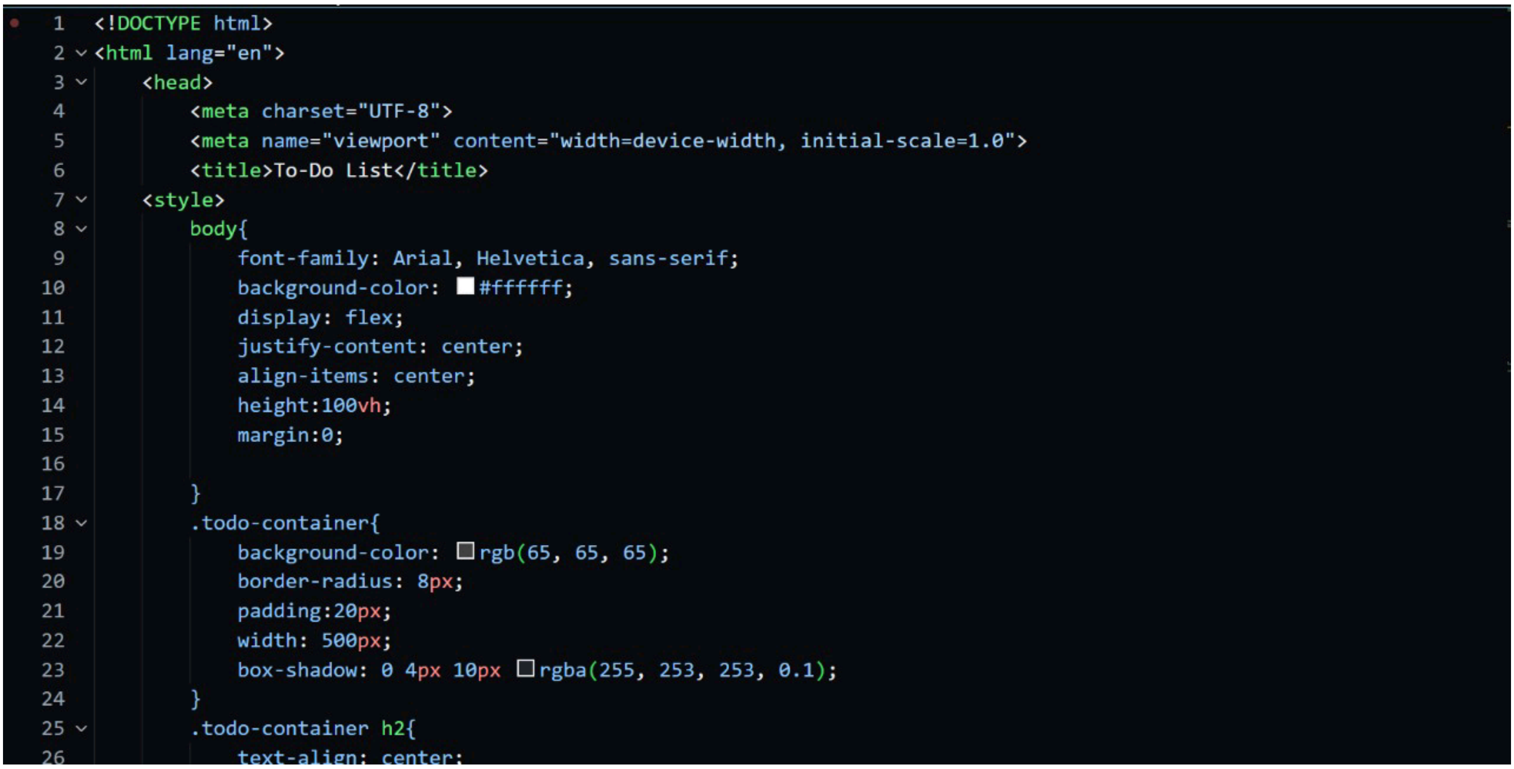
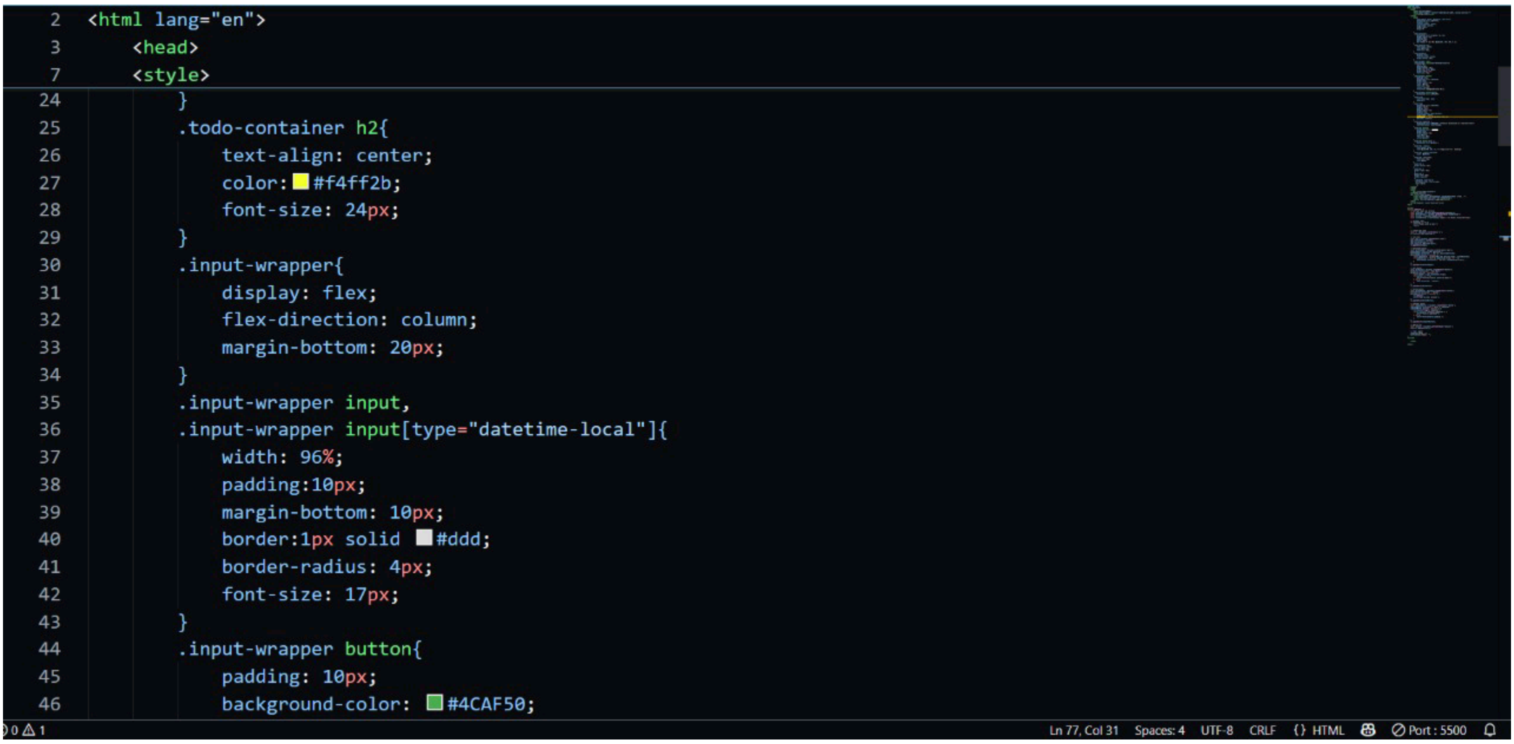
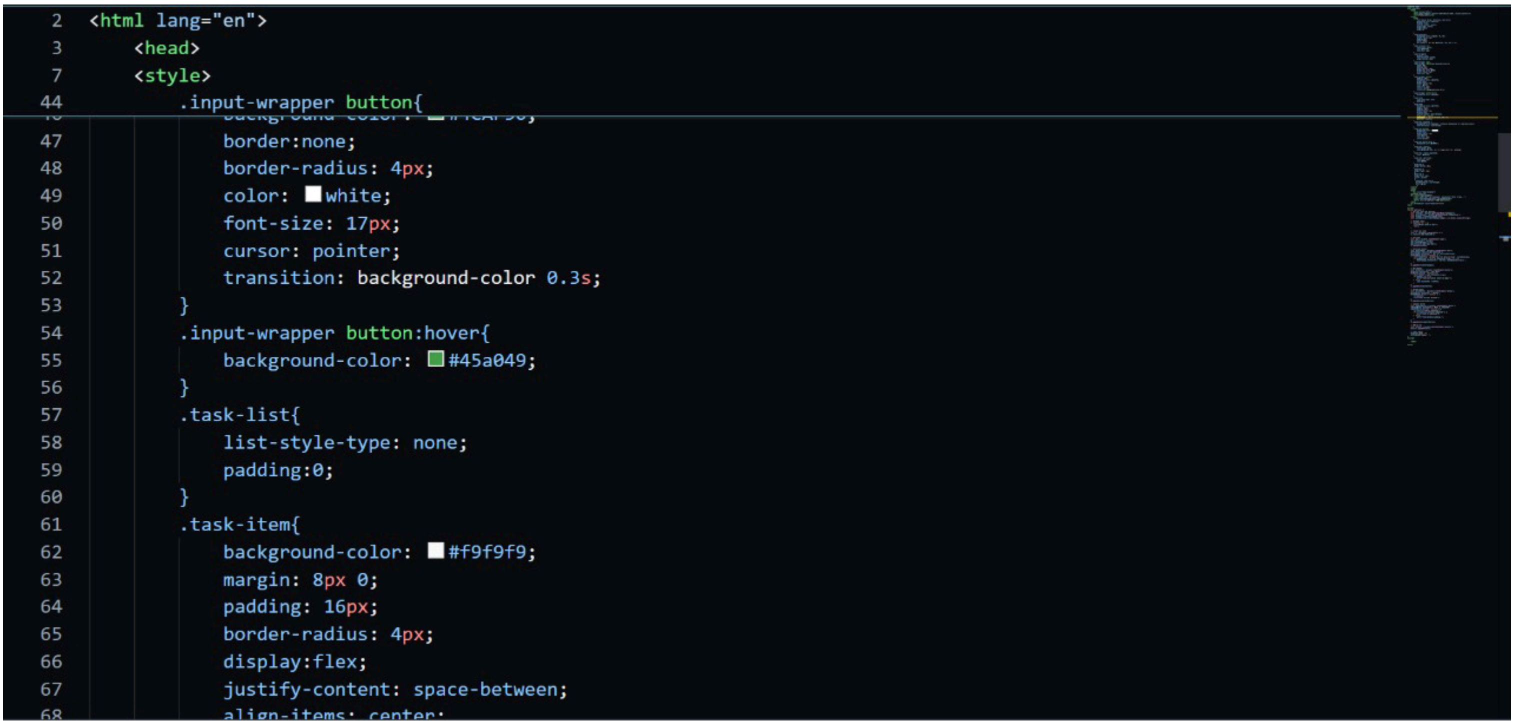


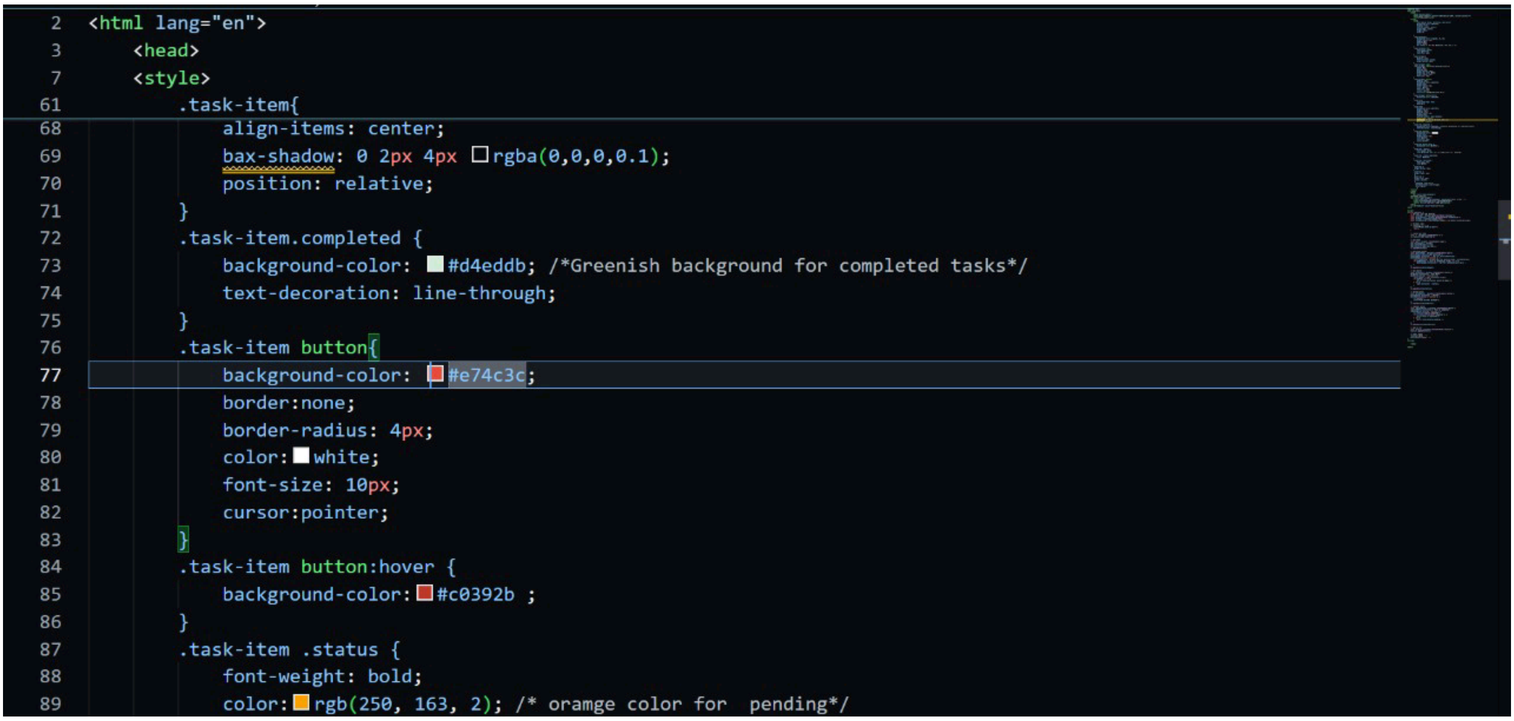
```
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <title>To-Do List</title>
    <style>
        body{
            font-family: Arial, Helvetica, sans-serif;
            background-color: #ffffff;
            display: flex;
            justify-content: center;
            align-items: center;
            height:100vh;
            margin:0;

        }
        .todo-container{
            background-color: rgb(65, 65, 65);
            border-radius: 8px;
            padding:20px;
            width: 500px;
            box-shadow: 0 4px 10px rgba(255, 253, 253, 0.1);
        }
        .todo-container h2{
            text-align: center;
```



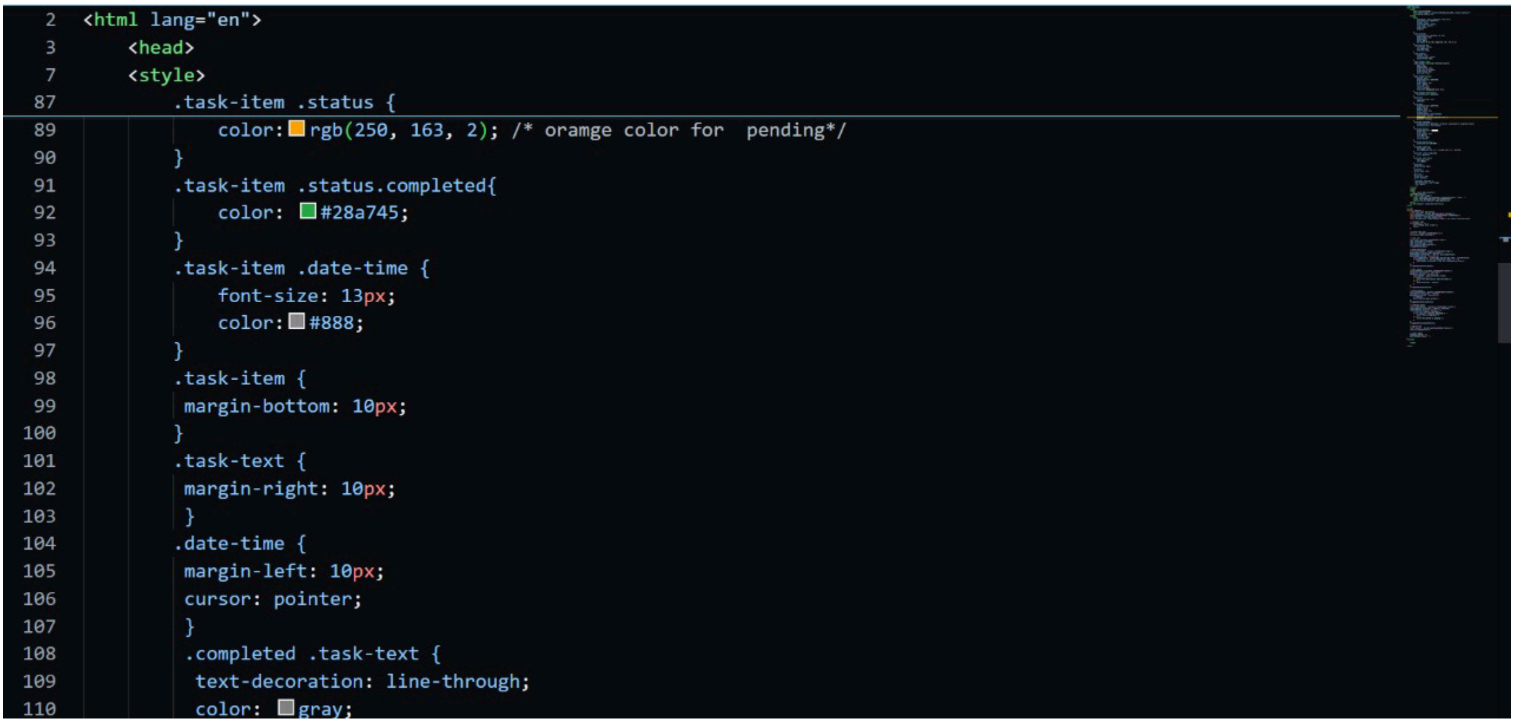
```
<html lang="en">
    <head>
    <style>
        }
        .todo-container h2{
            text-align: center;
