session05_DB/rest/book-service/listBooks");
41      String data = resource.get(String.class);
42
43      Gson son = new Gson();
44      GenericType<List<Book>> listBookType = new GenericType<List<Book>>() {};
45      List<Book> listBook = son.fromJson(data, listBookType.getType());
46
47      //xu ly success
48      Object success = request.getAttribute("success");
49      if(success!=null) {
50          request.setAttribute("success", success);
51      }
52      request.setAttribute("list", listBook);
53      request.getRequestDispatcher("listBooks.jsp").forward(request, response);
54  }
55
56  /**
57   * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
58   */
59  protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
60      // TODO Auto-generated method stub
61      doGet(request, response);
62  }
63
64  }
65
```

Insert title here



localhost:8080/Webservice\_Client/



## LIST BOOKS

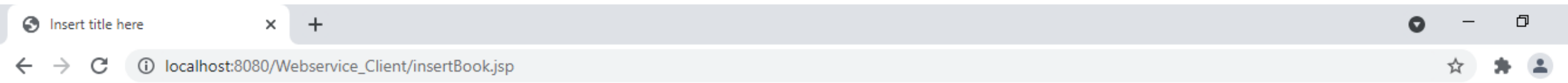
Book Id	Book Name	Author	Publisher	Publish Date	Pages	Price
1	Lập trình Java	Nguyễn Thanh Hải	NXB Giáo Dục	Mon Dec 21 00:00:00 ICT 2020	1500	300000.0

# Tạo trang jsp để thêm mới dữ liệu

```
insertBook.jsp
10 <h1>INSERT NEW BOOK</h1>
11 <h3 style="color:red">${error}</h3>
12
13 <form action="InsertBook" method="post">
14   <table>
15     <tr>
16       <td>Book Name:</td>
17       <td><input type="text" name="bookName"/></td>
18     </tr>
19     <tr>
20       <td>Author:</td>
21       <td><input type="text" name="author"/></td>
22     </tr>
23     <tr>
24       <td>Publisher:</td>
25       <td><input type="text" name="publisher"/></td>
26     </tr>
27     <tr>
28       <td>Publish Date:</td>
29       <td><input type="date" name="publishDate"/></td>
30     </tr>
31     <tr>
32       <td>Pages:</td>
33       <td><input type="number" name="pages"/></td>
34     </tr>
35     <tr>
36       <td>Price:</td>
37       <td><input type="number" name="price"/></td>
38     </tr>
39     <tr>
40       <td></td>
41       <td>
42         <input type="submit" value="Insert"/>
43         <input type="reset" value="Clear"/>
44       </td>
45     </tr>
46   </table>
47 </form>
48 </body>
```



```
InsertBook.java
41 request.setCharacterEncoding("UTF-8");
42 String bookName = request.getParameter("bookName");
43 String author = request.getParameter("author");
44 String publisher = request.getParameter("publisher");
45 Date publishDate = null;
46 SimpleDateFormat sf = new SimpleDateFormat("yyyy-MM-dd");
47 try {
48     publishDate = sf.parse(request.getParameter("publishDate"));
49 } catch (ParseException e) {
50     // TODO Auto-generated catch block
51     e.printStackTrace();
52 }
53 int pages = Integer.parseInt(request.getParameter("pages"));
54 double price = Double.parseDouble(request.getParameter("price"));
55
56 Book book = new Book();
57 book.setBookName(bookName);
58 book.setAuthor(author);
59 book.setPublisher(publisher);
60 book.setPublishDate(publishDate);
61 book.setPages(pages);
62 book.setPrice(price);
63
64 //goi webservice de insert
65 Gson son = new Gson();
66 String jsonDataBook = son.toJson(book);
67
68 Client client = Client.create();
69 WebResource resource = client.resource("http://localhost:8080/WebService_Session05_DB/rest/book-service/insertBook");
70 ClientResponse clientResponse = resource.type("application/json").post(ClientResponse.class, jsonDataBook);
71 String result = clientResponse.getEntity(String.class);
72 Boolean bl = son.fromJson(result, Boolean.class);
73
74 if(bl) {
75     request.setAttribute("success", "Insert successfully!");
76     request.getRequestDispatcher("LoadBooks").forward(request, response);
77 }else {
78     request.setAttribute("error", "Insert failed!");
79     request.getRequestDispatcher("insertBook.jsp").forward(request, response);
```




## INSERT NEW BOOK

Book Name:

Author:

Publisher:

Publish Date:  

Pages:

Price:



## LIST BOOKS

Insert successfully!

Book Id	Book Name	Author	Pubisher	Publish Date	Pages	Price	
1	Lập trình Java	Nguyễn Thanh Hải	NXB Giáo Dục	Mon Dec 21 00:00:00 ICT 2020	1500	300000.0	<a href="#">detail</a>
2	Lập trình Java	Nguyễn Hải Nam	NXB Giáo Dục	Mon Dec 21 00:00:00 ICT 2020	700	250000.0	<a href="#">detail</a>

[Add New Book](#)

