KIET GROUP OF INSTITUTIONS

FULL STACK WEB DEVELOPMENT INTERNSHIP

NAME: Naman Sharma
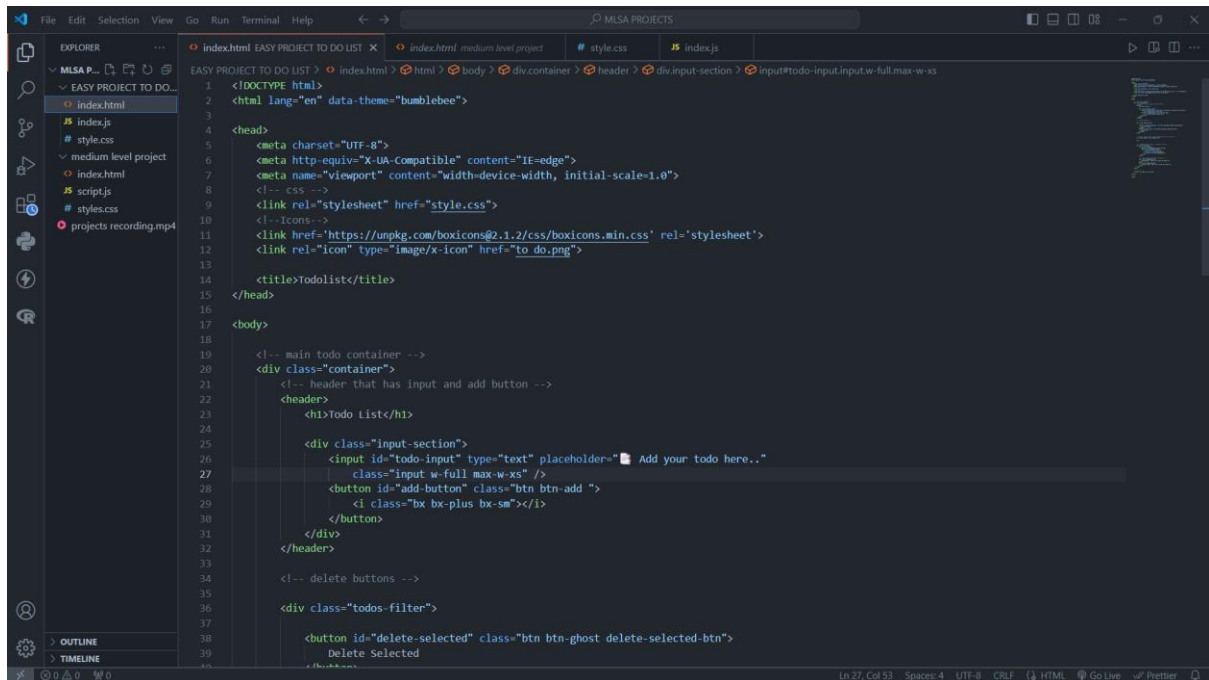
Branch: CSE

Year: 3rd year(5th Semester)
Section: B

PROJECT 1:

CODE

```html
<!DOCTYPE html>
<html lang="en" data-theme="bumblebee">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <!-- css -->
    <link rel="stylesheet" href="style.css">
    <!--Icons-->
    <link href='https://unpkg.com/boxicons@2.1.2/css/boxicons.min.css' rel='stylesheet'>
    <link rel="icon" type="image/x-icon" href="to do.png">

    <title>Todolist</title>
</head>

<body>

    <!-- main todo container -->
    <div class="container">
        <!-- header that has input and add button -->
        <header>
            <h1>Todo List</h1>

            <div class="input-section">
                <input id="todo-input" type="text" placeholder=" Add your todo here.."
