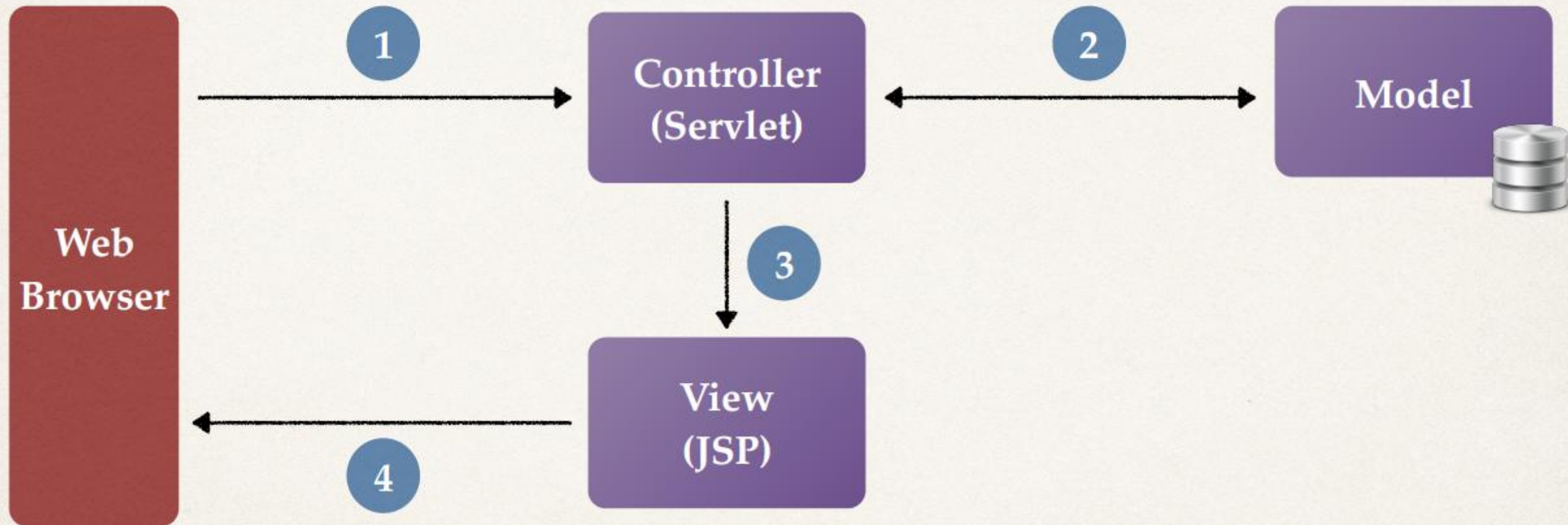


# MVC with Servlets & JSP

# Best Practice

- Integrate Servlets and JSP together!
  - Servlet does the business logic
  - JSP handles the presentation view
- Model-View-Controller (MVC) Design Pattern

# Model-View-Controller (MVC)



# Benefits of MVC

- Minimizes HTML code in Servlet
  - no more: `out.println(...)` in Servlet code
- Minimize Java business logic in JSPs
  - no more large scriptlets in JSP code



# Servlet Can Call a JSP

- Servlet can call JSP using a *request dispatcher*

**// Step 1: get request dispatcher**

```
RequestDispatcher dispatcher =  
    request.getRequestDispatcher("/view_students.jsp");
```

**// Step 2: forward to the JSP**

```
dispatcher.forward(request, response);
```

# Sending Data to JSP

- Servlet can add data to request object

// Step 0: Add data

```
String[] students = {"Susan", "Anil", "Mohamed", "Trupti"};  
request.setAttribute("student_list", students);
```