```
detailBook.jsp
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6 <meta charset="UTF-8">
7 <title>Insert title here</title>
8 </head>
9 <body>
10   <h1>DETAIL BOOK</h1>
11   <table>
12     <tr>
13       <td>Book Name:</td>
14       <td>${b.bookName}</td>
15     </tr>
16     <tr>
17       <td>Author:</td>
18       <td>${b.author}</td>
19     </tr>
20     <tr>
21       <td>Publisher:</td>
22       <td>${b.publisher}</td>
23     </tr>
24     <tr>
25       <td>Publish Date:</td>
26       <td>${b.publishDate}</td>
27     </tr>
28     <tr>
29       <td>Pages:</td>
30       <td>${b.pages}</td>
31     </tr>
32     <tr>
33       <td>Price:</td>
34       <td>${b.price}</td>
35     </tr>
36   </table>
37   <a href="index.jsp">Back</a>
38 </body>
39 </html>
```

Insert title here

x

+

localhost:8080/Webservice\_Client/DetailBook?bookId=2



## DETAIL BOOK

Book Name: Lập trình Java

Author: Nguyễn Hải Nam

Publisher: NXB Giáo Dục

Publish Date: Mon Dec 21 00:00:00 ICT 2020

Pages: 700

Price: 250000.0

[Back](#)

# Tạo servlet xử lý dữ liệu lên trang update

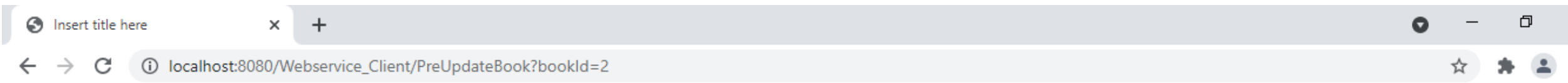
```
PreUpdateBook.java
22 private static final long serialVersionUID = 1L;
23
24 /**
25  * @see HttpServlet#HttpServlet()
26  */
27 public PreUpdateBook() {
28     super();
29     // TODO Auto-generated constructor stub
30 }
31
32 /**
33  * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
34  */
35 protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
36     // TODO Auto-generated method stub
37     int bookId = Integer.parseInt(request.getParameter("bookId"));
38
39     Client client = Client.create();
40     WebResource resource = client.resource("http://localhost:8080/Webservice_Session05_DB/rest/book-service/getBookById/"+bookId);
41     resource.accept(MediaType.APPLICATION_JSON);
42     String data = resource.get(String.class);
43
44     Gson son = new Gson();
45     Book book = son.fromJson(data, Book.class);
46
47     request.setAttribute("b", book);
48     request.getRequestDispatcher("updateBook.jsp").forward(request, response);
49 }
50
51 /**
52  * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
53  */
54 protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
55     // TODO Auto-generated method stub
56     doGet(request, response);
57 }
58
59 }
```

# Trang jsp hiển thị dữ liệu để update


