Ain Shams University – Faculty of Engineering I-CHEP

University of East London - Computer Engineering and Software Systems Program

Web Development
CSE361

Internet Programming
CESS5005

Lab Assignment

Submitted to:

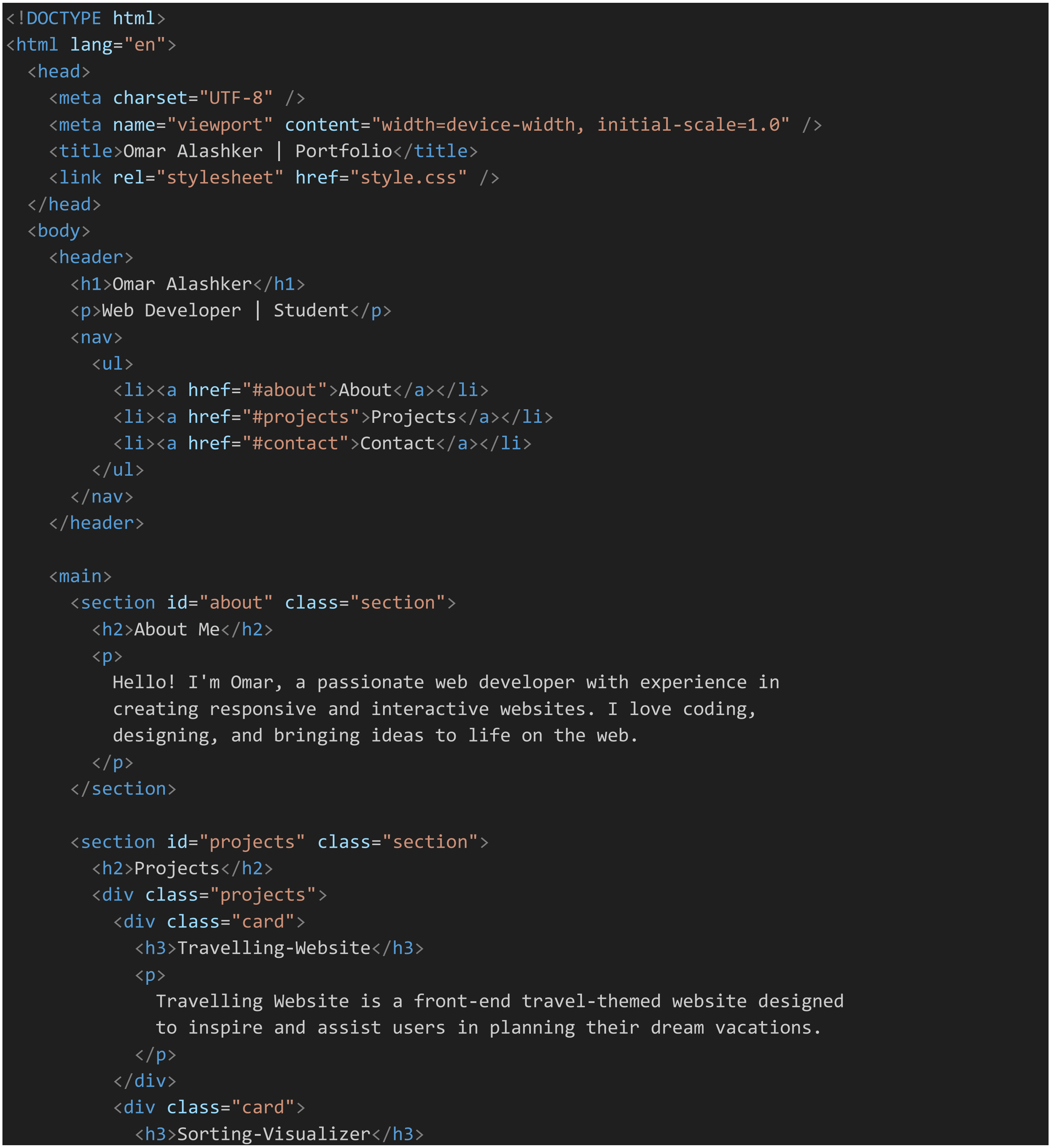
Dr. Islam Tharwat Abdel Halim

Date:
November 4, 2025

Prepared by: **Omar Yasser Alashker "23P0294"**

Lab 2 submission:

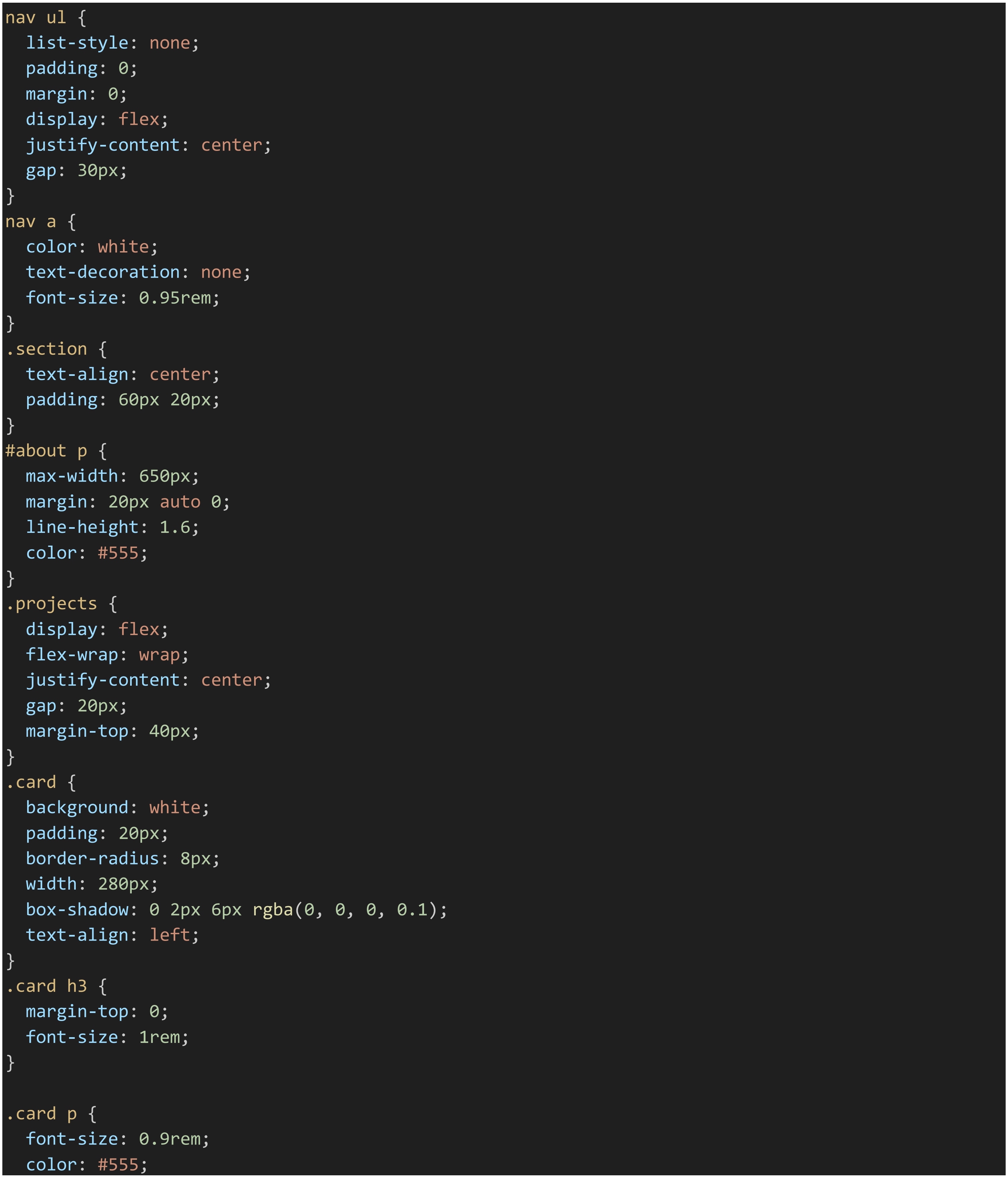
index.html:



```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Omar Alashker | Portfolio</title>
    <link rel="stylesheet" href="style.css" />
  </head>
  <body>
    <header>
      <h1>Omar Alashker</h1>
      <p>Web Developer | Student</p>
      <nav>
        <ul>
          <li><a href="#about">About</a></li>
          <li><a href="#projects">Projects</a></li>
          <li><a href="#contact">Contact</a></li>
        </ul>
      </nav>
    </header>

    <main>
      <section id="about" class="section">
        <h2>About Me</h2>
        <p>
          Hello! I'm Omar, a passionate web developer with experience in
          creating responsive and interactive websites. I love coding,
          designing, and bringing ideas to life on the web.
        </p>
      </section>

      <section id="projects" class="section">
        <h2>Projects</h2>
        <div class="projects">
          <div class="card">
            <h3>Travelling-Website</h3>
            <p>
              Travelling Website is a front-end travel-themed website designed
              to inspire and assist users in planning their dream vacations.
            </p>
          </div>
          <div class="card">
            <h3>Sorting-Visualizer</h3>
            <p>
              A sorting viusalizer that includes sorting algorithms such as
              "Bubble Sort", "Selection Sort", "Insertion Sort", "Merge Sort",
              and "Quick Sort".
            </p>
          </div>
          <div class="card">
            <h3>Ecommerce_DB</h3>
            <p>
              This project contains the full relational database schema for a
              fictional e-commerce platform.
            </p>
          </div>
        </div>
      </section>
    </main>
  </body>
</html>
```

