Lesson 9 Tomcat

ToDoListServlet.java

```
package CenModule9;

import jakarta.servlet.ServletException;
import jakarta.servlet.annotation.WebServlet;
import jakarta.servlet.http.HttpServlet;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;
import java.io.IOException;
import java.io.PrintWriter;
import java.util.ArrayList;

/**
 * Servlet implementation class ToDoListServlet
 */
@WebServlet("/ToDoListServlet")
public class ToDoListServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    private ArrayList<String> items;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public ToDoListServlet() {
        super();
        items = new ArrayList<>();
    }

    protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        response.getWriter().append("Served at:
").append(request.getContextPath());
    }

    protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        String command = request.getParameter("command");

        if (command != null && !command.isEmpty()) {
            if (command.equalsIgnoreCase("add")) {
                String item = request.getParameter("item");
                addItem(item);
            } else if (command.equalsIgnoreCase("delete")) {
                int index = Integer.parseInt(request.getParameter("index"));
                deleteItem(index - 1);
            }
        }

        // Set the items list as a request attribute
        request.setAttribute("items", items);
```

```
        // Forward the request to the index.html file
        request.getRequestDispatcher("index.html").forward(request,
response);
    }

    /**
     * Adds a to-do item to the list.
     *
     * @param item The to-do item to be added.
     */
    public void addItem(String item) {
        items.add(item);
    }

    /**
     * Deletes a to-do item from the list based on its index number.
     *
     * @param index The index of the item to be deleted.
     */
    public void deleteItem(int index) {
        if (index >= 0 && index < items.size()) {
            items.remove(index);
        }
    }
}
```