            color: #f4ff2b;
            font-size: 24px;
        }
        .input-wrapper{
            display: flex;
            flex-direction: column;
            margin-bottom: 20px;
        }
        .input-wrapper input,
        .input-wrapper input[type="datetime-local"]{
            width: 96%;
            padding:10px;
            margin-bottom: 10px;
            border:1px solid #ddd;
            border-radius: 4px;
            font-size: 17px;
        }
        .input-wrapper button{
            padding: 10px;
            background-color: #4CAF50;
```



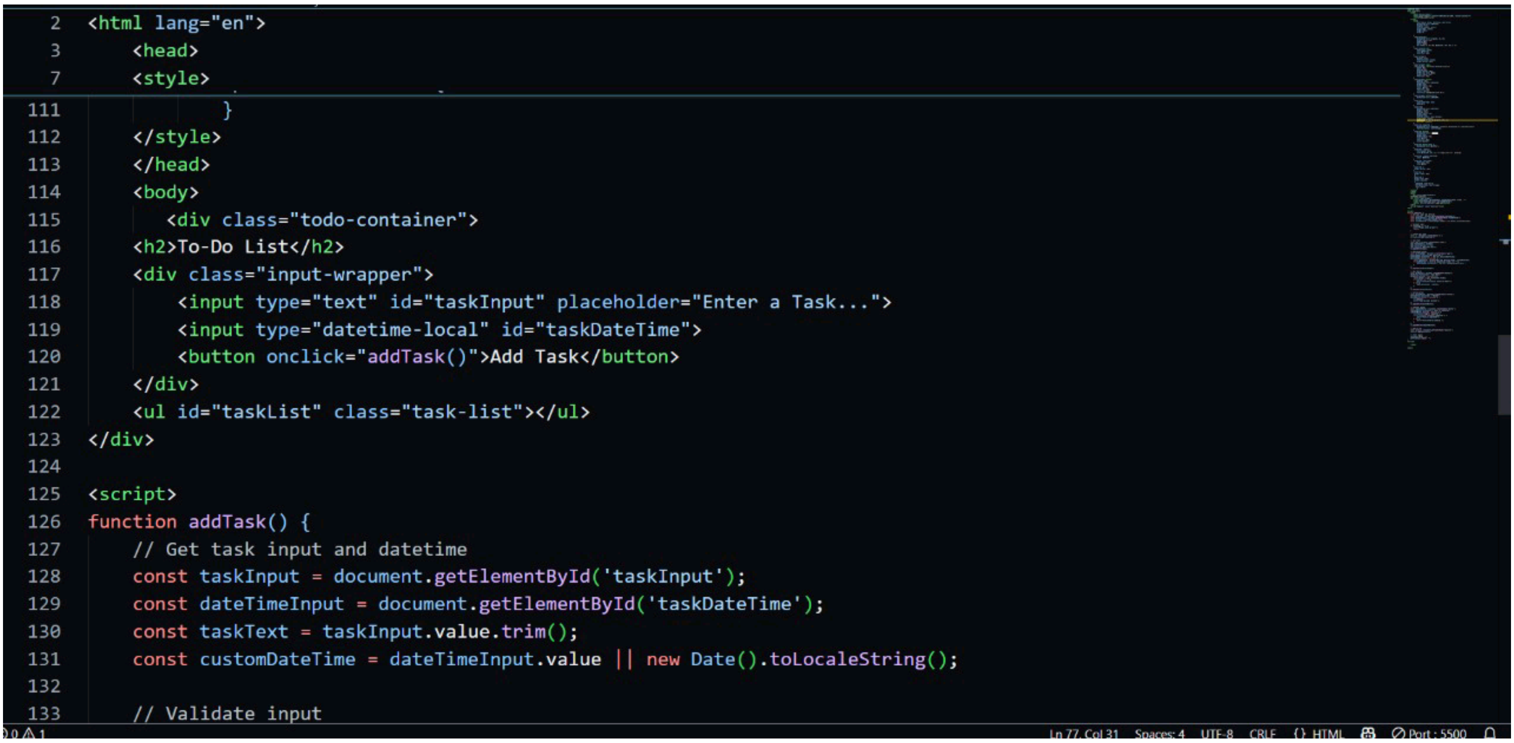
```
<html lang="en">
    <head>
        <style>
            .input-wrapper button{
                background-color: #4CAF50;
                border:none;
                border-radius: 4px;
                color: white;
                font-size: 17px;
                cursor: pointer;
                transition: background-color 0.3s;
            }
            .input-wrapper button:hover{
