

Experimental Report

Development and Deployment of Java Servlets Using VS Code and Apache Tomcat



Student Name: Mohammad Abdullah

Student ID: 202322240356

Course: Java Web

Date: 13 November 2025

Instructor: D.cora

Department of Computer Science
China West Normal University
Academic Year: 2025

Contents

1	Objective	2
2	Environment Setup & Tools	3
3	Project Design and Implementation	4
3.1	Project Structure	4
3.2	pom.xml Configuration	4
3.3	ChineseServlet Implementation	6
3.4	ShowTimesServlet Implementation	8
3.5	Front-End (index.html)	13
4	Build and Deployment Procedure	14
4.1	Build Using Maven	14
4.2	Deploy to Tomcat	15
5	Testing and Results	16
6	Common Issues and Resolutions	17
7	Conclusion	18

CHAPTER 1

Objective

The primary objective of this experiment is to design, implement, and deploy two Java Servlets using Visual Studio Code, Apache Maven, and Apache Tomcat 11. The experiment demonstrates:

- UTF-8 based web content generation using Servlets.
- Visit-count persistence using ServletContext.
- Maven-based project management and Tomcat deployment.

CHAPTER 2

Environment Setup & Tools

Table 2.1: Software and Tools Used

Component	Version / Description
Operating System	Windows 11
IDE	Visual Studio Code 1.94+
Extensions	Extension Pack for Java, Community Server Connectors
JDK	Eclipse Adoptium JDK 17.0.15
Build Tool	Apache Maven 3.9.11
Application Server	Apache Tomcat 11.0.0-M6

Figure: Example of configured Tomcat Server view in VS Code.

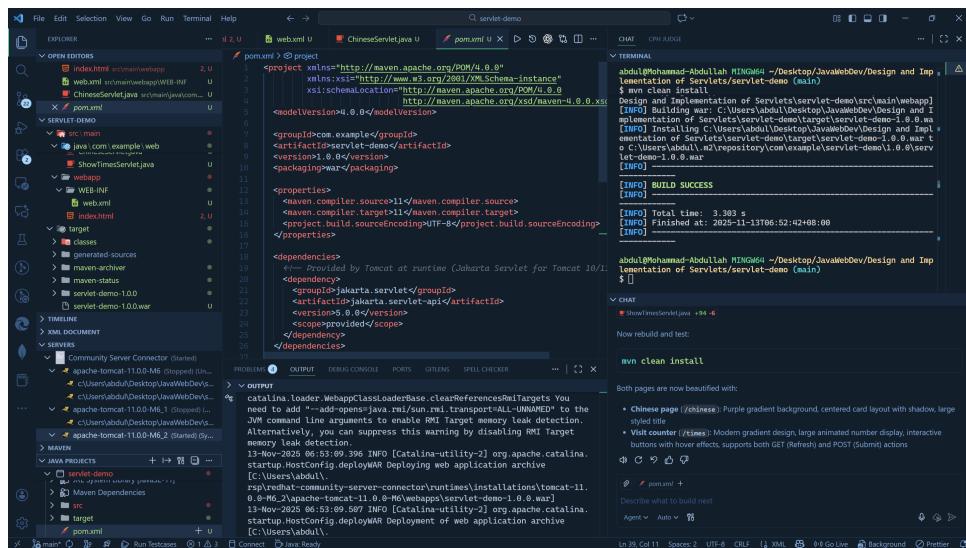


Figure 2.1: VS Code with Tomcat integration.

CHAPTER 3

Project Design and Implementation

3.1 Project Structure

The Maven Web Application was created using the `maven-archetype-webapp` template.

```
servlet-demo/
  src/
    main/
      java/
        com/example/web/
          ChineseServlet.java
          ShowTimesServlet.java
      webapp/
        WEB-INF/
          web.xml
          index.html
    pom.xml
  target/servlet-demo.war
```

3.2 pom.xml Configuration

Listing 3.1: pom.xml configuration for Jakarta Servlet API 5.0.0

```
1 <project xmlns="http://maven.apache.org/POM/4.0.0"
2   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3   xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
```