                    class="input w-full max-w-xs" />
                <button id="add-button" class="btn btn-add ">
                    <i class="bx bx-plus bx-sm"></i>
                </button>
            </div>
        </header>

        <!-- delete buttons -->

        <div class="todos-filter">

            <button id="delete-selected" class="btn btn-ghost delete-selected-btn">
                Delete Selected
            </button>
```
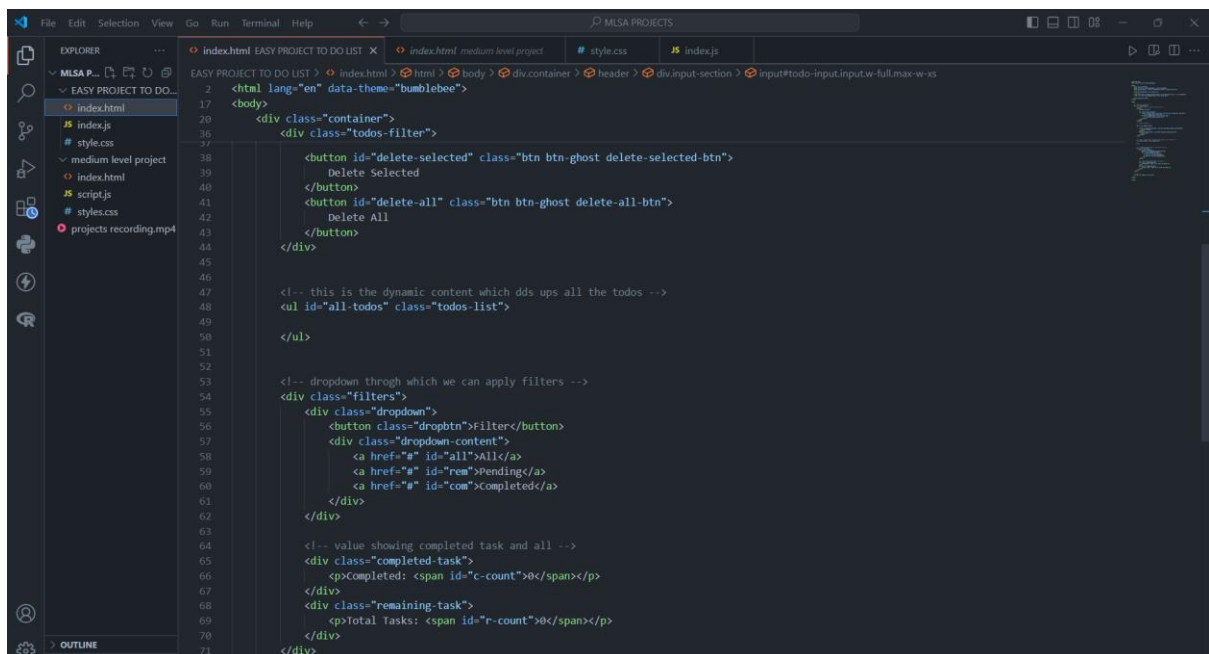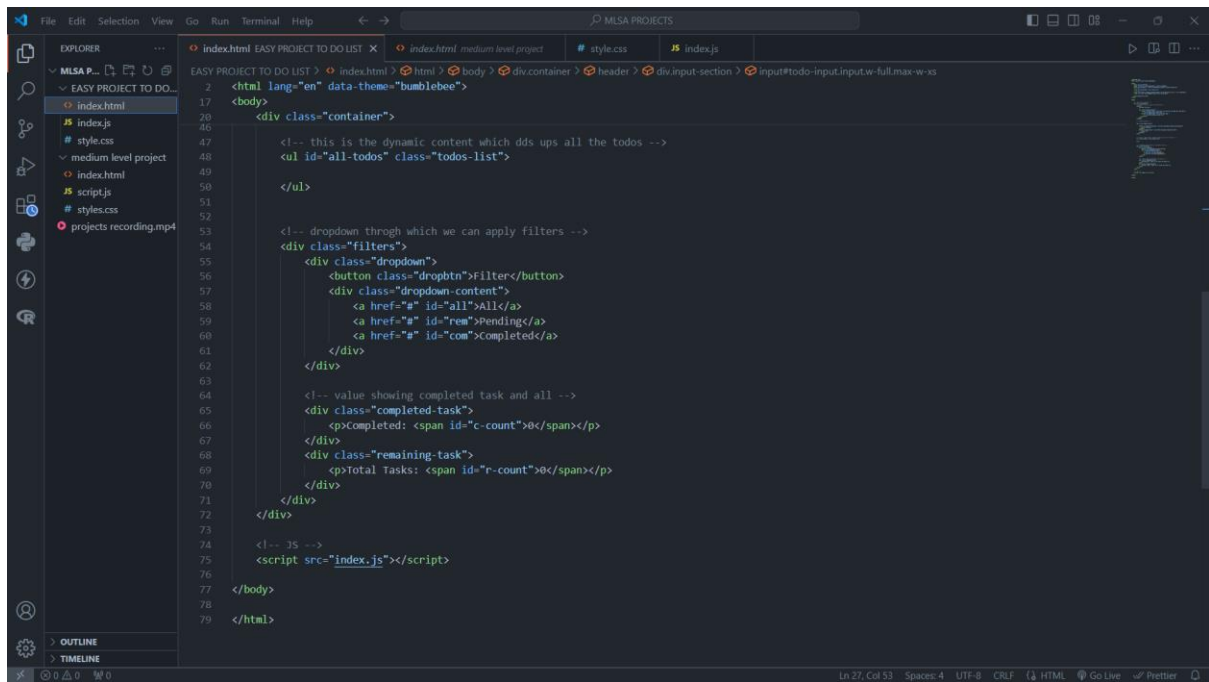
```html
<html lang="en" data-theme="bumblebee">
<body>
    <div class="container">
        <div class="todos-filter">

            <button id="delete-selected" class="btn btn-ghost delete-selected-btn">
                Delete Selected
            </button>