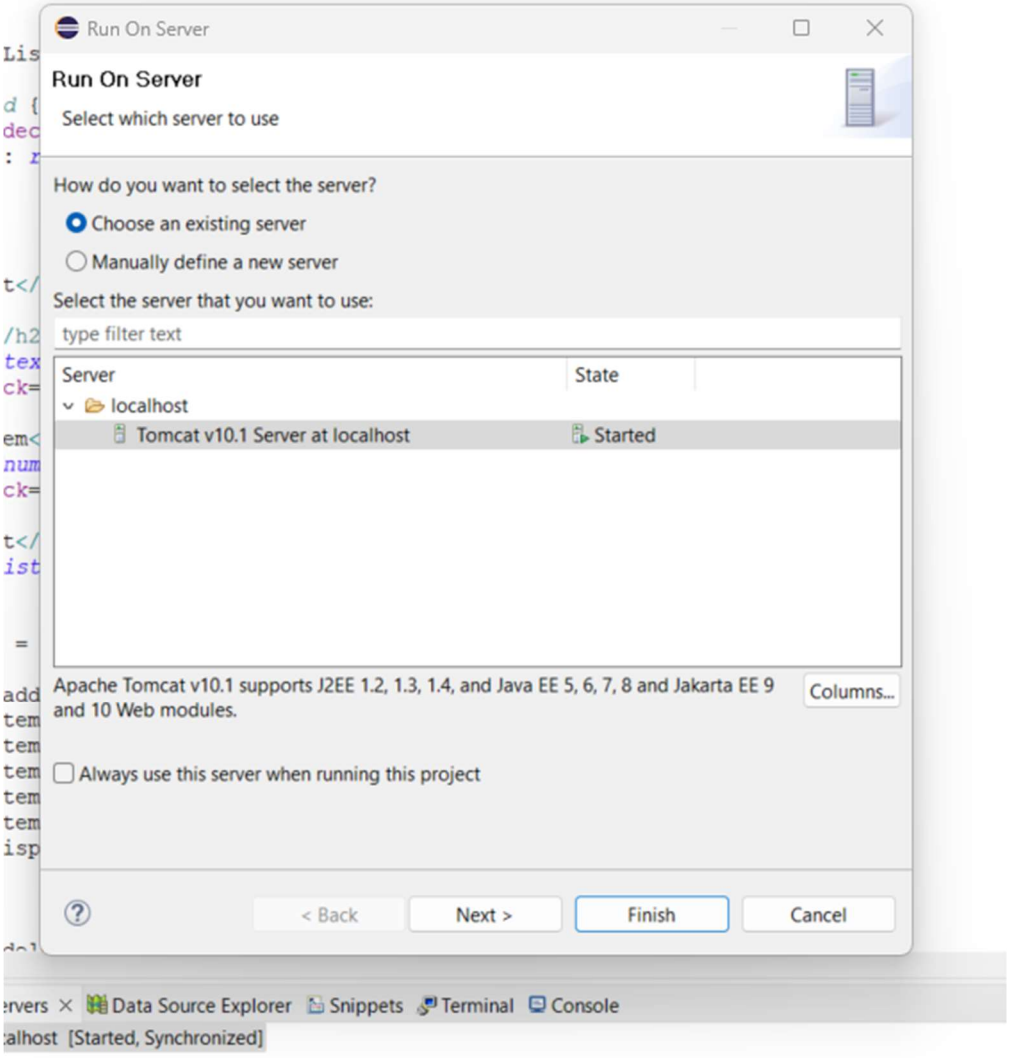
Index.html

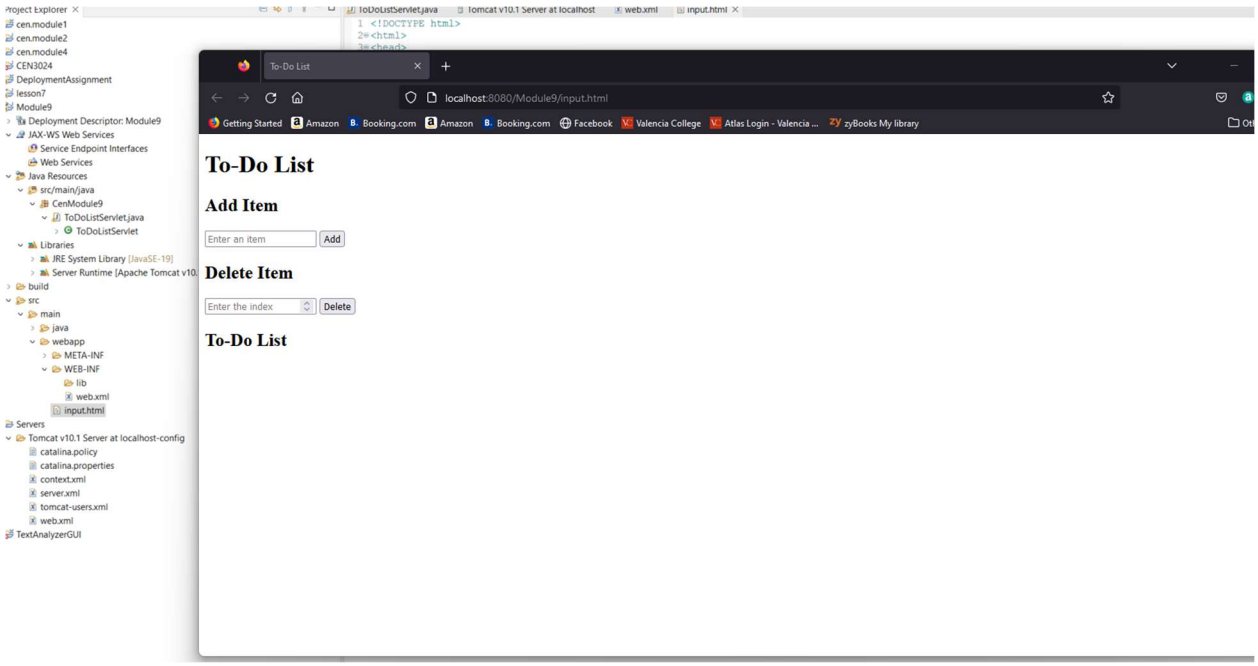
```
<!DOCTYPE html>
<html>
<head>
  <title>To-Do List</title>
  <style>
    .completed {
      text-decoration: line-through;
      color: red;
    }
  </style>
</head>
<body>
  <h1>To-Do List</h1>
```