Figure 3.1: Project structure in VS Code.

```
4           http://maven.apache.org/xsd/maven-4.0.0.
          xsd">
5 <modelVersion>4.0.0</modelVersion>
6
7 <groupId>com.example</groupId>
8 <artifactId>servlet-demo</artifactId>
9 <version>1.0.0</version>
10 <packaging>war</packaging>
11
12 <properties>
13   <maven.compiler.source>11</maven.compiler.source>
14   <maven.compiler.target>11</maven.compiler.target>
15   <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
16 </properties>
17
18 <dependencies>
```

```

19      <!-- Provided by Tomcat at runtime (Jakarta Servlet for Tomcat
20          10/11+) -->
21
22  <dependency>
23    <groupId>jakarta.servlet</groupId>
24    <artifactId>jakarta.servlet-api</artifactId>
25    <version>5.0.0</version>
26    <scope>provided</scope>
27  </dependency>
28
29 </dependencies>
30
31
32 <build>
33   <plugins>
34     <plugin>
35       <artifactId>maven-war-plugin</artifactId>
36       <version>3.4.0</version>
37       <configuration>
38         <failOnMissingWebXml>false</failOnMissingWebXml>
39       </configuration>
40     </plugin>
41   </plugins>
42 </build>
43
44 </project>

```

3.3 ChineseServlet Implementation

Listing 3.2: ChineseServlet.java – UTF-8 text rendering in Servlet

```

1 package com.example.web;
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 jakarta.servlet.ServletOutputStream;
9 import java.io.IOException;
10 import java.nio.charset.StandardCharsets;
11
12 @WebServlet(name = "ChineseServlet", urlPatterns = "/chinese")
13 public class ChineseServlet extends HttpServlet {
14
15     @Override
16     protected void doGet(HttpServletRequest request,
17                           HttpServletResponse response)
18             throws ServletException, IOException {

```

```
18
19     // Set browser encoding to UTF-8
20     response.setContentType("text/html; charset=UTF-8");
21     response.setCharacterEncoding("UTF-8");
22
23     String text = "    J   a   v   a   W   e   b       ";
24
25     // Use output stream and write UTF-8 bytes
26     try (ServletOutputStream out = response.getOutputStream()) {
27         String html = "<!DOCTYPE html>\n" +
28             "<html lang='zh-CN'>\n" +
29             "<head>\n" +
30             "    <meta charset='UTF-8'>\n" +
31             "    <meta name='viewport' content='width=device-
32             width, initial-scale=1.0'>\n" +
33             "    <title>           - Java Web -</
34             title>\n" +
35             "    <style>\n" +
36             "        * { margin: 0; padding: 0; box-sizing:
37             border-box; }\n" +
38             "        body {\n" +
39             "            font-family: 'Microsoft YaHei', ,
40             "            , Arial, sans-serif;\n" +
41             "            background: linear-gradient(135deg,
42             "#667eea 0%, #764ba2 100%);\n" +
43             "            min-height: 100vh;\n" +
44             "            display: flex;\n" +
45             "            justify-content: center;\n" +
46             "            align-items: center;\n" +
47             "            padding: 20px;\n" +
48             "        }\n" +
49             ".container {\n" +
50             "            background: white;\n" +
51             "            border-radius: 12px;\n" +
52             "            box-shadow: 0 10px 40px rgba(0, 0, 0,
53             "            0.2);\n" +
54             "            padding: 60px 40px;\n" +
55             "            text-align: center;\n" +
56             "            max-width: 500px;\n" +
57             "        }\n" +
58             "h1 {\n" +
59             "    color: #333;\n" +
60             "    font-size: 2.5em;\n" +
61             "    margin-bottom: 20px;\n" +
62             "    text-shadow: 2px 2px 4px rgba(0, 0, 0,
63             "    0.1);\n" +
```

```
57         "      }\n58         ".badge +\n59             "      display: inline-block;\n" +\n60             "      background: linear-gradient(135deg,\n" +\n61                 "#667eeaa 0%, #764ba2 100%);\n" +\n62                 "      color: white;\n" +\n63                 "      padding: 10px 20px;\n" +\n64                 "      border-radius: 50px;\n" +\n65                 "      font-size: 0.9em;\n" +\n66                 "      margin-top: 15px;\n" +\n67             "}\n68     "
```

```
"</style>\n" +\n"</head>\n" +\n"<body>\n" +\n"    <div class='container'>\n" +\n"        <h1>" + text + "</h1>\n" +\n"        <div class='badge'>\n" +\n"            </div>\n" +\n"    </div>\n" +\n"    </body>\n" +\n"    </html>";\n\n    out.write(html.getBytes(StandardCharsets.UTF_8));\n}\n}\n}
```