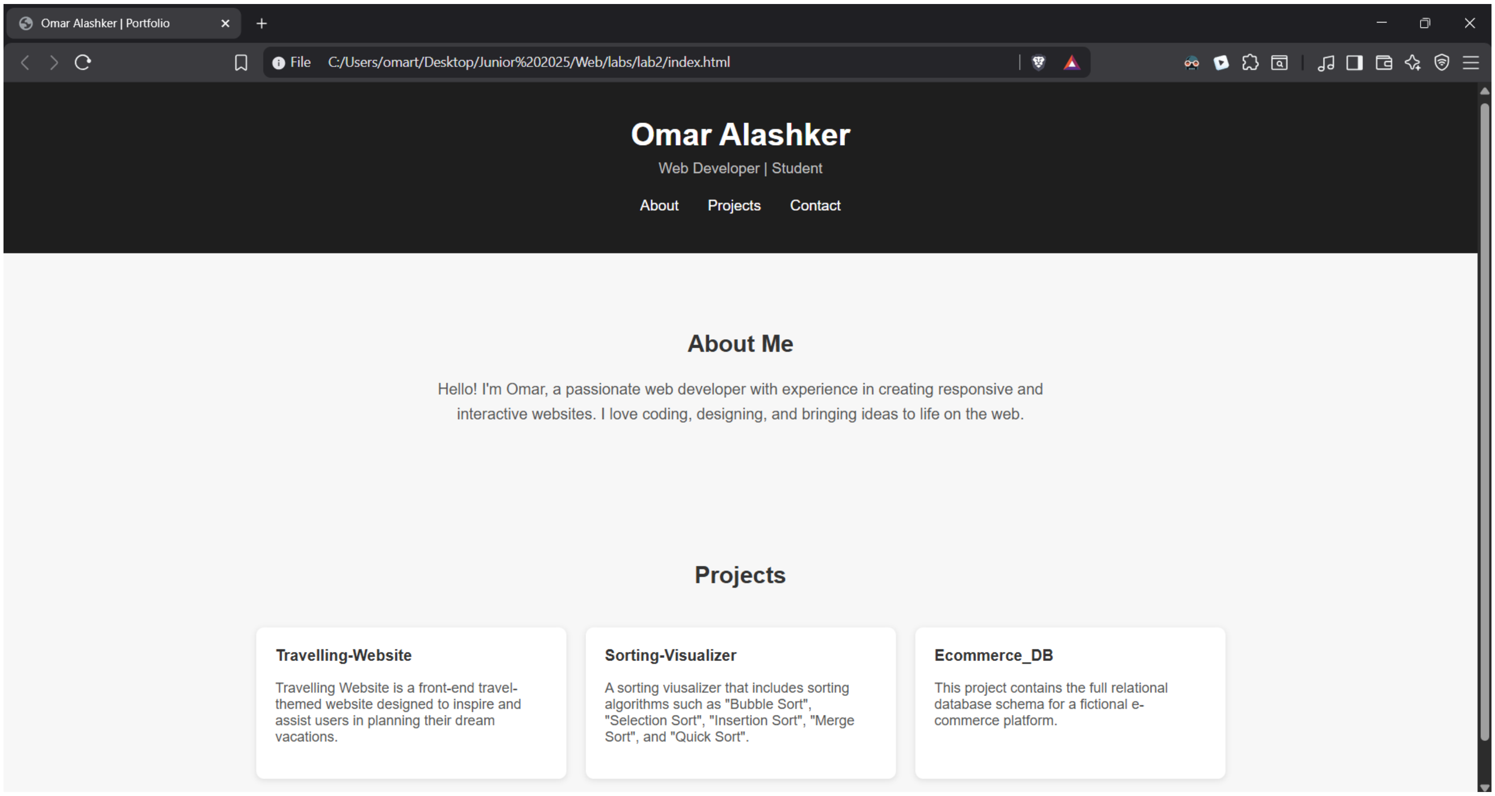
style.css:

```
body {
  margin: 0;
  font-family: Arial, Helvetica, sans-serif;
  background-color: #f7f7f7;
  color: #333;
}
header {
  background-color: #white;
  color: white;
  text-align: center;
  padding: 40px 20px;
}

header h1 {
  margin: 0;
  font-size: 2rem;
}

header p {
  margin: 8px 0 20px;
  font-size: 0.95rem;
  color: #bfbfbf;
}
```



```css
nav ul {
  list-style: none;
  padding: 0;
  margin: 0;
  display: flex;
  justify-content: center;
  gap: 30px;
}
nav a {
  color: white;
  text-decoration: none;
  font-size: 0.95rem;
}
.section {
  text-align: center;
  padding: 60px 20px;
}
#about p {
  max-width: 650px;
  margin: 20px auto 0;
  line-height: 1.6;
  color: #555;
}
.projects {
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  gap: 20px;
  margin-top: 40px;
}
.card {
  background: white;
  padding: 20px;
  border-radius: 8px;
  width: 280px;
  box-shadow: 0 2px 6px rgba(0, 0, 0, 0.1);
  text-align: left;
}
.card h3 {
  margin-top: 0;
  font-size: 1rem;
}

.card p {
  font-size: 0.9rem;
  color: #555;
}
}
```

GUI:


Omar Alashker | Portfolio
File C:/Users/omart/Desktop/Junior%202025/Web/labs/lab2/index.html
About
Projects
Contact

Omar Alashker

Web Developer | Student

About Me

Hello! I'm Omar, a passionate web developer with experience in creating responsive and interactive websites. I love coding, designing, and bringing ideas to life on the web.