                background-color: #45a049;
            }
            .task-list{
                list-style-type: none;
                padding:0;
            }
            .task-item{
                background-color: #f9f9f9;
                margin: 8px 0;
                padding: 16px;
                border-radius: 4px;
                display:flex;
                justify-content: space-between;
                align-items: center;
```



```
<html lang="en">
    <head>
    <style>
        .task-item{
            align-items: center;
            bax-shadow: 0 2px 4px rgba(0,0,0,0.1);
            position: relative;
        }
        .task-item.completed {
            background-color: #d4eddb; /*Greenish background for completed tasks*/
            text-decoration: line-through;
        }
        .task-item button{
            background-color: #e74c3c;
            border:none;
            border-radius: 4px;
            color: white;
            font-size: 10px;
            cursor:pointer;
        }
        .task-item button:hover {
            background-color: #c0392b ;
        }
        .task-item .status {
            font-weight: bold;
            color: rgb(250, 163, 2); /* oramge color for  pending*/
```

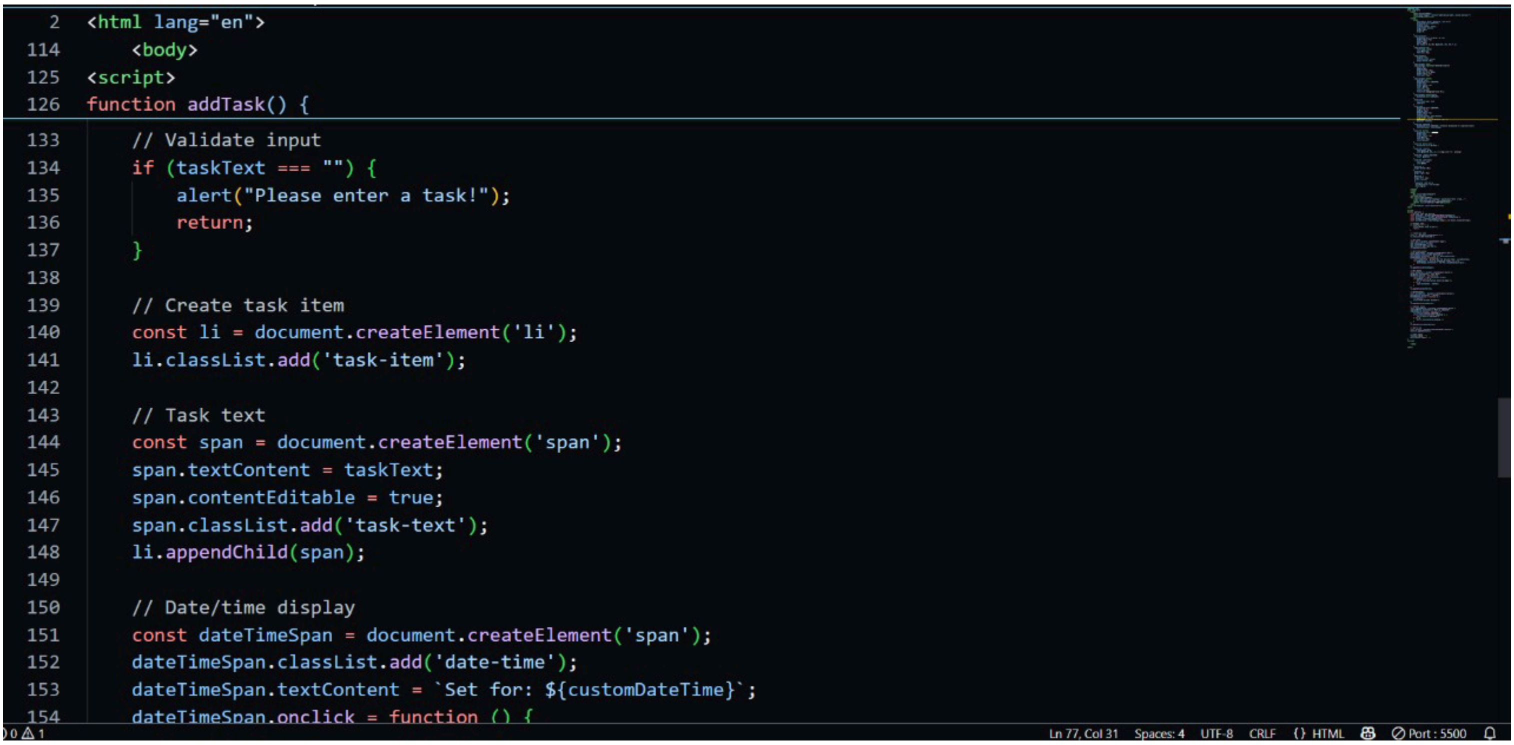


```
<html lang="en">
    <head>
    <style>
        .task-item .status {
            color: rgb(250, 163, 2); /* oramge color for  pending*/
        }
        .task-item .status.completed{
            color:  #28a745;
        }
        .task-item .date-time {
            font-size: 13px;
            color: #888;
        }
        .task-item {
         margin-bottom: 10px;
        }
        .task-text {
         margin-right: 10px;
         }
        .date-time {
         margin-left: 10px;
         cursor: pointer;
         }
         .completed .task-text {
          text-decoration: line-through;
          color:  gray;
```



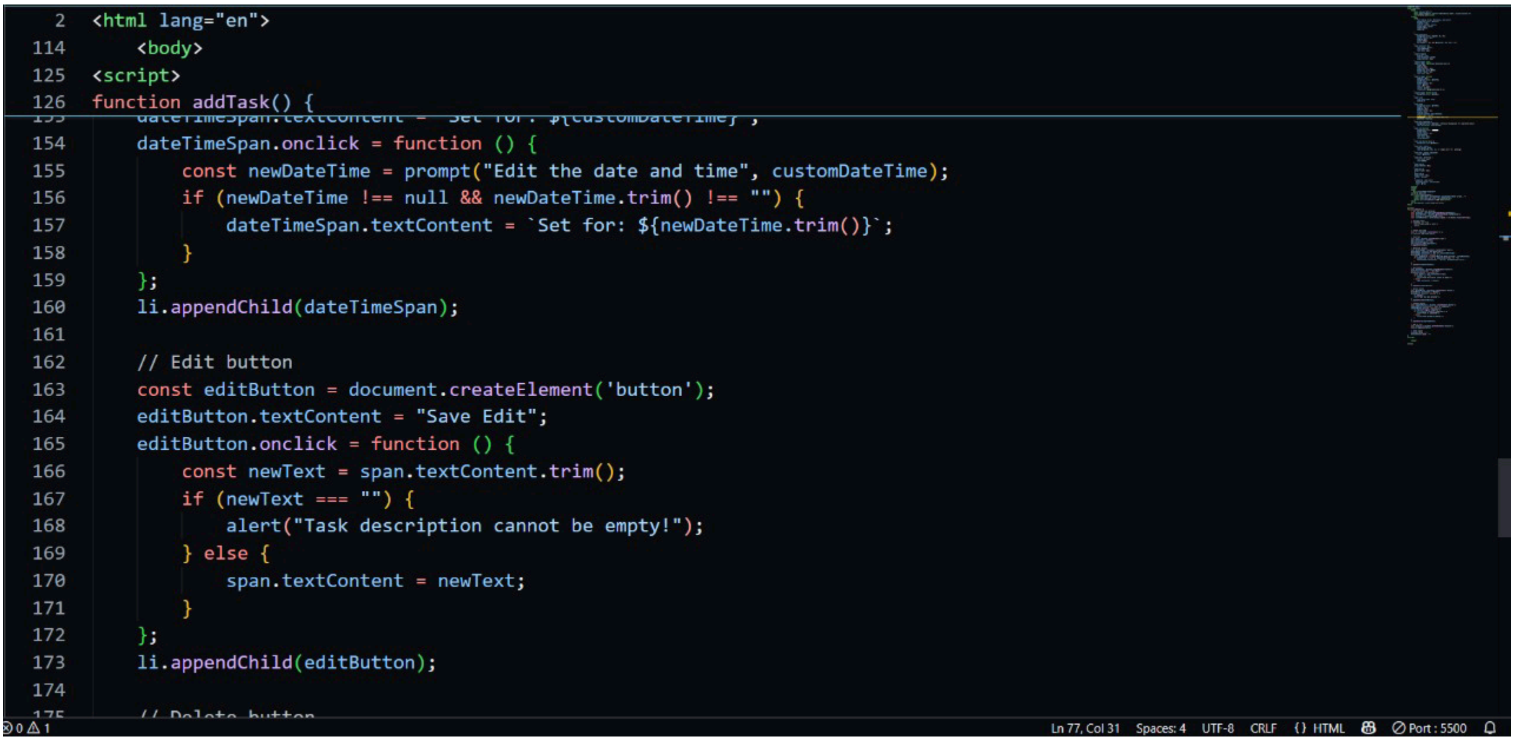