            <button id="delete-all" class="btn btn-ghost delete-all-btn">
                Delete All
            </button>
        </div>


        <!-- this is the dynamic content which dds ups all the todos -->
        <ul id="all-todos" class="todos-list">

        </ul>


        <!-- dropdown throgh which we can apply filters -->
        <div class="filters">
            <div class="dropdown">
                <button class="dropbtn">Filter</button>
                <div class="dropdown-content">
                    <a href="#" id="all">All</a>
                    <a href="#" id="rem">Pending</a>
                    <a href="#" id="com">Completed</a>
                </div>
            </div>

            <!-- value showing completed task and all -->
            <div class="completed-task">
                <p>Completed: <span id="c-count">0</span></p>
            </div>
            <div class="remaining-task">
                <p>Total Tasks: <span id="r-count">0</span></p>
            </div>
        </div>
    </div>
```
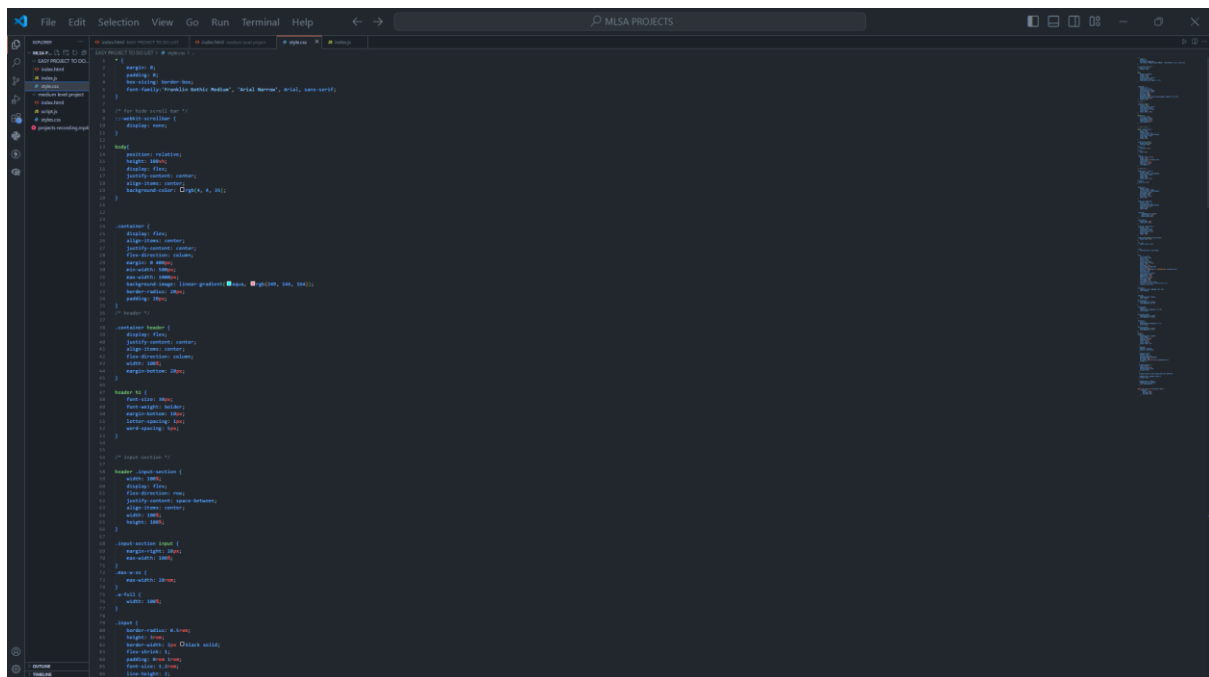
index.html EASY PROJECT TO DO LIST ×   index.html medium level project   style.css   index.js

EASY PROJECT TO DO LIST › ◇ index.html › ⬡ html › ⬡ body › ⬡ div.container › ⬡ header › ⬡ div.input-section › ⬡ input#todo-input.input.w-full.max-w-xs

```html
<html lang="en" data-theme="bumblebee">
<body>
    <div class="container">

        <!-- this is the dynamic content which dds ups all the todos -->
        <ul id="all-todos" class="todos-list">

        </ul>


        <!-- dropdown throgh which we can apply filters -->
        <div class="filters">