Projects

Travelling-Website

Travelling Website is a front-end travel-themed website designed to inspire and assist users in planning their dream vacations.

Sorting-Visualizer

A sorting visualizer that includes sorting algorithms such as "Bubble Sort", "Selection Sort", "Insertion Sort", "Merge Sort", and "Quick Sort".

Ecommerce_DB

This project contains the full relational database schema for a fictional e-commerce platform.

Lab 3:

Index.html:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Simple Calculator</title>
    <link rel="stylesheet" href="style.css" />
  </head>
  <body>
    <div class="calculator">
      <h2>Calculator</h2>
      <input type="text" id="display" readonly />
      <div class="buttons">
        <button class="btn" onclick="clearDisplay()">C</button>
        <button class="btn" onclick="deleteLast()">←</button>
        <button class="btn" onclick="append('%')">%</button>
        <button class="btn" onclick="append('/')">÷</button>

        <button class="btn" onclick="append('7')">7</button>
        <button class="btn" onclick="append('8')">8</button>
        <button class="btn" onclick="append('9')">9</button>
        <button class="btn" onclick="append('*')">×</button>

        <button class="btn" onclick="append('4')">4</button>
        <button class="btn" onclick="append('5')">5</button>
        <button class="btn" onclick="append('6')">6</button>
        <button class="btn" onclick="append('-')">−</button>

        <button class="btn" onclick="append('1')">1</button>
        <button class="btn" onclick="append('2')">2</button>
        <button class="btn" onclick="append('3')">3</button>
        <button class="btn" onclick="append('+')">+</button>

        <button class="btn zero" onclick="append('0')">0</button>
        <button class="btn" onclick="append('.')">.</button>
        <button class="btn equals" onclick="calculate()">=</button>
      </div>
    </div>

    <script src="script.js"></script>
  </body>
</html>
```

style.css:

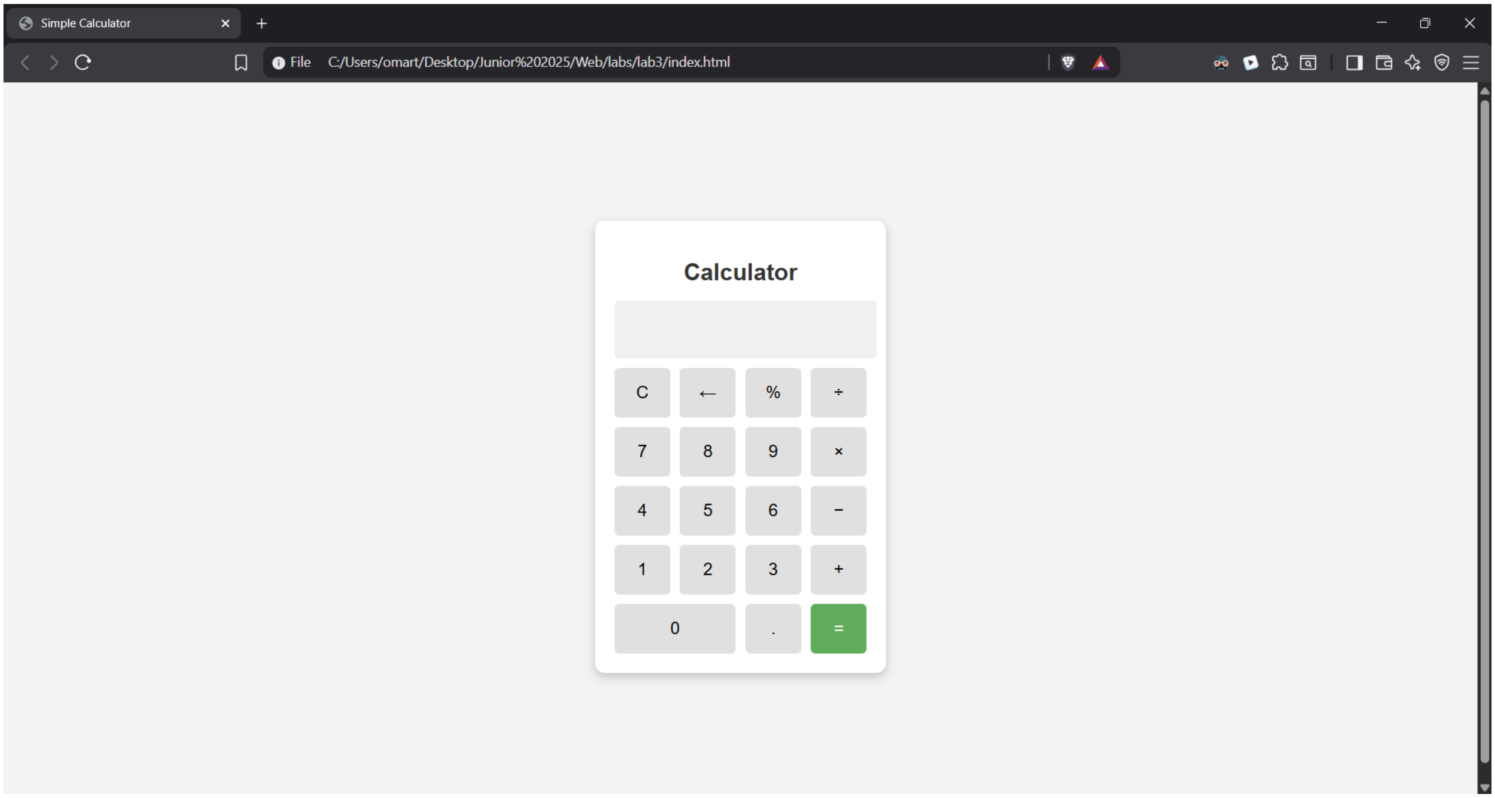
```
body {
  background-color: #f3f3f3;
  font-family: Arial, sans-serif;
  display: flex;
  justify-content: center;
  align-items: center;
  height: 100vh;
}