```
<html lang="en">
    <head>
    <style>
            }
    </style>
    </head>
    <body>
       <div class="todo-container">
    <h2>To-Do List</h2>
    <div class="input-wrapper">
        <input type="text" id="taskInput" placeholder="Enter a Task...">
        <input type="datetime-local" id="taskDateTime">
        <button onclick="addTask()">Add Task</button>
    </div>
    <ul id="taskList" class="task-list"></ul>
</div>

<script>
function addTask() {
    // Get task input and datetime
    const taskInput = document.getElementById('taskInput');
    const dateTimeInput = document.getElementById('taskDateTime');
    const taskText = taskInput.value.trim();
    const customDateTime = dateTimeInput.value || new Date().toLocaleString();

    // Validate input
```

⊗ 0 ⚠ 1   Ln 77, Col 31   Spaces: 4   UTF-8   CRLF   {} HTML   ⊘ Port : 5500

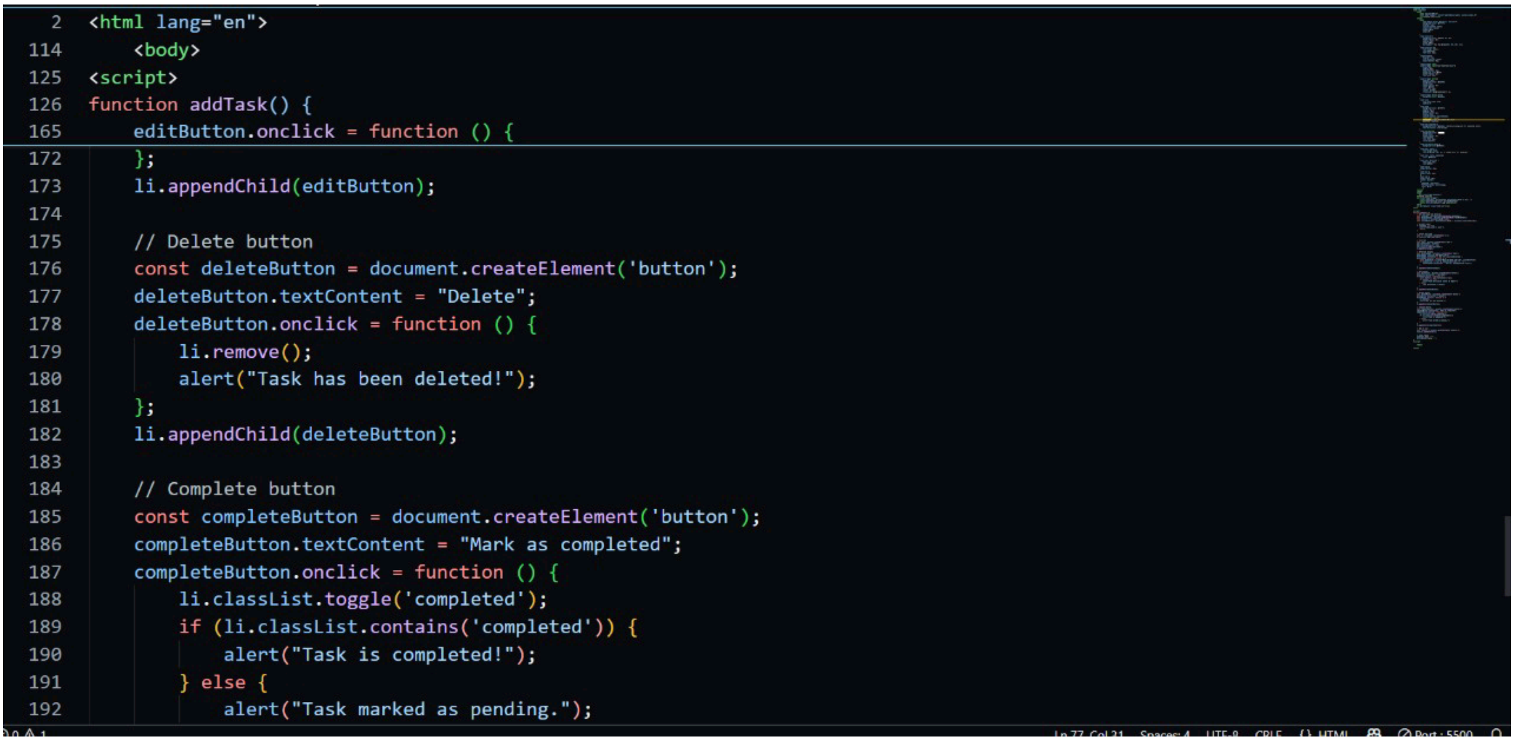


```
<html lang="en">
    <body>
<script>
function addTask() {
    // Validate input