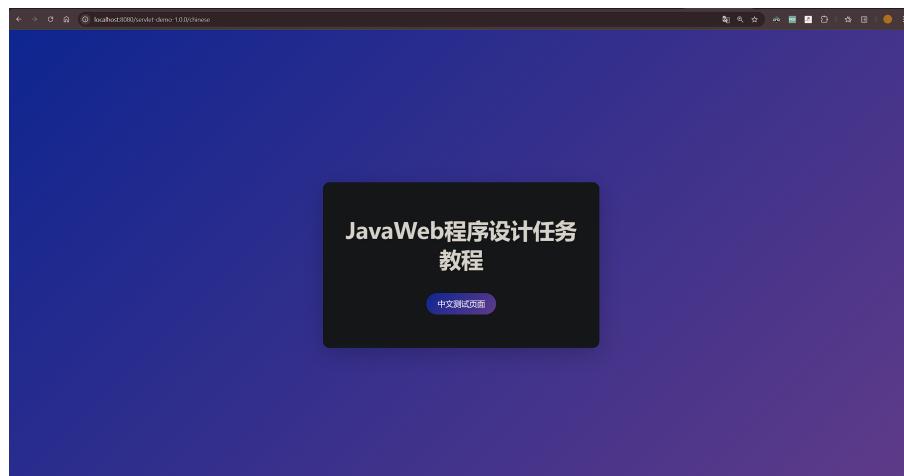


Figure 3.2: Browser output of ChineseServlet displaying UTF-8 text.