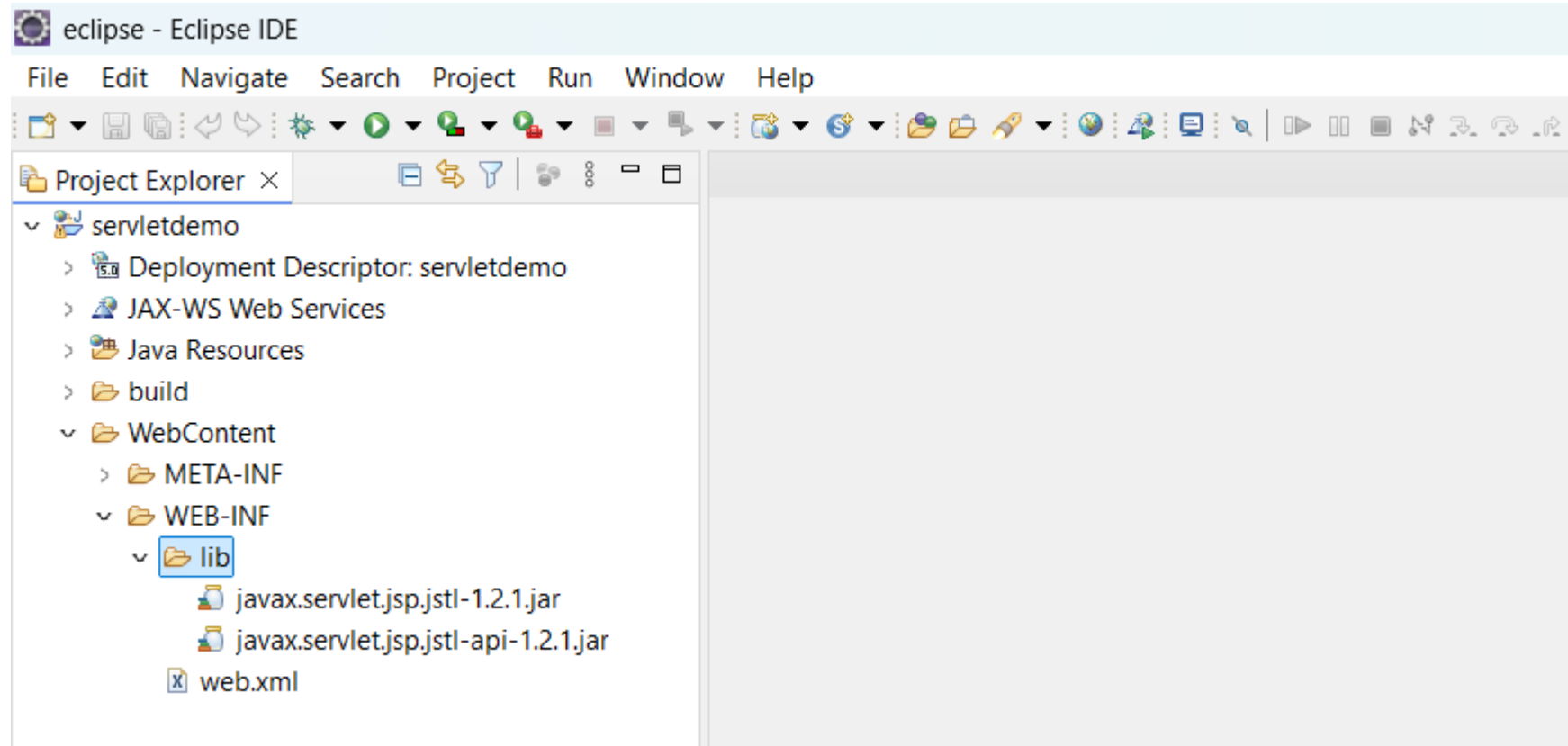
// Step 1: get request dispatcher

```
RequestDispatcher dispatcher =  
    request.getRequestDispatcher("/view_students.jsp");
```

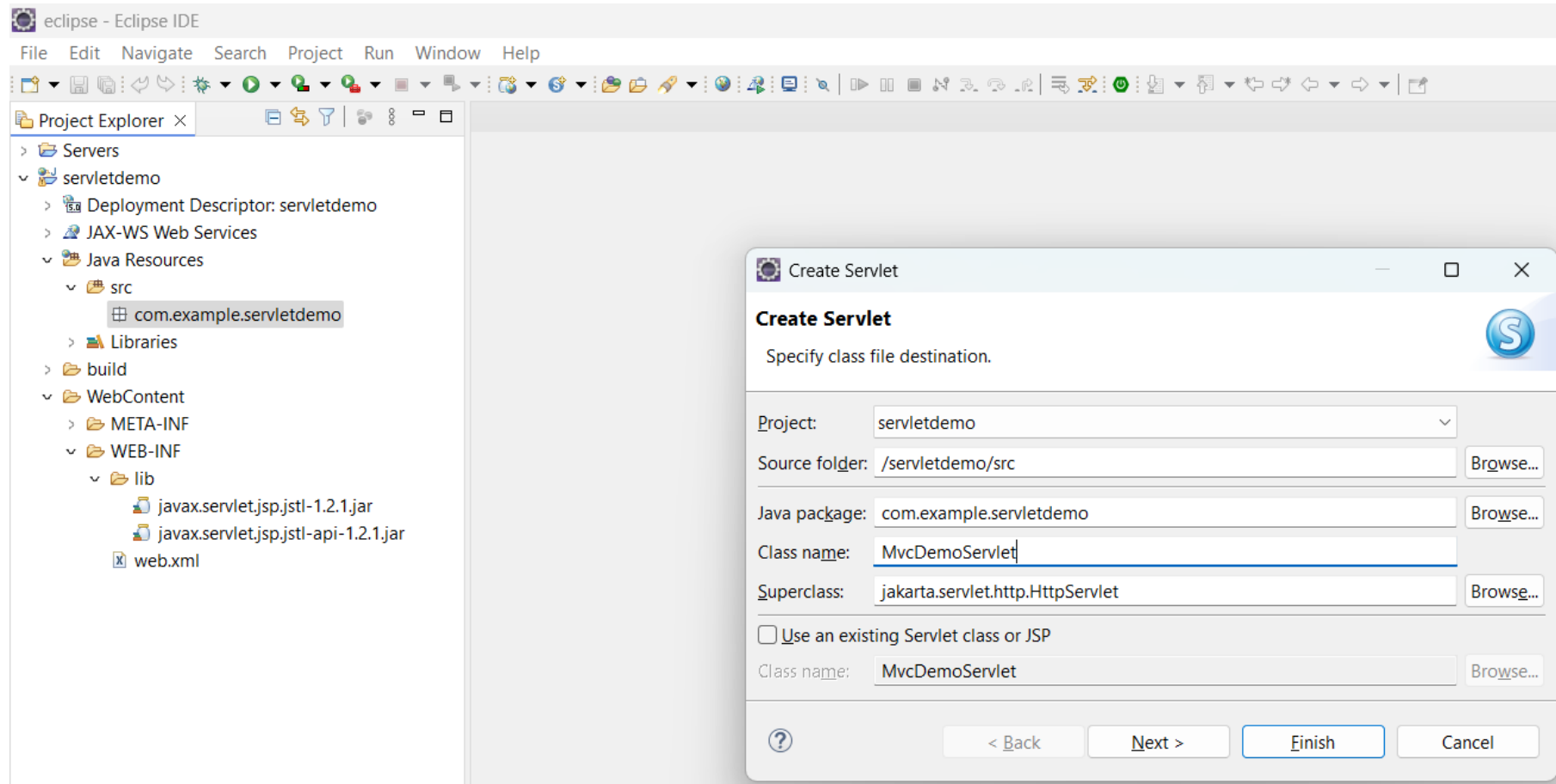
// Step 2: forward to the JSP

```
dispatcher.forward(request, response);
```