.calculator {
  background-color: #ffffff;
  padding: 20px;
  border-radius: 10px;
  box-shadow: 0 4px 10px rgba(0, 0, 0, 0.2);
  width: 260px;
}

h2 {
  text-align: center;
  margin-bottom: 15px;
  color: #333;
}

#display {
  width: 100%;
  height: 50px;
  text-align: right;
  font-size: 1.4rem;
  padding: 5px;
  border: none;
  border-radius: 5px;
  background-color: #f0f0f0;
  margin-bottom: 10px;
}

.buttons {
  display: grid;
  grid-template-columns: repeat(4, 1fr);
  gap: 10px;
}

.btn {
  padding: 15px;
  font-size: 1.1rem;
  border: none;
  border-radius: 5px;
  background-color: #e0e0e0;
  cursor: pointer;
  transition: background 0.2s;
}

.btn:hover {
  background-color: #d0d0d0;
}

.equals {
  background-color: #4caf50;
  color: white;
```

```css
}

.equals:hover {
  background-color: #45a049;
}

.zero {
  grid-column: span 2;
}
body {
  background-color: #f3f3f3;
  font-family: Arial, sans-serif;
  display: flex;
  justify-content: center;
  align-items: center;
  height: 100vh;
}

.calculator {
  background-color: #ffffff;
  padding: 20px;
  border-radius: 10px;
  box-shadow: 0 4px 10px rgba(0, 0, 0, 0.2);
  width: 260px;
}

h2 {
  text-align: center;
  margin-bottom: 15px;
  color: #333;
}

#display {
  width: 100%;
  height: 50px;
  text-align: right;
  font-size: 1.4rem;
  padding: 5px;
  border: none;
  border-radius: 5px;
  background-color: #f0f0f0;
  margin-bottom: 10px;
}

.buttons {
  display: grid;
  grid-template-columns: repeat(4, 1fr);
  gap: 10px;
}

.btn {
  padding: 15px;
  font-size: 1.1rem;
  border: none;
  border-radius: 5px;
  background-color: #e0e0e0;
  cursor: pointer;
  transition: background 0.2s;
}

.btn:hover {
  background-color: #d0d0d0;
}

.equals {
  background-color: #4caf50;
  color: white;
}

.equals:hover {
  background-color: #45a049;
}

.zero {
  grid-column: span 2;
}
```

script.js:

```javascript
const display = document.getElementById("display");