3.4 ShowTimesServlet Implementation

Listing 3.3: ShowTimesServlet.java – Counting website visits using ServletContext

```
1 package com.example.web;
2
3 import jakarta.servlet.ServletContext;
4 import jakarta.servlet.ServletException;
5 import jakarta.servlet.annotation.WebServlet;
6 import jakarta.servlet.http.HttpServlet;
7 import jakarta.servlet.http.HttpServletRequest;
8 import jakarta.servlet.http.HttpServletResponse;
9 import java.io.IOException;
10 import java.io.PrintWriter;
11 import java.util.concurrent.atomic.AtomicLong;
12
13 @WebServlet(name = "ShowTimesServlet", urlPatterns = "/times")
14 public class ShowTimesServlet extends HttpServlet {
15
16     private static final String ATTR = "visitCount";
17
18     @Override
19     public void init() throws ServletException {
20         // Initialize the counter once per webapp
21         ServletContext ctx = getServletContext();
22         synchronized (ctx) {
23             if (ctx.getAttribute(ATTR) == null) {
24                 ctx.setAttribute(ATTR, new AtomicLong(0));
25             }
26         }
27     }
28
29     @Override
30     protected void doPost(HttpServletRequest request,
31                          HttpServletResponse response)
32             throws ServletException, IOException {
33         // Delegate to doGet
34         doGet(request, response);
35     }
36
37     @Override
38     protected void doGet(HttpServletRequest request,
39                          HttpServletResponse response)
40             throws IOException {
41         response.setContentType("text/html; charset=UTF-8");
42         response.setCharacterEncoding("UTF-8");
43
44         ServletContext ctx = getServletContext();
45         AtomicLong counter;
46         synchronized (ctx) {
```

```
45     counter = (AtomicLong) ctx.getAttribute(ATTR);
46     if (counter == null) { // safety in case the context was
47         reloaded
48         counter = new AtomicLong(0);
49         ctx.setAttribute(ATTR, counter);
50     }
51
52     long visits = counter.incrementAndGet(); // increase by 1 each
53     request
54
55     try (PrintWriter out = response.getWriter()) {
56         String html = "<!DOCTYPE html>\n" +
57             "<html lang='en'>\n" +
58             "    <head>\n" +
59             "        <meta charset='UTF-8'>\n" +
60             "        <meta name='viewport' content='width=device-
61             width, initial-scale=1.0'>\n" +
62             "        <title>Visit Counter</title>\n" +
63             "        <style>\n" +
64             "            * { margin: 0; padding: 0; box-sizing:
65             border-box; }\n" +
66             "            body {\n" +
67             "                font-family: 'Segoe UI', Tahoma,
68             Geneva, Verdana, sans-serif;\n" +
69             "                background: linear-gradient(135deg,
70             #667eea 0%, #764ba2 100%);\n" +
71             "                min-height: 100vh;\n" +
72             "                display: flex;\n" +
73             "                justify-content: center;\n" +
74             "                align-items: center;\n" +
75             "                padding: 20px;\n" +
76             "            }\n" +
77             ".container {\n" +
78             "                background: white;\n" +
79             "                border-radius: 12px;\n" +
80             "                box-shadow: 0 10px 40px rgba(0, 0, 0,
81             0.2);\n" +
82             "                padding: 60px 40px;\n" +
83             "                text-align: center;\n" +
84             "                max-width: 500px;\n" +
85             "            }\n" +
86             "h1 {\n" +
87             "    color: #333;\n" +
88             "    font-size: 2.5em;\n" +
89             "    margin-bottom: 10px;\n" +
```

```
84          "      }\n" +\n85          "      .visit-count {\n" +\n86              "          font-size: 3.5em;\n" +\n87              "          background: linear-gradient(135deg,\n" +\n88                  "#667eea 0%, #764ba2 100%);\n" +\n89                  "          -webkit-background-clip: text;\n" +\n90                  "          -webkit-text-fill-color: transparent;\n" +\n91          "      }\n" +\n92          "      background-clip: text;\n" +\n93          "      font-weight: bold;\n" +\n94          "      margin: 20px 0;\n" +\n95          "  }\n" +\n96          "  p {\n" +\n97              "      color: #666;\n" +\n98              "      font-size: 1.1em;\n" +\n99              "      margin-bottom: 30px;\n" +\n100             "      line-height: 1.6;\n" +\n101            "  }\n" +\n102            ".button-group {\n" +\n103                "      display: flex;\n" +\n104                "      gap: 15px;\n" +\n105                "      justify-content: center;\n" +\n106            "  }\n" +\n107            "  button {\n" +\n108                "      padding: 12px 30px;\n" +\n109                "      font-size: 1em;\n" +\n110                "      border: none;\n" +\n111                "      border-radius: 50px;\n" +\n112                "      cursor: pointer;\n" +\n113                "      font-weight: bold;\n" +\n114                "      transition: all 0.3s ease;\n" +\n115            "  }\n" +\n116            "  .btn-refresh {\n" +\n117                "      background: linear-gradient(135deg,\n" +\n118                    "#667eea 0%, #764ba2 100%);\n" +\n119                    "      color: white;\n" +\n120                    "  }\n" +\n121                    "  .btn-refresh:hover {\n" +\n122                        "      transform: translateY(-2px);\n" +\n123                        "      box-shadow: 0 5px 15px rgba(102, 126,\n" +\n124                            " 234, 0.4);\n" +\n125                        "  }\n" +\n126                        "  .btn-post {\n" +\n127                            "      background: #f0f0f0;\n" +\n128                            "      color: #333;\n" +\n129                            "      border: 2px solid #667eea;\n" +\n130                            "  }
```

```
126          "      }\n" +\n127          "      .btn-post:hover {\n" +\n128              "          background: #667eea;\n" +\n129              "          color: white;\n" +\n130              "          transform: translateY(-2px);\n" +\n131          "      }\n" +\n132          "  </style>\n" +\n133      "
```

```
"</head>\n" +\n134      "<body>\n" +\n135      "  <div class='container'>\n" +\n136          "    <h1>      Website Visits</h1>\n" +\n137          "    <div class='visit-count'>" + visits + "</\n" +\n138              "        <p>Click the button below to increase the\n" +\n139              "        visit counter</p>\n" +\n140              "        <div class='button-group'>\n" +\n141                  "            <button class='btn-refresh' onclick='\n" +\n142                      "                location.reload()\n" +\n143                          "            '>&nbsp; Refresh</button>\n" +\n144                          "            <form method='POST' style='display:\n" +\n145                              " inline;'\n" +\n146                              "            <button type='submit' class='btn-\n" +\n147                                  " post'>      Submit (POST)</button>\n" +\n148                          "            </form>\n" +\n149          "        </div>\n" +\n150      "
```

```
"</body>\n" +\n151      "</html>";\n\n        out.print(html);\n    }\n}\n}
```