```
updateBook.jsp
13
14<form action="UpdateBook" method="post">
15  <table>
16    <tr>
17      <td>Book Id:</td>
18      <td><input type="text" name="bookId" value="${b.bookId}" readonly="readonly"/></td>
19    </tr>
20    <tr>
21      <td>Book Name:</td>
22      <td><input type="text" name="bookName" value="${b.bookName}"/></td>
23    </tr>
24    <tr>
25      <td>Author:</td>
26      <td><input type="text" name="author" value="${b.author}"/></td>
27    </tr>
28    <tr>
29      <td>Publisher:</td>
30      <td><input type="text" name="publisher" value="${b.publisher}"/></td>
31    </tr>
32    <tr>
33      <td>Publish Date:</td>
34      <td><input type="date" name="publishDate" value="<fmt:formatDate pattern="yyyy-MM-dd" value="${b.publishDate}"/></td>
35    </tr>
36    <tr>
37      <td>Pages:</td>
38      <td><input type="number" name="pages" value="${b.pages}"/></td>
39    </tr>
40    <tr>
41      <td>Price:</td>
42      <td><input type="number" name="price" value="${b.price}"/></td>
43    </tr>
44    <tr>
45      <td></td>
46      <td>
47        <input type="submit" value="Update"/>
48        <input type="reset" value="Clear"/>
49      </td>
50    </tr>
51  </table>
```

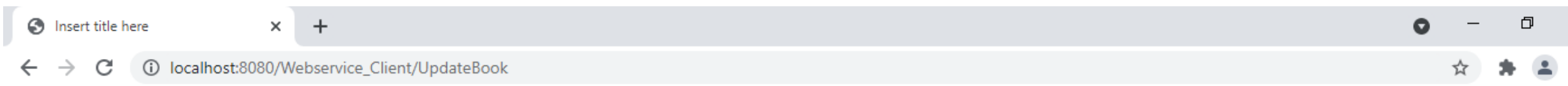
```
UpdateBook.java
39 protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
40     request.setCharacterEncoding("UTF-8");
41     int bookId = Integer.parseInt(request.getParameter("bookId"));
42     String bookName = request.getParameter("bookName");
43     String author = request.getParameter("author");
44     String publisher = request.getParameter("publisher");
45     Date publishDate = null;
46     SimpleDateFormat sf = new SimpleDateFormat("yyyy-MM-dd");
47     try {
48         publishDate = sf.parse(request.getParameter("publishDate"));
49     } catch (ParseException e) {
50         // TODO Auto-generated catch block
51         e.printStackTrace();
52     }
53     int pages = Integer.parseInt(request.getParameter("pages"));
54     double price = Double.parseDouble(request.getParameter("price"));
55
56     Book book = new Book();
57     book.setBookId(bookId);
58     book.setBookName(bookName);
59     book.setAuthor(author);
60     book.setPublisher(publisher);
61     book.setPublishDate(publishDate);
62     book.setPages(pages);
63     book.setPrice(price);
64
65     //goi webservice de insert
66     Gson son = new Gson();
67     String jsonDataBook = son.toJson(book);
68
69     Client client = Client.create();
70     WebResource resource = client.resource("http://localhost:8080/Webservice_Session05_DB/rest/book-service/updateBook");
71     ClientResponse clientResponse = resource.type("application/json").post(ClientResponse.class, jsonDataBook);
72     String result = clientResponse.getEntity(String.class);
73     Boolean bl = son.fromJson(result, Boolean.class);
74
75     if(bl) {
76         request.setAttribute("success", "Update successfully!");
77         request.getRequestDispatcher("LoadBooks").forward(request, response);
78     }
```





## UPDATE BOOK

Book Id:	<input type="text" value="2"/>
Book Name:	<input type="text" value="Lập trình Java"/>
Author:	<input type="text" value="Nguyễn Hải Nam"/>
Publisher:	<input type="text" value="NXB Giáo Dục"/>
Publish Date:	<input type="text" value="21/12/2020"/> 
Pages:	<input type="text" value="700"/>
Price:	<input type="text" value="250000,0"/>
	<input type="button" value="Update"/> <input type="button" value="Clear"/>



## LIST BOOKS

Upupdate successfully!

Book Id	Book Name	Author	Pubisher	Publish Date	Pages	Price	
1	Lập trình Java	Nguyễn Thanh Hải	NXB Giáo Dục	Mon Dec 21 00:00:00 ICT 2020	1500	300000.0	<a href="#">detail</a>
2	Lập trình Web API	Nguyễn Đức Nam	NXB Giáo Dục	Mon Dec 21 00:00:00 ICT 2020	700	250000.0	<a href="#">detail</a>

[Add New Book](#)

```
DeleteBook.java
26  */
27  public DeleteBook() {
28      super();
29      // TODO Auto-generated constructor stub
30  }
31
32  /**
33   * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
34   */
35  protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
36      // TODO Auto-generated method stub
37      int bookId = Integer.parseInt(request.getParameter("bookId"));
38
39      Client client = Client.create();
40      WebResource resource = client.resource("http://localhost:8080/Webservice_Session05_DB/rest/book-service/deleteBook/"+bookId);
41      resource.accept(MediaType.APPLICATION_JSON);
42      String data = resource.type(MediaType.APPLICATION_JSON).delete(String.class);
43
44      Gson son = new Gson();
45      Boolean bl = son.fromJson(data, Boolean.class);
46
47      if(bl) {
48          request.setAttribute("success", "Delete successfully!");
49      } else {
50          request.setAttribute("error", "Delete failed!");
51      }
52      request.getRequestDispatcher("LoadBooks").forward(request, response);
53  }
54
55  /**
56   * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
57   */
58  protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
59      // TODO Auto-generated method stub
60      doGet(request, response);
61  }
62
63 }
```

Insert title here x +

localhost:8080/Webservice\_Client/

## LIST BOOKS

Book Id	Book Name	Author	Pubisher	
1	Lập trình Java	Nguyễn Thanh Hải	NXB Giáo Dục	Mon
2	Lập trình Web API	Nguyễn Đức Nam	NXB Giáo Dục	Mon Dec 21 00:00:00 ICT 2020

[Add New Book](#)

localhost:8080 cho biết  
Sure?  
OK Hủy

700250000.0detaildelete

Insert title here x +

localhost:8080/Webservice\_Client/DeleteBook?bookId=2

## LIST BOOKS

Delete successfully!

Book Id	Book Name	Author	Publisher	Publish Date	Pages	Price	Detail	Delete
1	Lập trình Java	Nguyễn Thanh Hải	NXB Giáo Dục	Mon Dec 21 00:00:00 ICT 2020	1500	300000.0	<a href="#">detail</a>	<a href="#">delete</a>

[Add New Book](#)

# HỎI ĐÁP





# TRẢI NGHIỆM THỰC HÀNH



## TRÂN TRỌNG CẢM ƠN!



238 Hoàng Quốc Việt, Bắc Từ Liêm, Hà Nội



0968.27.6996



[tuyensinh@bachkhoa-aptech.edu.vn](mailto:tuyensinh@bachkhoa-aptech.edu.vn)



[www.bachkhoa-aptech.edu.vn](http://www.bachkhoa-aptech.edu.vn)