# Create project & copy/paste jstl libraries



# Create a controller servlet



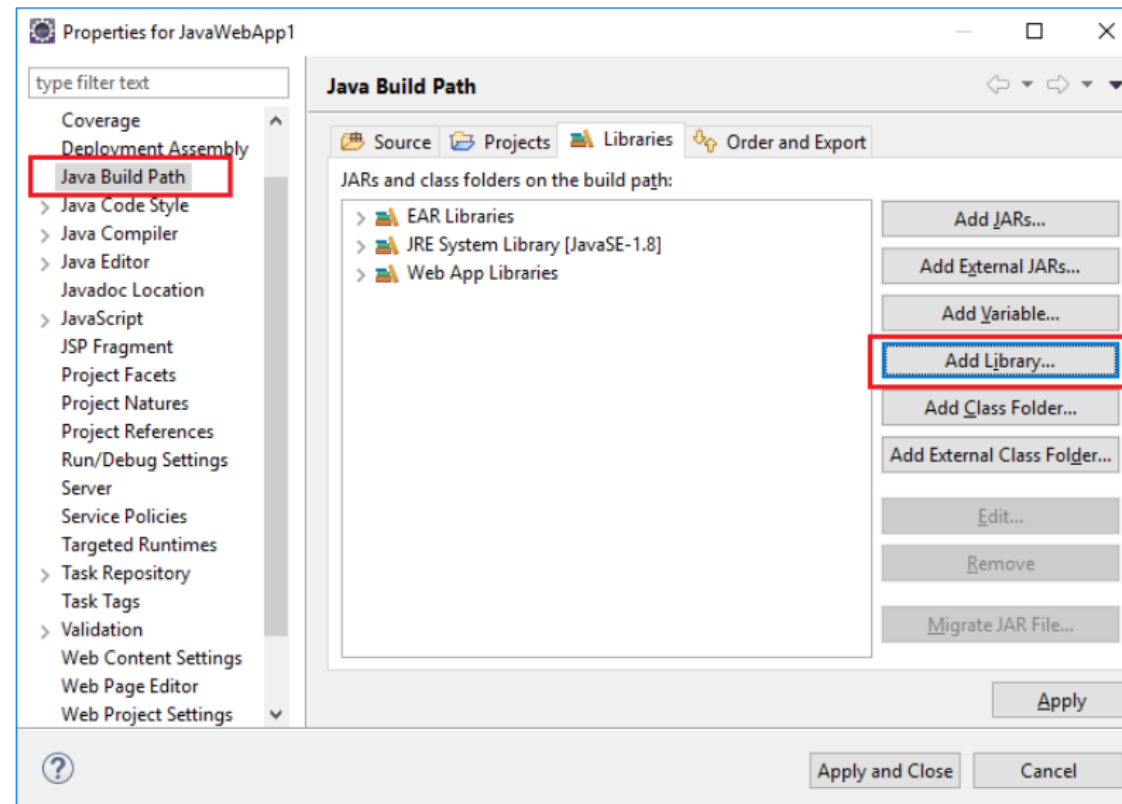


# An error

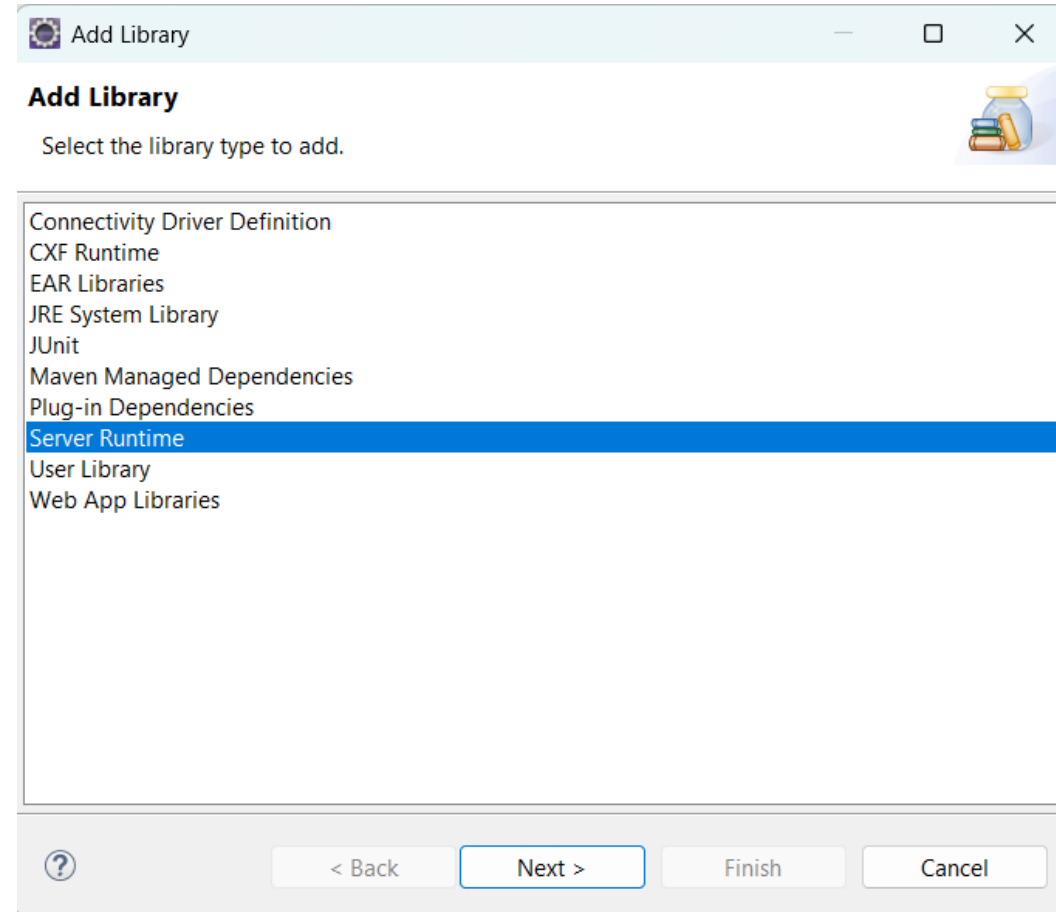
```
MvcDemoServlet.java ×
1 package com.example.servletdemo;
2
3 import jakarta.servlet.ServletException;
4 import jakarta.servlet.annotation.WebServlet;
5 import jakarta.servlet.http.HttpServlet;
6 import jakarta.servlet.http.HttpServletRequest;
7 import jakarta.servlet.http.HttpServletResponse;
8 import java.io.IOException;
9
10 /**
11  * Servlet implementation class MvcDemoServlet