    if (taskText === "") {
        alert("Please enter a task!");
        return;
    }

    // Create task item
    const li = document.createElement('li');
    li.classList.add('task-item');

    // Task text
    const span = document.createElement('span');
    span.textContent = taskText;
    span.contentEditable = true;
    span.classList.add('task-text');
    li.appendChild(span);

    // Date/time display
    const dateTimeSpan = document.createElement('span');
    dateTimeSpan.classList.add('date-time');
    dateTimeSpan.textContent = `Set for: ${customDateTime}`;
    dateTimeSpan.onclick = function () {
```

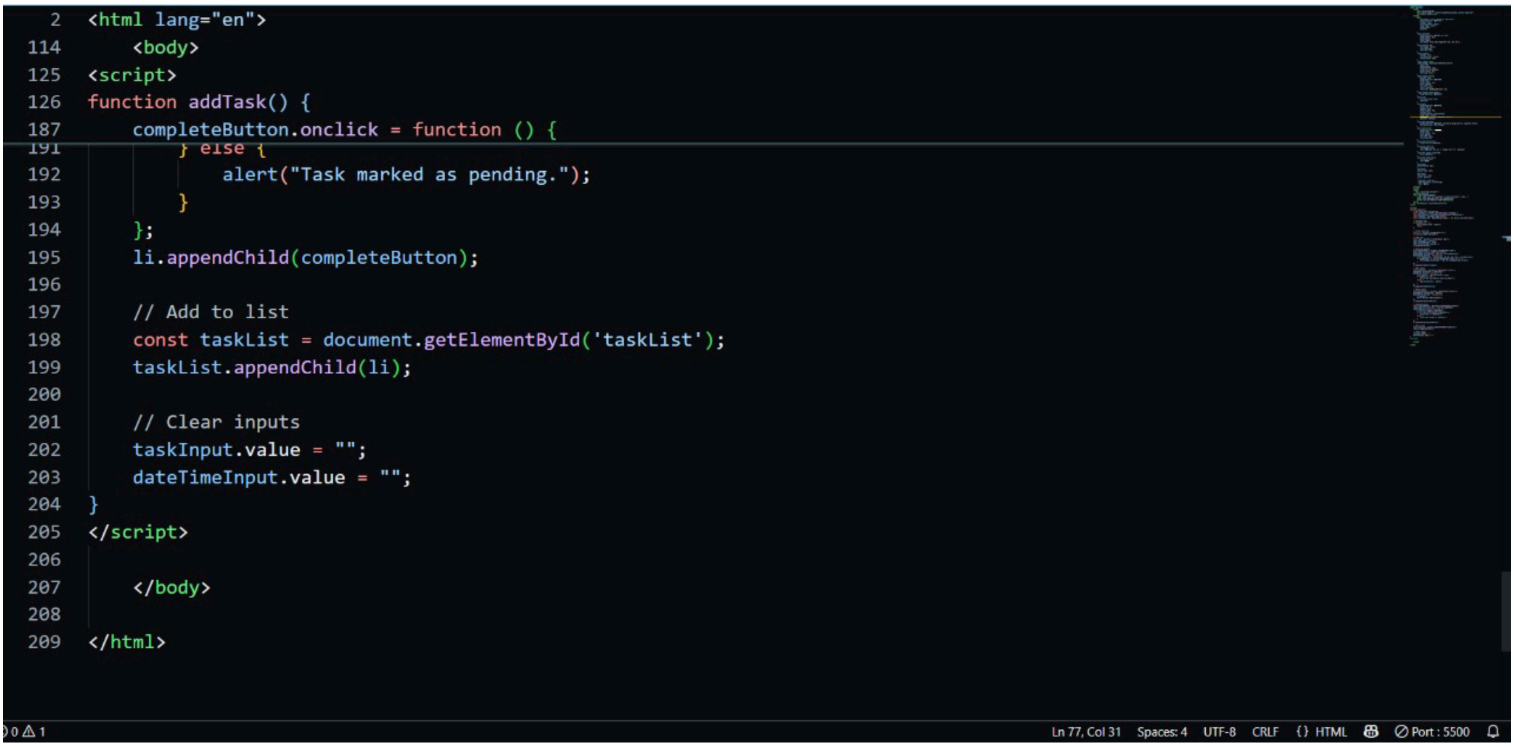
0 ⚠ 1   Ln 77, Col 31   Spaces: 4   UTF-8   CRLF   {} HTML   Port : 5500



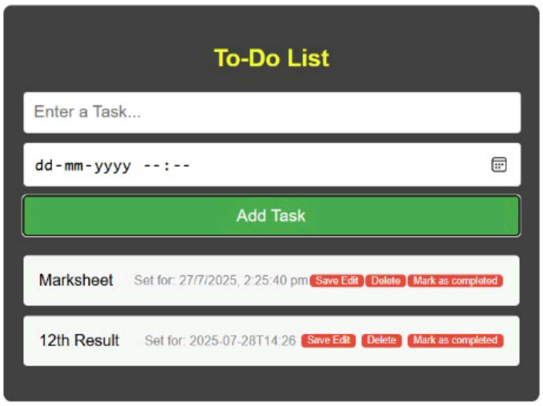
```
<html lang="en">
    <body>
<script>
function addTask() {
    dateTimeSpan.textContent = `Set for: ${customDateTime}`;
    dateTimeSpan.onclick = function () {
        const newDateTime = prompt("Edit the date and time", customDateTime);