function append(value) {
  display.value += value;
}

function clearDisplay() {
  display.value = "";
}

function deleteLast() {
  display.value = display.value.slice(0, -1);
}

function calculate() {
  try {
    display.value = eval(display.value);
  } catch {
    display.value = "Error";
  }
}
```

GUI:


Simple Calculator
File
C:/Users/omart/Desktop/Junior%202025/Web/labs/lab3/index.html
Calculator
C
←
%
÷
7
8
9
×
4
5
6
−
1
2
3
+
0
.
=

Lab 4:

Index.html:

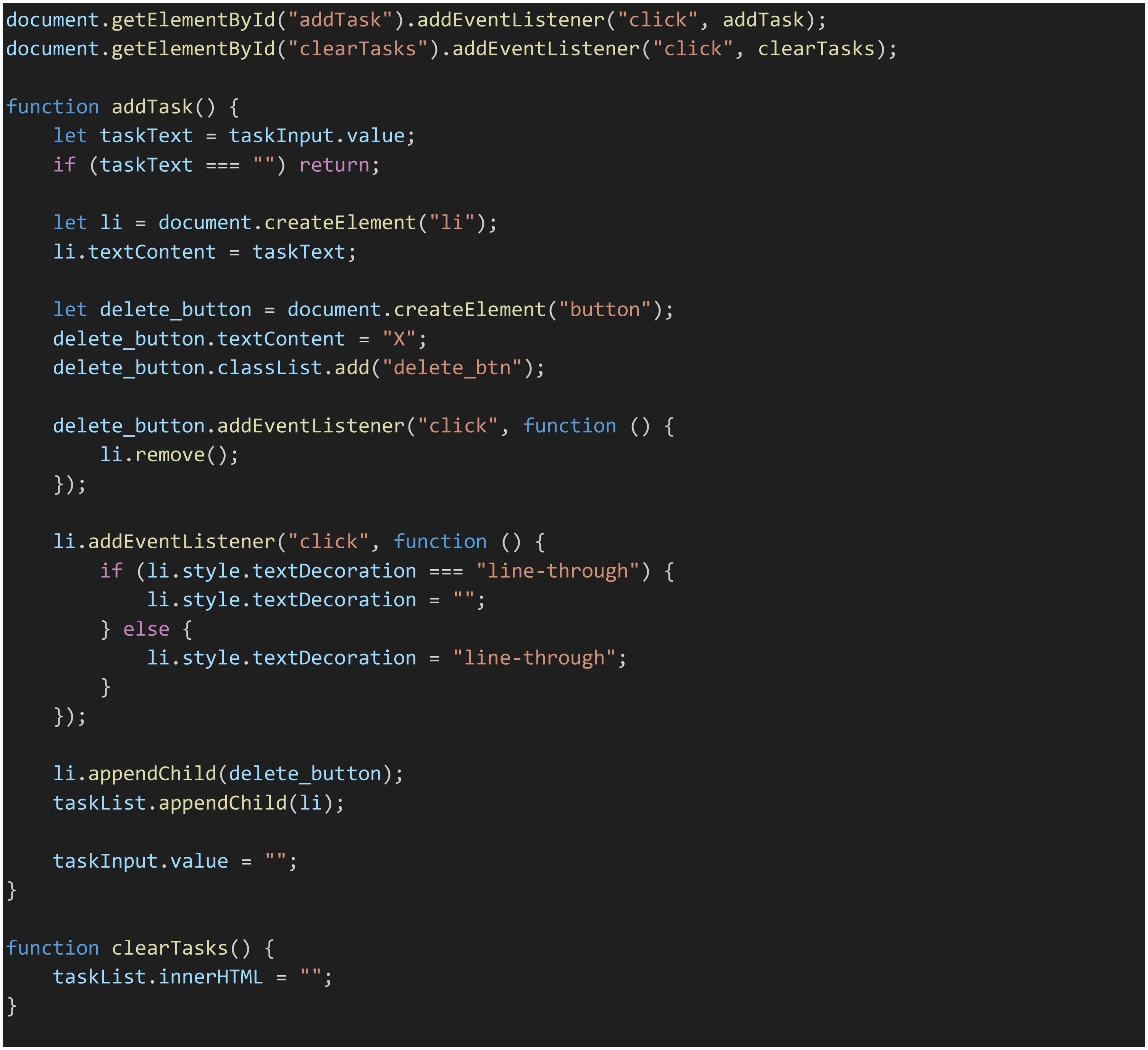
```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="styles.css">
    <title>To-do List</title>
</head>
<body>
    <h1>To-do List</h1>
    <input type="text" id="task" placeholder="Enter a new task">
    <button id="addTask">Add</button>
    <button id="clearTasks">Clear All</button>
    <ul id="taskList"></ul>
    <script src="index.js"></script>
</body>
</html>
```

styles.css:

```css
body {
  background-color: rgb(213, 210, 205);
  font-family: Arial, Helvetica, sans-serif;
}