12  */
13 @WebServlet("/MvcDemoServlet")
14 public class MvcDemoServlet extends HttpServlet {
15     private static final long serialVersionUID = 1L;
16
17     /**
18      * Default constructor.
19      */
20     public MvcDemoServlet() {
21         // TODO Auto-generated constructor stub
22     }
23 }
```

# The fix

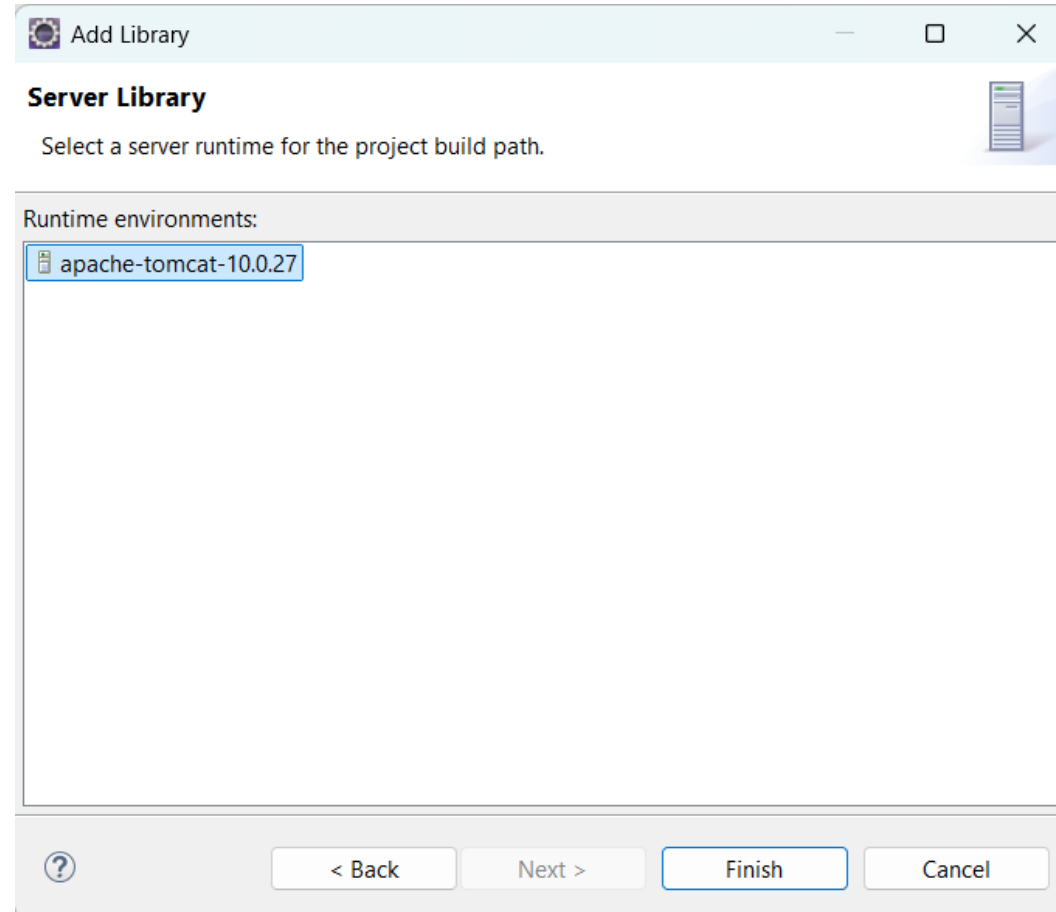
In Eclipse, right click on the project, click **Properties**. In the project properties dialog, click **Java Build Path**, and click the button **Add Library...** as shown below:



# The fix



# The fix

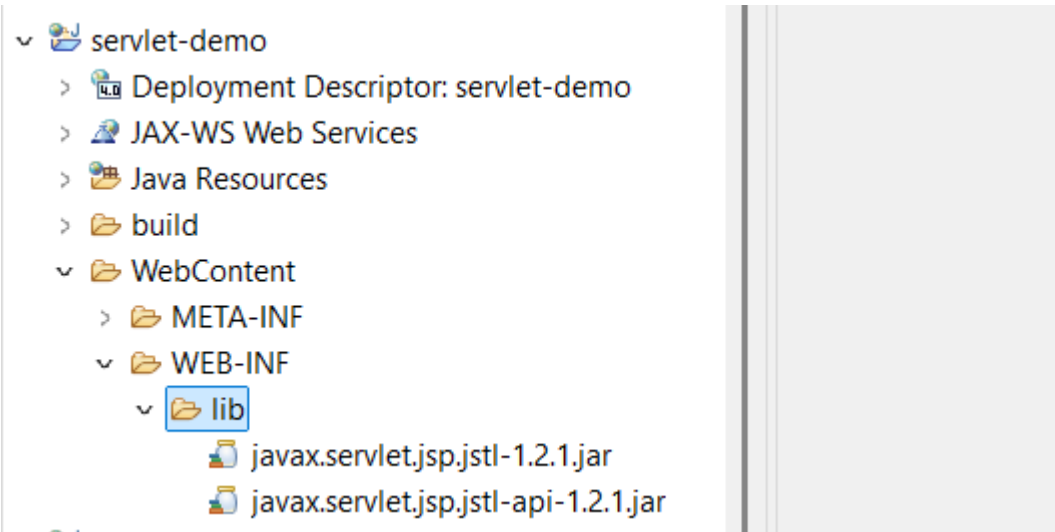


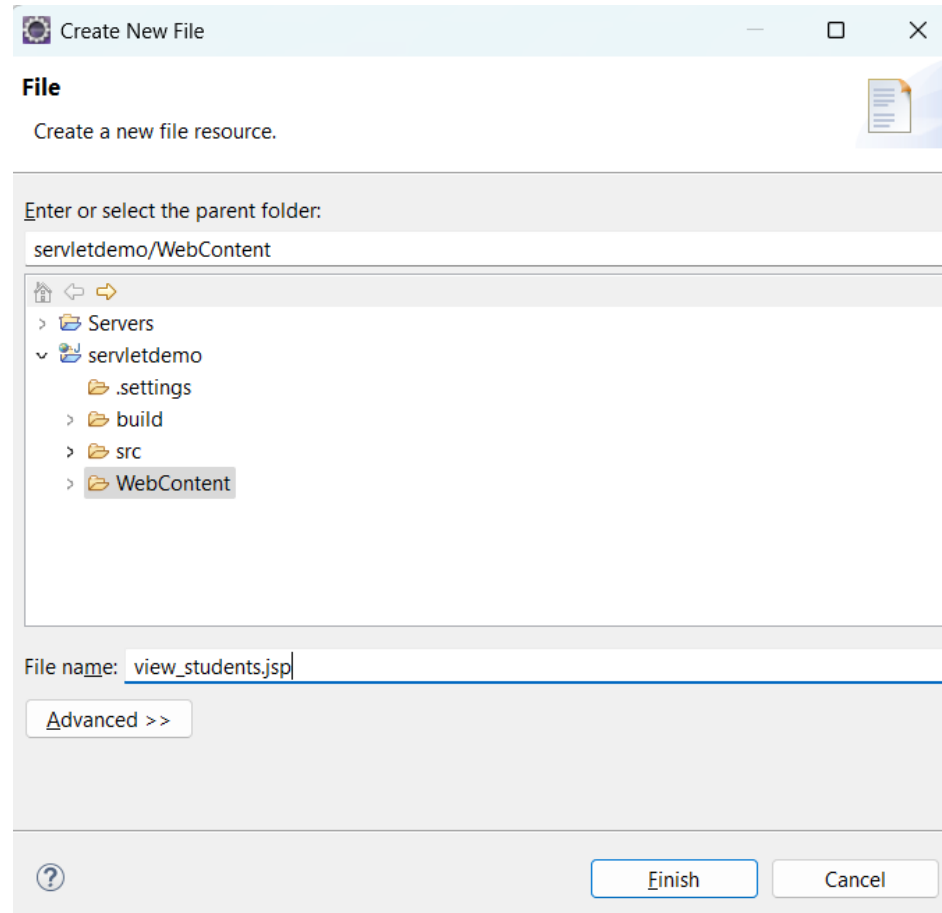


# Fixed

```
MvcDemoServlet.java ×
1 package com.example.servletdemo;
2
3 import jakarta.servlet.ServletException;
4 import jakarta.servlet.annotation.WebServlet;
5 import jakarta.servlet.http.HttpServlet;
6 import jakarta.servlet.http.HttpServletRequest;
7 import jakarta.servlet.http.HttpServletResponse;
8 import java.io.IOException;
9
10 /**
11  * Servlet implementation class MvcDemoServlet
12  */
13 @WebServlet("/MvcDemoServlet")
14 public class MvcDemoServlet extends HttpServlet {
15     private static final long serialVersionUID = 1L;
16
17     /**
18      * Default constructor.
19      */
20     public MvcDemoServlet() {
21         // TODO Auto-generated constructor stub
22     }
23
24     /**
25      * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
26      */
27     protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException
```