        if (newDateTime !== null && newDateTime.trim() !== "") {
            dateTimeSpan.textContent = `Set for: ${newDateTime.trim()}`;
        }
    };
    li.appendChild(dateTimeSpan);

    // Edit button
    const editButton = document.createElement('button');
    editButton.textContent = "Save Edit";
    editButton.onclick = function () {
        const newText = span.textContent.trim();
        if (newText === "") {
            alert("Task description cannot be empty!");
        } else {
            span.textContent = newText;
        }
    };
    li.appendChild(editButton);

    // Delete button
```



```
<html lang="en">
    <body>
<script>
function addTask() {
    editButton.onclick = function () {
    };
    li.appendChild(editButton);

    // Delete button
    const deleteButton = document.createElement('button');
    deleteButton.textContent = "Delete";
    deleteButton.onclick = function () {
        li.remove();
        alert("Task has been deleted!");
    };
    li.appendChild(deleteButton);

    // Complete button
    const completeButton = document.createElement('button');
    completeButton.textContent = "Mark as completed";
    completeButton.onclick = function () {
        li.classList.toggle('completed');
        if (li.classList.contains('completed')) {
            alert("Task is completed!");
        } else {
            alert("Task marked as pending.");
```



```
<html lang="en">
    <body>
<script>
function addTask() {
    completeButton.onclick = function () {
        } else {
            alert("Task marked as pending.");
        }
    };
    li.appendChild(completeButton);

    // Add to list
    const taskList = document.getElementById('taskList');
    taskList.appendChild(li);

    // Clear inputs
    taskInput.value = "";
    dateTimeInput.value = "";
}
</script>

    </body>

</html>
```



Marksheet Set for: 27/7/2025, 2:25:40 pm

12th Result Set for: 2025-07-28T14:26