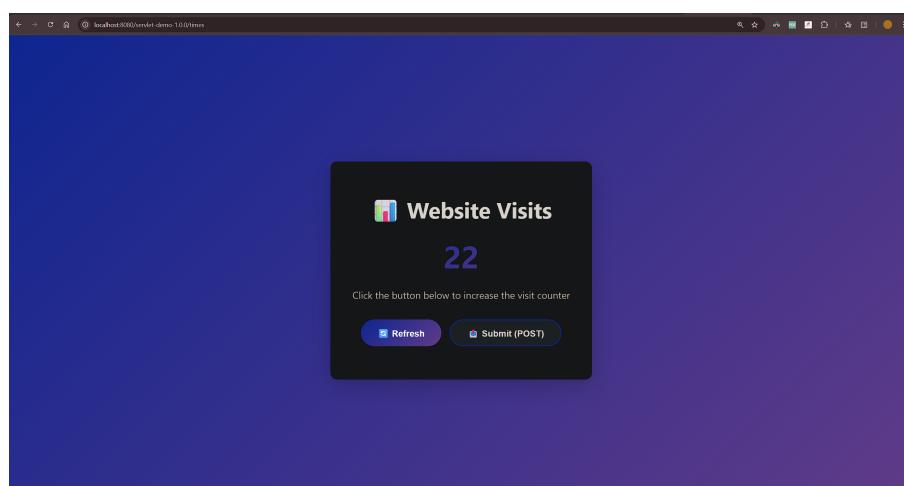


Figure 3.3: Browser output of ShowTimesServlet with visit counter.

3.5 Front-End (index.html)

Listing 3.4: index.html linking to both servlets

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8" />
5     <title>Servlet Demo</title>
6   </head>
7   <body style="font-family: sans-serif">
8     <h2>Servlet Links</h2>
9     <ul>
10       <li><a href="chinese">ChineseServlet (/chinese)</a></li>
11       <li><a href="times">ShowTimesServlet (/times)</a></li>
12     </ul>
13   </body>
14 </html>
```

CHAPTER 4

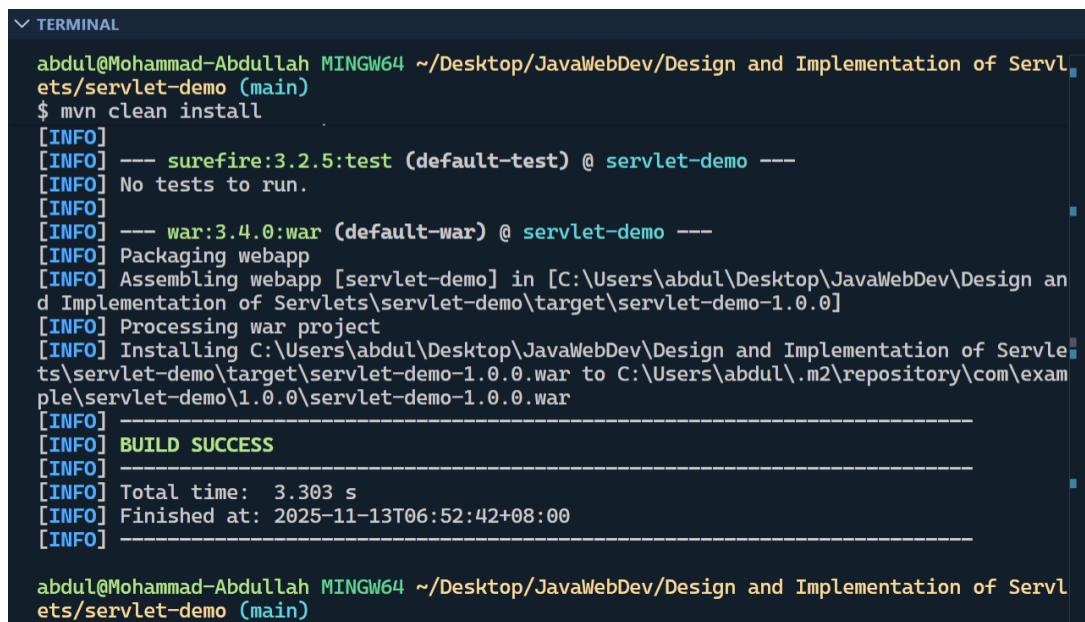
Build and Deployment Procedure

4.1 Build Using Maven

Run the following in Command Prompt:

```
1 cd C:\Users\Mohammad\Documents\servlet-demo  
2 mvn clean install
```

The WAR file is generated at: target/servlet-demo.war



A screenshot of a terminal window titled "TERMINAL". The command "mvn clean install" is run in a directory named "servlet-demo". The output shows the Maven build process, including the execution of tests (surefire), the creation of a WAR file (war:3.4.0:war), and the final "BUILD SUCCESS" message. The terminal also displays the total time taken and the finished date and time.