# Copy/Paste the jstl library files



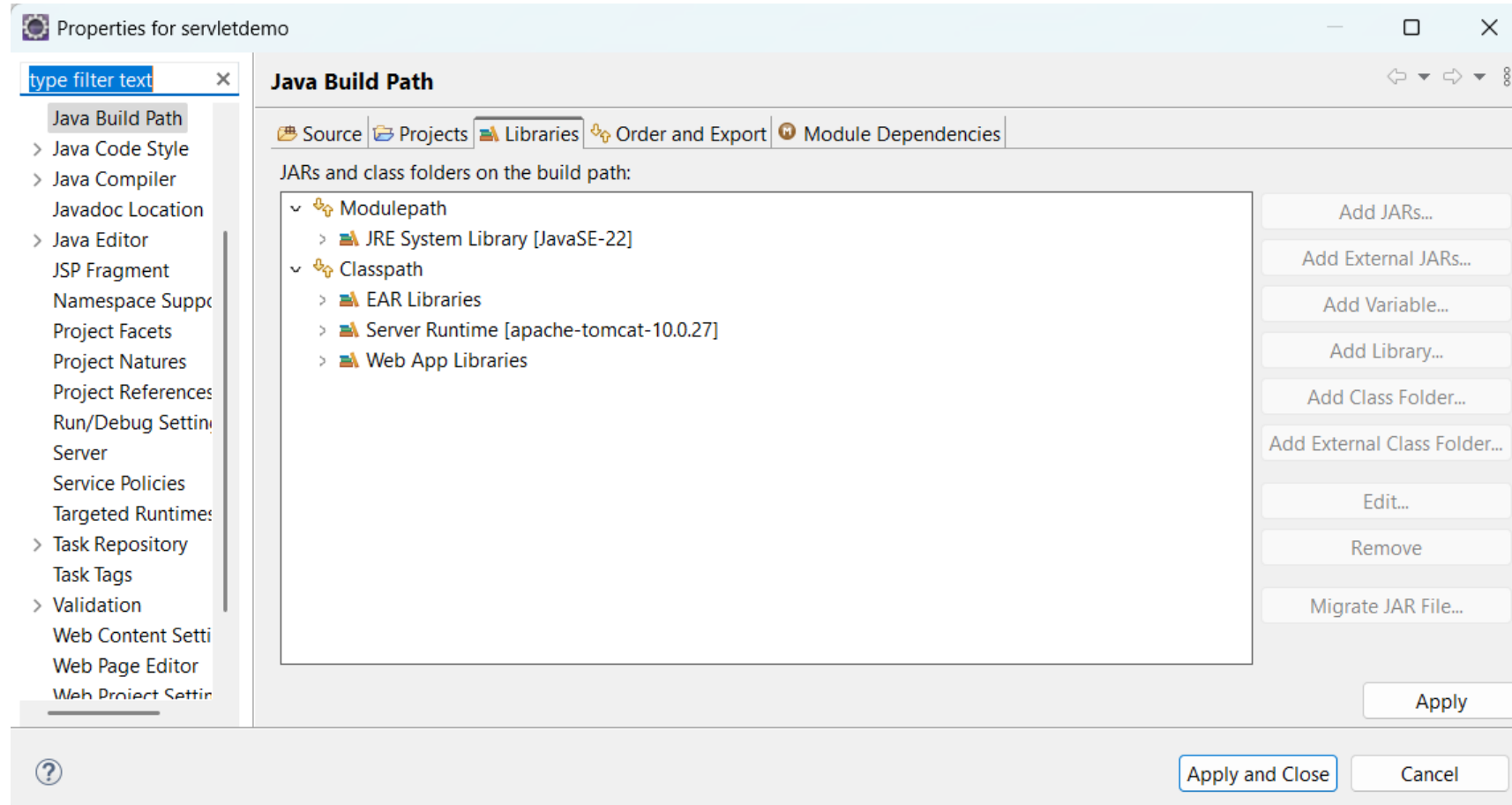


MvcDemoServlet.java view\_students.jsp ×

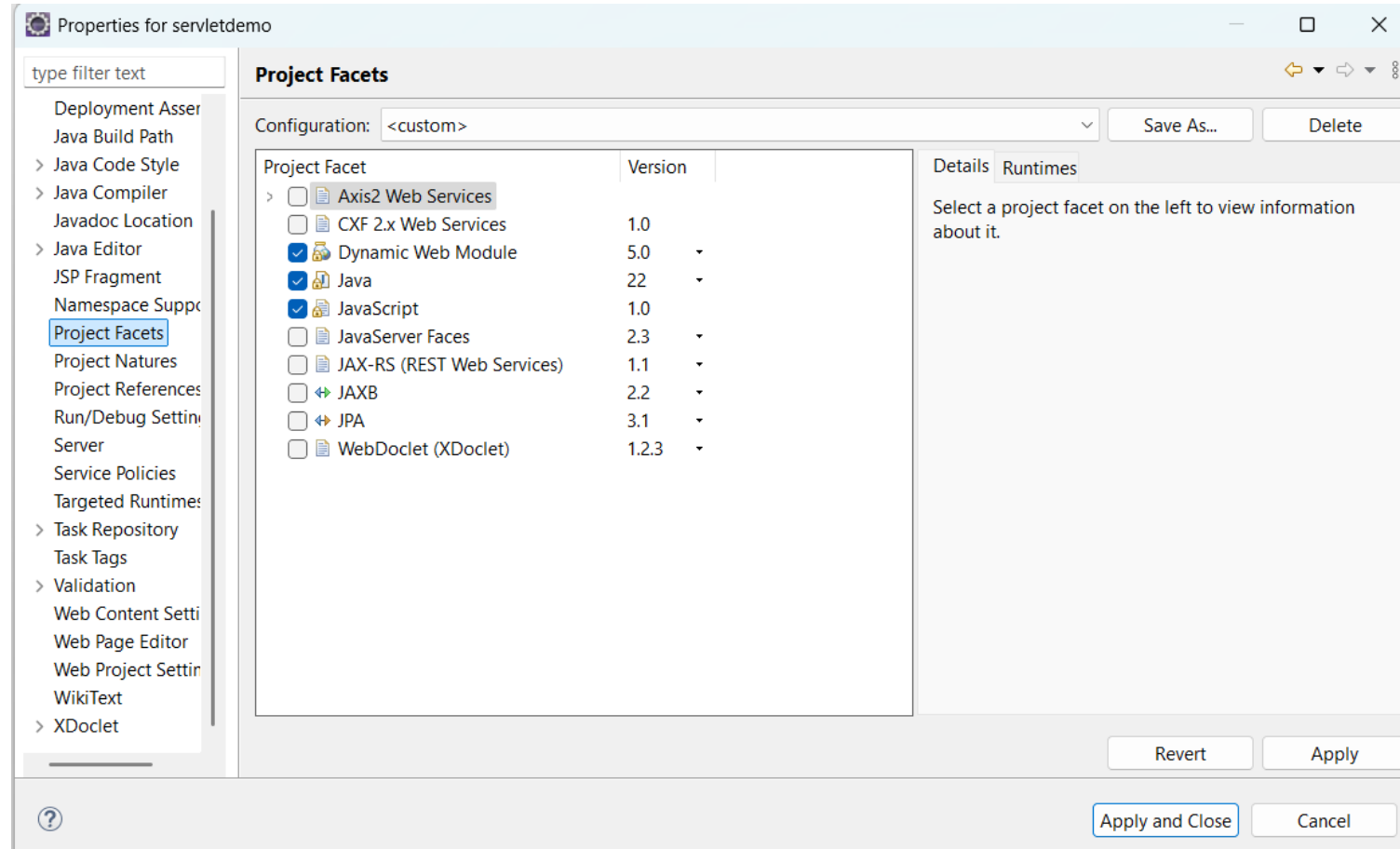
```
1 <%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
2
3 <html>
4 <body>
5
6     <c:forEach var="tempStudent" items="${student_list}">
7
8         ${tempStudent} <br/>
9
10    </c:forEach>
11
12 </body>
13 </html>
```