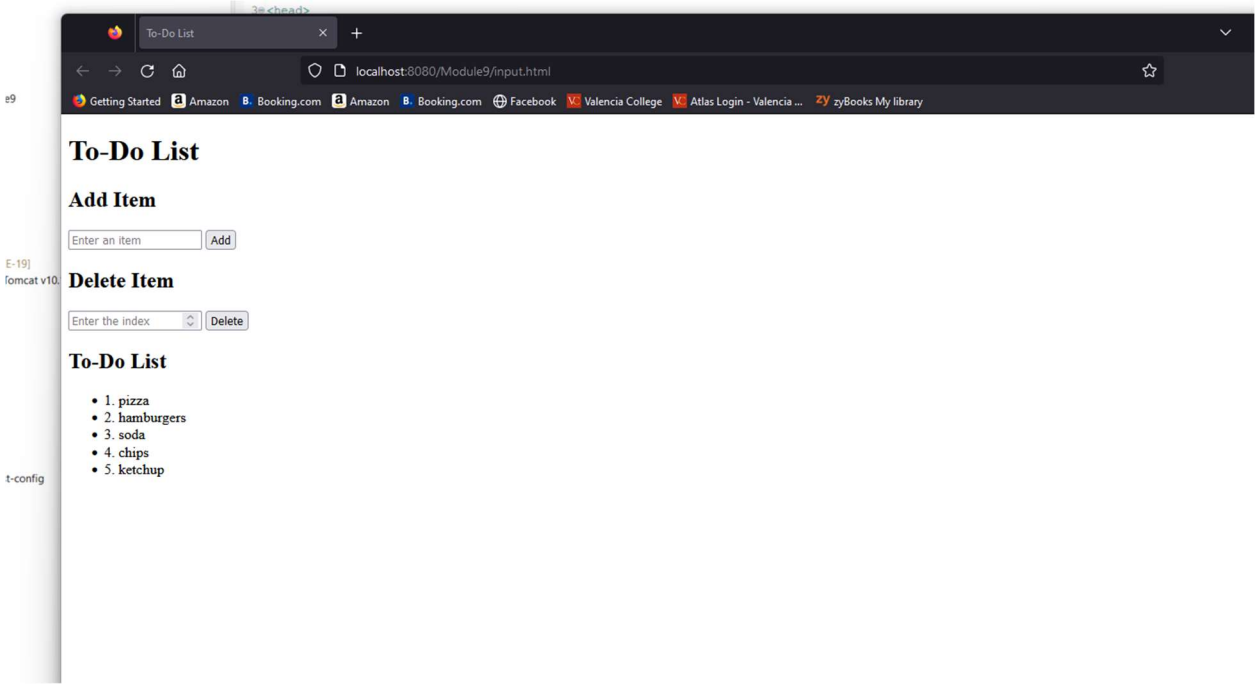
```
<h2>Add Item</h2>
<input type="text" id="itemInput" placeholder="Enter an item">
<button onclick="addItem()">Add</button>

<h2>Delete Item</h2>
<input type="number" id="indexInput" placeholder="Enter the index">
<button onclick="deleteItem()">Delete</button>

<h2>To-Do List</h2>
<ul id="todoList"></ul>

<script>
  var items = [];

  function addItem() {
    var itemInput = document.getElementById("itemInput");
    var item = itemInput.value;
    if (item.trim() !== "") {
      items.push({ text: item, completed: false });
      itemInput.value = "";
      displayList();
    }
  }

  function deleteItem() {
    var indexInput = document.getElementById("indexInput");
    var index = parseInt(indexInput.value) - 1;
    if (index >= 0 && index < items.length) {
      items[index].completed = true;
```