.delete_btn {
  margin: 10px;
}
```

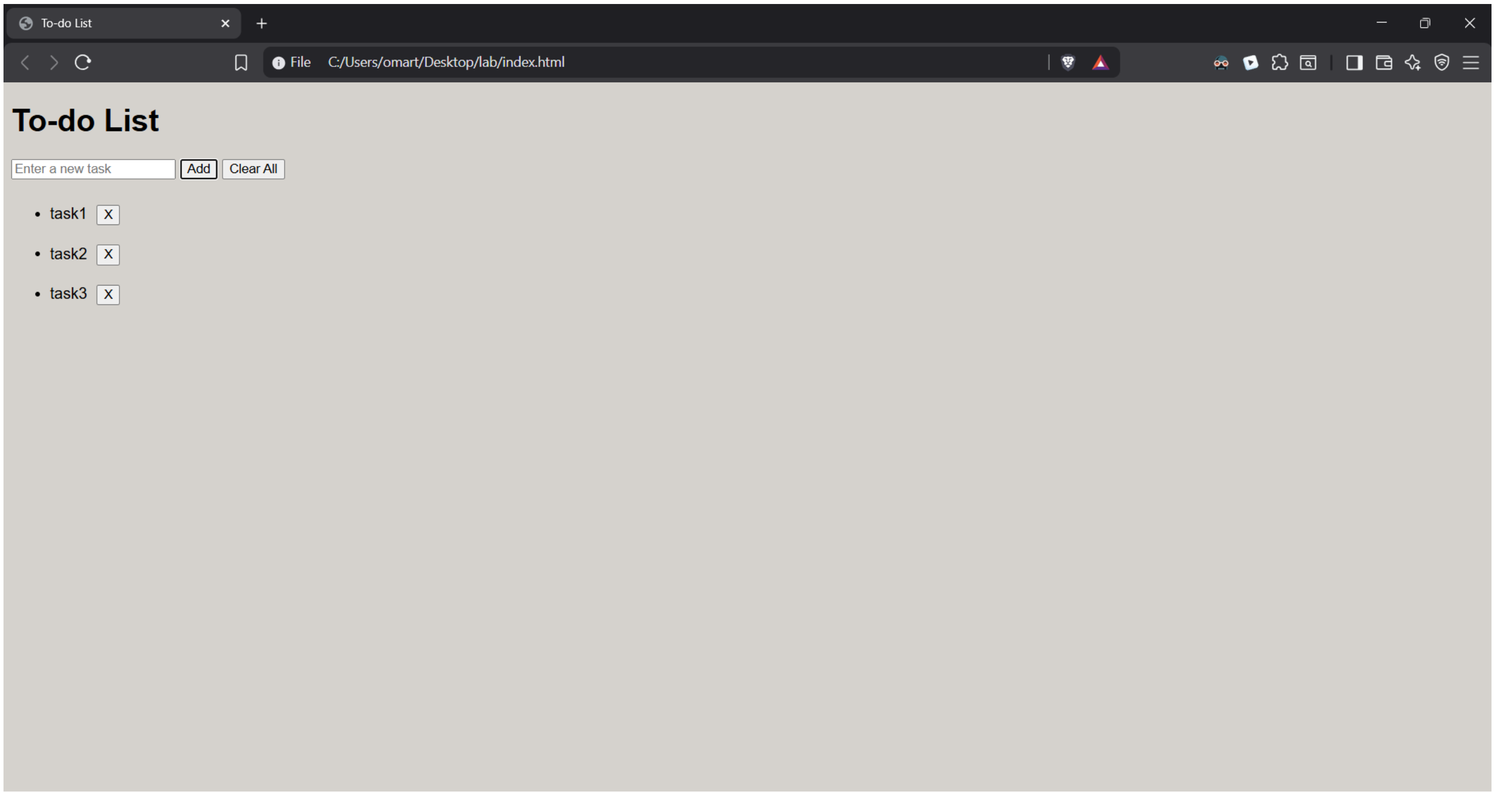
index.js:

```javascript
let taskInput = document.getElementById("task");
let taskList = document.getElementById("taskList");
```



```javascript
document.getElementById("addTask").addEventListener("click", addTask);
document.getElementById("clearTasks").addEventListener("click", clearTasks);

function addTask() {
    let taskText = taskInput.value;
    if (taskText === "") return;

    let li = document.createElement("li");
    li.textContent = taskText;

    let delete_button = document.createElement("button");
    delete_button.textContent = "X";
    delete_button.classList.add("delete_btn");

    delete_button.addEventListener("click", function () {
        li.remove();
    });

    li.addEventListener("click", function () {
        if (li.style.textDecoration === "line-through") {
            li.style.textDecoration = "";
        } else {
            li.style.textDecoration = "line-through";
        }
    });

    li.appendChild(delete_button);
    taskList.appendChild(li);

    taskInput.value = "";
}

function clearTasks() {
    taskList.innerHTML = "";
}
```

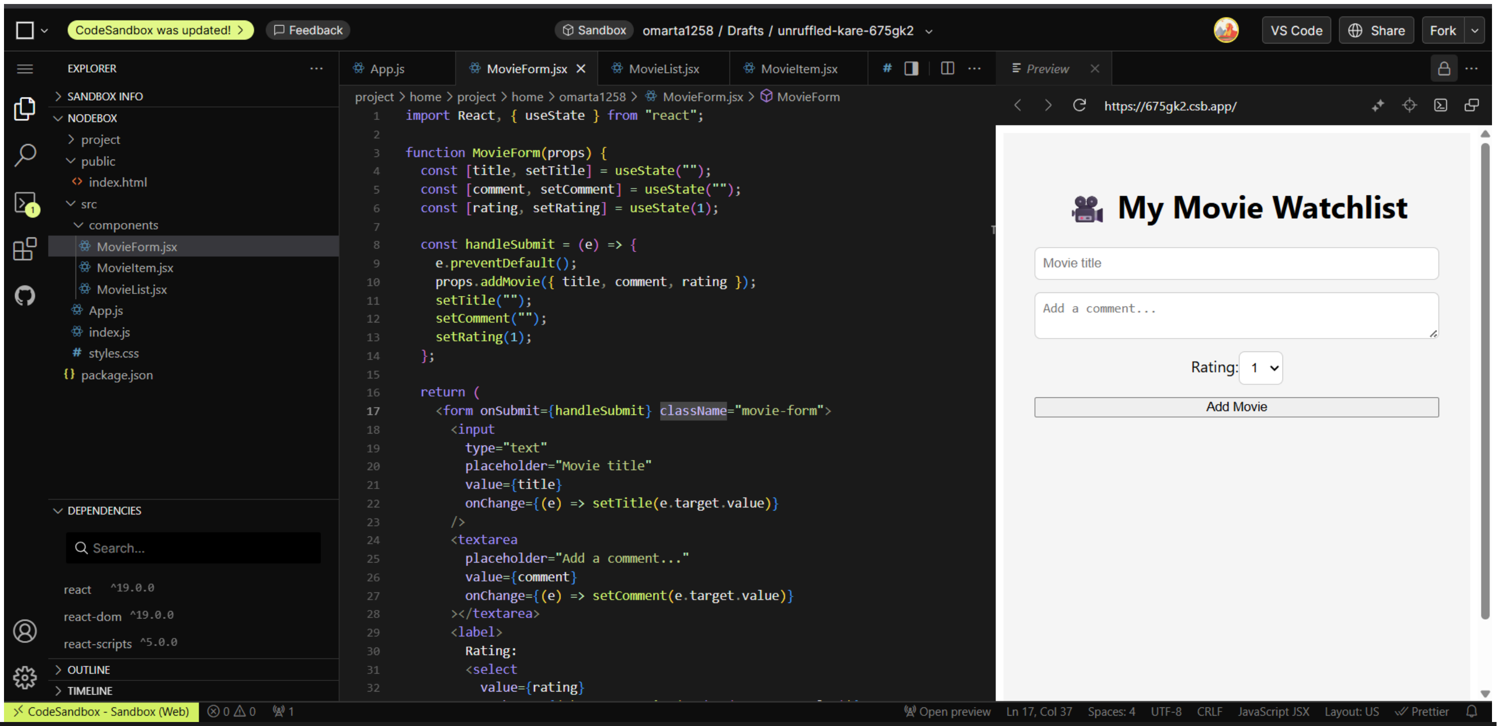
GUI:



To-do List

- task1
- task2
- task3

Lab 5:



```
import React, { useState } from "react";

function MovieForm(props) {
  const [title, setTitle] = useState("");
  const [comment, setComment] = useState("");
  const [rating, setRating] = useState(1);

  const handleSubmit = (e) => {
    e.preventDefault();
    props.addMovie({ title, comment, rating });
    setTitle("");
    setComment("");
    setRating(1);
  };

  return (
    <form onSubmit={handleSubmit} className="movie-form">
      <input
        type="text"
        placeholder="Movie title"
        value={title}
        onChange={(e) => setTitle(e.target.value)}
      />
      <textarea
        placeholder="Add a comment..."
        value={comment}
        onChange={(e) => setComment(e.target.value)}
      ></textarea>
      <label>
        Rating:
        <select
          value={rating}
```

🎥 My Movie Watchlist

Movie title

Add a comment...

Rating: 1

Add Movie

App.js:

```
import React, { useState } from "react";
import MovieForm from "./components/MovieForm.jsx";
import MovieList from "./components/MovieList.jsx";
import "./styles.css";

function App() {
  const [movies, setMovies] = useState([]);

  const addMovie = (movie) => {
    setMovies([...movies, movie]);
  };

  const removeMovie = (index) => {
    setMovies(movies.filter((_, i) => i !== index));
  };

  return (
    <div className="app-container">
      <h1>🎥 My Movie Watchlist</h1>
      <MovieForm addMovie={addMovie} />
      <MovieList movies={movies} removeMovie={removeMovie} />
    </div>
  );
}

export default App;
```

Movieform.jsx:

```
import React, { useState } from "react";

function MovieForm(props) {
  const [title, setTitle] = useState("");
  const [comment, setComment] = useState("");
  const [rating, setRating] = useState(1);

  const handleSubmit = (e) => {
    e.preventDefault();
    props.addMovie({ title, comment, rating });
    setTitle("");
    setComment("");
    setRating(1);
  };

  return (
    <form onSubmit={handleSubmit} className="movie-form">
      <input
        type="text"
        placeholder="Movie title"
        value={title}
        onChange={(e) => setTitle(e.target.value)}
      />
      <textarea
        placeholder="Add a comment..."
        value={comment}
        onChange={(e) => setComment(e.target.value)}
      ></textarea>
      <label>
        Rating:
        <select
          value={rating}
          onChange={(e) => setRating(Number(e.target.value))}
        >
          {[1, 2, 3, 4, 5].map((star) => (
            <option key={star} value={star}>
              {star}
            </option>
          ))}
        </select>
      </label>
      <button type="submit">Add Movie</button>
    </form>
  );
}

export default MovieForm;
```

MovieList.jsx:

```
import React from "react";
import MovieItem from "./MovieItem.jsx";

function MovieList(props) {
  return (
    <div className="movie-list">
      {props.movies.map((movie, index) => (
        <MovieItem
          key={index}
          index={index}
          movie={movie}
          removeMovie={props.removeMovie}
        />
      ))}
    </div>
  );
}

export default MovieList;
```

MovieItem.jsx:

```
import React from "react";

function MovieItem(props) {
  const stars = "☆".repeat(props.movie.rating);

  return (
    <div className="movie-item">
      <h3>{props.movie.title}</h3>
      <p>{stars}</p>
      <p>{props.movie.comment}</p>
      <button onClick={() => props.removeMovie(props.index)}>Remove</button>
    </div>
  );
}

export default MovieItem;
```