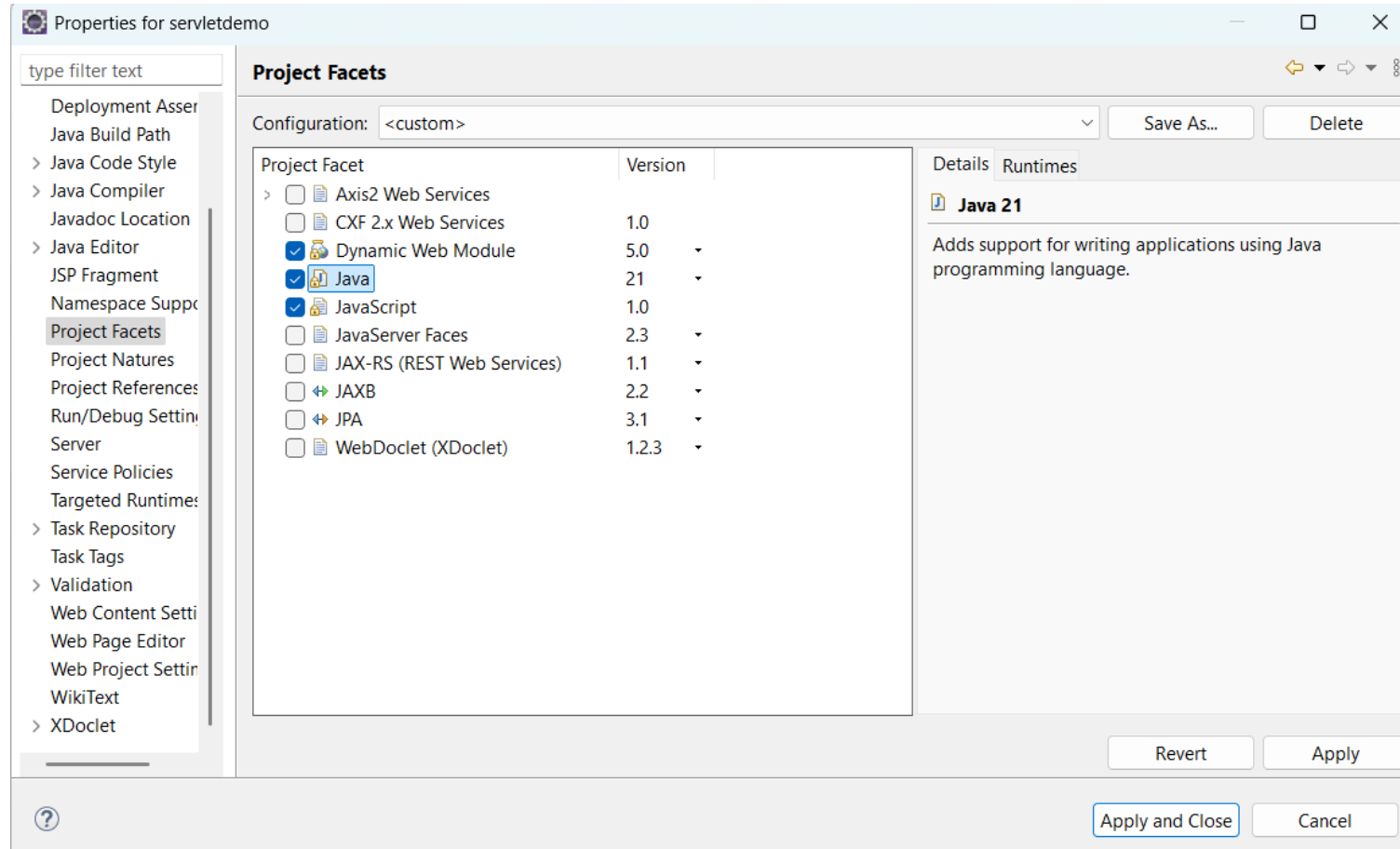
# Error: project facet java version 22 not supported



# Go to project facets



# Reduce the java version



Run On Server

Run On Server

Project facet Java version 21 is not supported.

How do you want to select the server?

☒ Choose an existing server

☐ Manually define a new server

Select the server that you want to use:

type filter text

Server	State
localhost	
Tomcat v10.0 Server at localhost	Stopped

Apache Tomcat v10.0 supports J2EE 1.2, 1.3, 1.4, and Java EE 5, 6, 7, 8 and Jakarta EE 9 Web modules.

Columns...

☐ Always use this server when running this project

?

< Back


Next >

Finish

Cancel



# Another error

    localhost:8080/servletdemo/MvcDemoServlet

---

**HTTP Status 500 – Internal Server Error**

---

**Type** Exception Report

**Message** java.lang.NoClassDefFoundError: javax/servlet/jsp/tagext/TagLibraryValidator

**Description** The server encountered an unexpected condition that prevented it from fulfilling the request.

**Exception**

```
jakarta.servlet.ServletException: java.lang.NoClassDefFoundError: javax/servlet/jsp/tagext/TagLibraryValidator
    org.apache.jasper.servlet.JspServlet.service(JspServlet.java:333)
    jakarta.servlet.http.HttpServlet.service(HttpServlet.java:792)
    org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:53)
    com.example.servletdemo.MvcDemoServlet.doGet(MvcDemoServlet.java:38)
    jakarta.servlet.http.HttpServlet.service(HttpServlet.java:683)
    jakarta.servlet.http.HttpServlet.service(HttpServlet.java:792)
    org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:53)
```

**Root Cause**

```
java.lang.NoClassDefFoundError: javax/servlet/jsp/tagext/TagLibraryValidator
    java.base/java.lang.ClassLoader.defineClass1(Native Method)
```

# Solution: add the following libraries

1. [jakarta.servlet.jsp.jstl-api-2.0.0.jar](#)

2. [jakarta.servlet.jsp.jstl-2.0.0 \(1\).jar](#)

Issue resolved