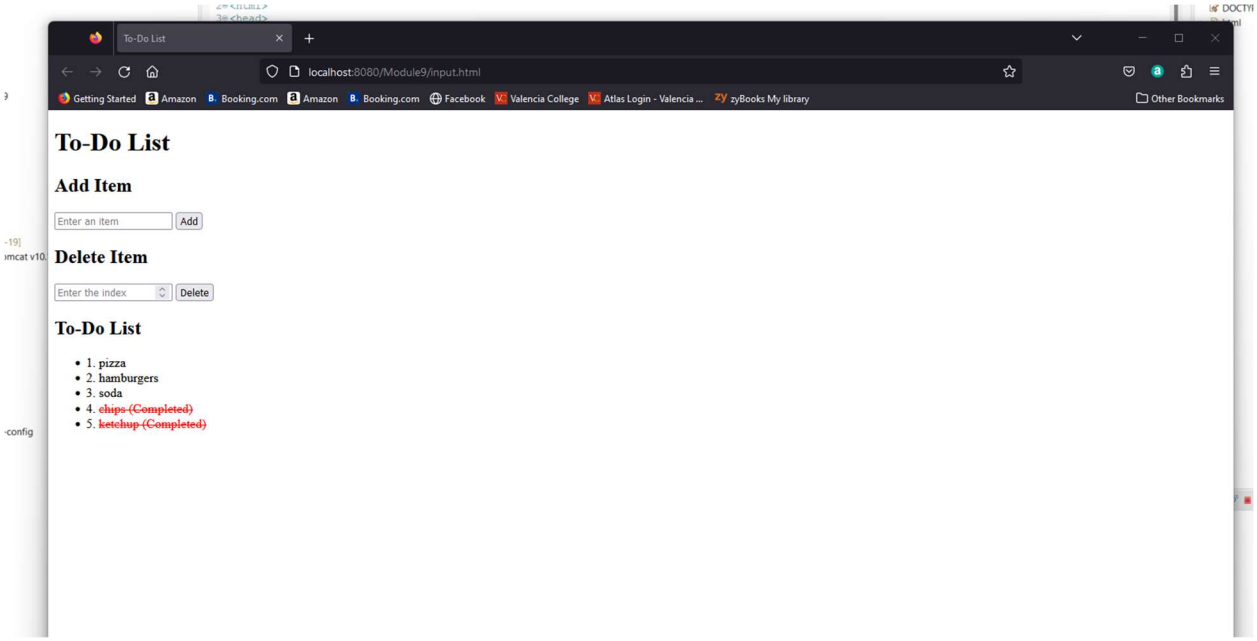
```
        indexInput.value = "";
        displayList();
    }
}

function displayList() {
    var todoList = document.getElementById("todoList");
    todoList.innerHTML = "";

    if (items.length === 0) {
        var emptyMessage = document.createElement("li");
        emptyMessage.textContent = "Your To-Do list is empty! Go relax and have fun!!!";
        todoList.appendChild(emptyMessage);
    } else {
        for (var i = 0; i < items.length; i++) {
            var listItem = document.createElement("li");
            listItem.textContent = (i + 1) + ". ";

            var itemText = document.createElement("span");
            if (items[i].completed) {
                itemText.classList.add("completed");
                itemText.textContent = items[i].text + " (Completed)";
            } else {
                itemText.textContent = items[i].text;
            }
            listItem.appendChild(itemText);

            todoList.appendChild(listItem);
        }
```

```
        }
      }
  </script>
</body>
</html>
```




Project Explorer
cen.module1
cen.module2
cen.module4
CEN3024
DeploymentAssignment
lesson7
Module9
Deployment Descriptor: Module9
JAX-WS Web Services
Service Endpoint Interfaces
Web Services
Java Resources
src/main/java
CenModule9
ToDoListServlet.java
ToDoListServlet
Libraries
JRE System Library [JavaSE-19]
Server Runtime [Apache Tomcat v10.
build
src
main
java
webapp
META-INF
WEB-INF
lib
web.xml
input.html
Servers
Tomcat v10.1 Server at localhost-config
catalina.policy
catalina.properties
context.xml
server.xml
tomcat-users.xml
web.xml
TextAnalyzerGUI
ToDoListServlet.java
Tomcat v10.1 Server at localhost
web.xml
input.html
1 <!DOCTYPE html>
2⊖<html>
3⊖<head>
To-Do List
localhost:8080/Module9/input.html
Getting Started
Amazon
Booking.com
Amazon
Booking.com
Facebook
Valencia College
Atlas Login - Valencia ...
zyBooks My library
To-Do List
Add Item
Enter an item
Add
Delete Item
Enter the index
Delete
To-Do List


3⊖<head>
To-Do List
localhost:8080/Module9/input.html
Getting Started
Amazon
Booking.com
Amazon
Booking.com
Facebook
Valencia College
Atlas Login - Valencia ...
zyBooks My library
To-Do List
Add Item
Enter an item
Add
Delete Item
Enter the index
Delete
To-Do List
1. pizza
2. hamburgers
3. soda
4. chips
5. ketchup


To-Do List
localhost:8080/Module9/input.html
Getting Started
Amazon
Booking.com
Amazon
Booking.com
Facebook
Valencia College
Atlas Login - Valencia ...
zyBooks My library
Other Bookmarks
To-Do List
Add Item
Enter an item
Add
Delete Item
Enter the index
Delete
To-Do List
1. pizza
2. hamburgers
3. soda
4. chips (Completed)
5. ketchup (Completed)