```
abduL@Mohammad-Abdullah MINGW64 ~/Desktop/JavaWebDev/Design and Implementation of Servlets/servlet-demo (main)  
$ mvn clean install  
[INFO] [INFO] --- surefire:3.2.5:test (default-test) @ servlet-demo ---  
[INFO] No tests to run.  
[INFO]  
[INFO] --- war:3.4.0:war (default-war) @ servlet-demo ---  
[INFO] Packaging webapp  
[INFO] Assembling webapp [servlet-demo] in [C:\Users\abdul\Desktop\JavaWebDev\Design and Implementation of Servlets\servlet-demo\target\servlet-demo-1.0.0]  
[INFO] Processing war project  
[INFO] Installing C:\Users\abdul\Desktop\JavaWebDev\Design and Implementation of Servlets\servlet-demo\target\servlet-demo-1.0.0.war to C:\Users\abdul\.m2\repository\com\example\servlet-demo\1.0.0\servlet-demo-1.0.0.war  
[INFO] -----  
[INFO] BUILD SUCCESS  
[INFO] -----  
[INFO] Total time: 3.303 s  
[INFO] Finished at: 2025-11-13T06:52:42+08:00  
[INFO] -----  
abduL@Mohammad-Abdullah MINGW64 ~/Desktop/JavaWebDev/Design and Implementation of Servlets/servlet-demo (main)
```

Figure 4.1: Maven build showing "BUILD SUCCESS" message.

4.2 Deploy to Tomcat

Copy the WAR file to: C:-tomcat-11.0.0-M6

Then start the server:

```
1 cd C:\apache-tomcat-11.0.0-M6\bin  
2 startup.bat
```

URLs:

- Base: <http://localhost:8080/servlet-demo/>
- ChineseServlet: <http://localhost:8080/servlet-demo/chinese>
- ShowTimesServlet: <http://localhost:8080/servlet-demo/times>

CHAPTER 5

Testing and Results

Table 5.1: Servlet Testing Summary

Servlet	Functionality	Result
ChineseServlet	UTF-8 Text Rendering	Displays "JavaWeb"
ShowTimesServlet	Visit Counter	Increments on each request
index.html	Navigation Page	Links to both servlets

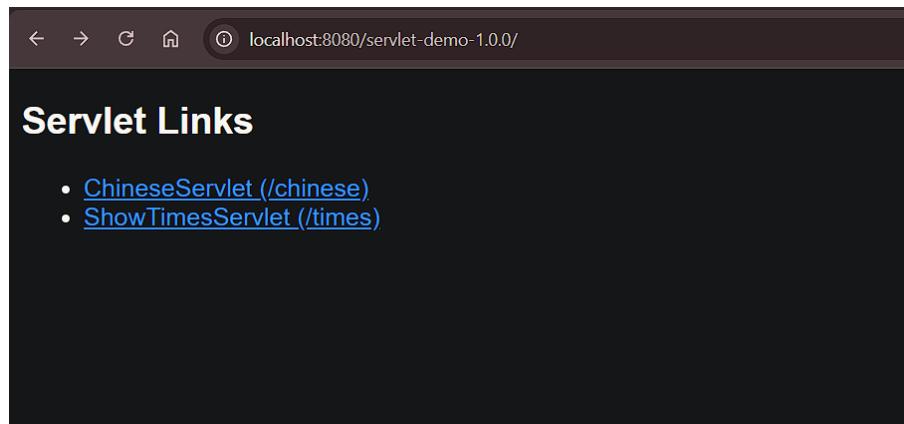


Figure 5.1: Index page providing servlet navigation.

CHAPTER 6

Common Issues and Resolutions

- **HTTP 404 Error:** Incorrect context path; fixed by using `/servlet-demo/`.
- **UTF-8 not displayed:** Added `setCharacterEncoding("UTF-8")` in response.
- **Counter reset after restart:** Stored counter in `ServletContext`.
- **Annotation not working:** Updated to Tomcat 11 (Jakarta EE 5 compatible).

CHAPTER 7

Conclusion

This experiment successfully demonstrated Java Servlet development using VS Code, Maven, and Tomcat 11. Both servlets performed as expected:

- **ChineseServlet** confirmed UTF-8 handling and dynamic HTML generation.
- **ShowTimesServlet** showed application-level state persistence using `ServletContext`.

The environment is now fully operational for further Jakarta EE web development.