            <div class="dropdown">
                <button class="dropbtn">Filter</button>
                <div class="dropdown-content">
                    <a href="#" id="all">All</a>
                    <a href="#" id="rem">Pending</a>
                    <a href="#" id="com">Completed</a>
                </div>
            </div>

            <!-- value showing completed task and all -->
            <div class="completed-task">
                <p>Completed: <span id="c-count">0</span></p>
            </div>
            <div class="remaining-task">
                <p>Total Tasks: <span id="r-count">0</span></p>
            </div>
        </div>
    </div>

    <!-- JS -->
    <script src="index.js"></script>

</body>

</html>
```
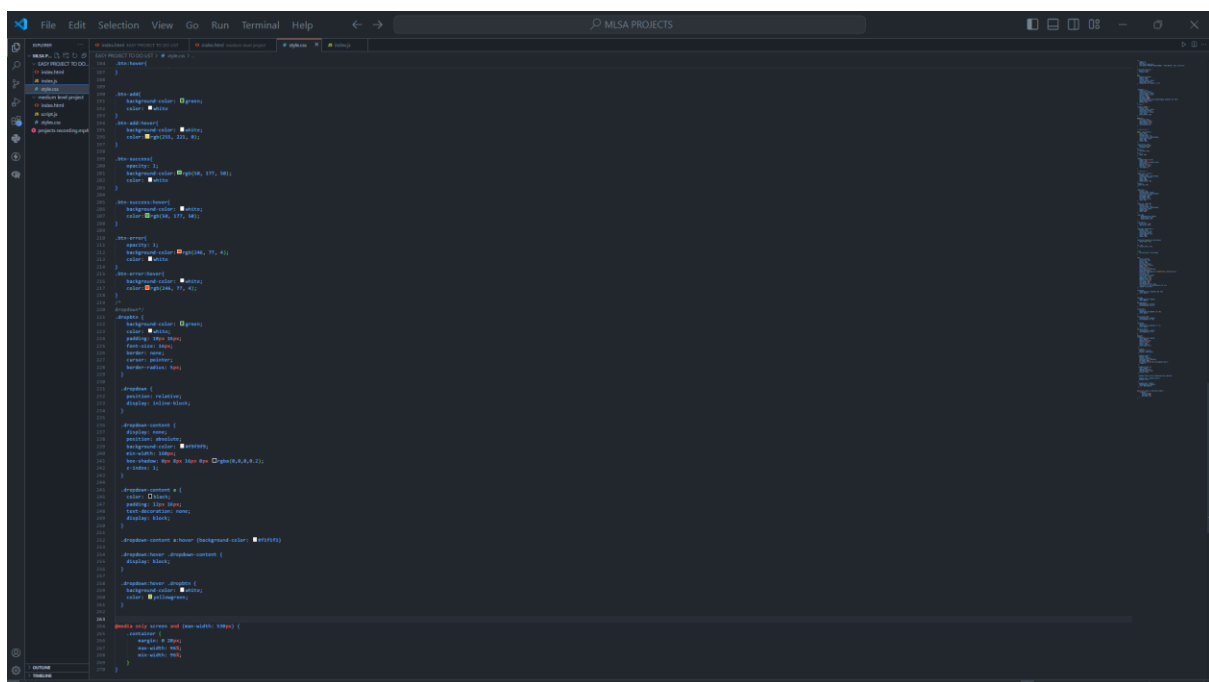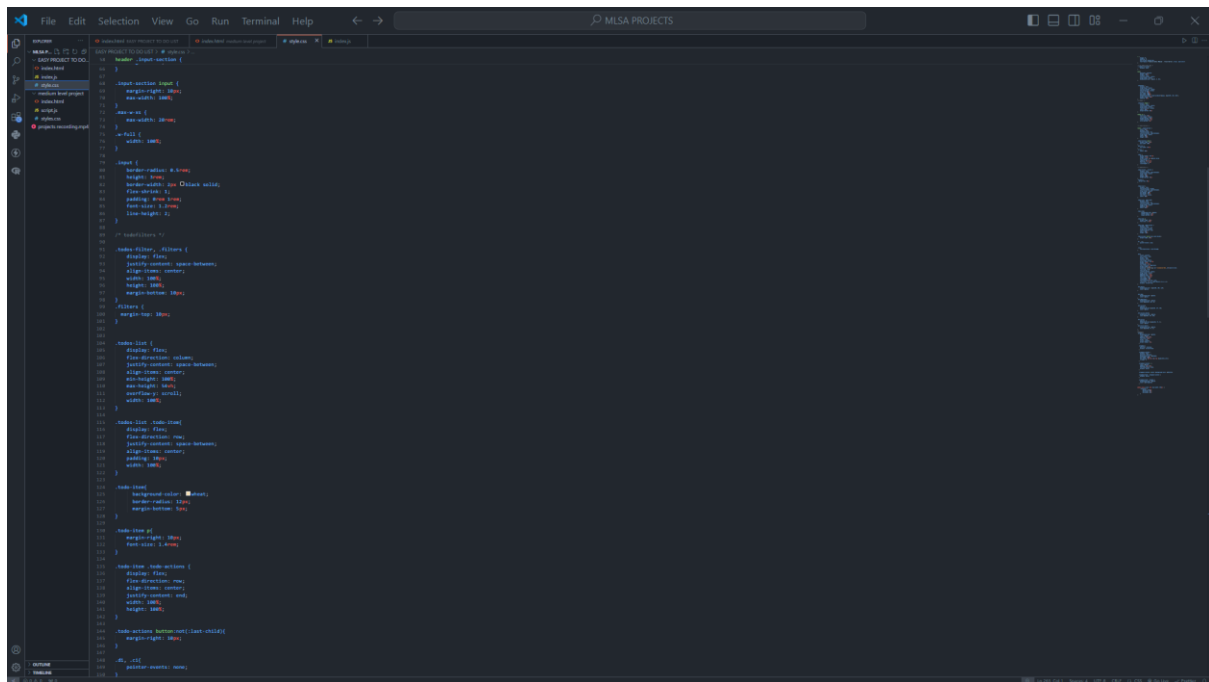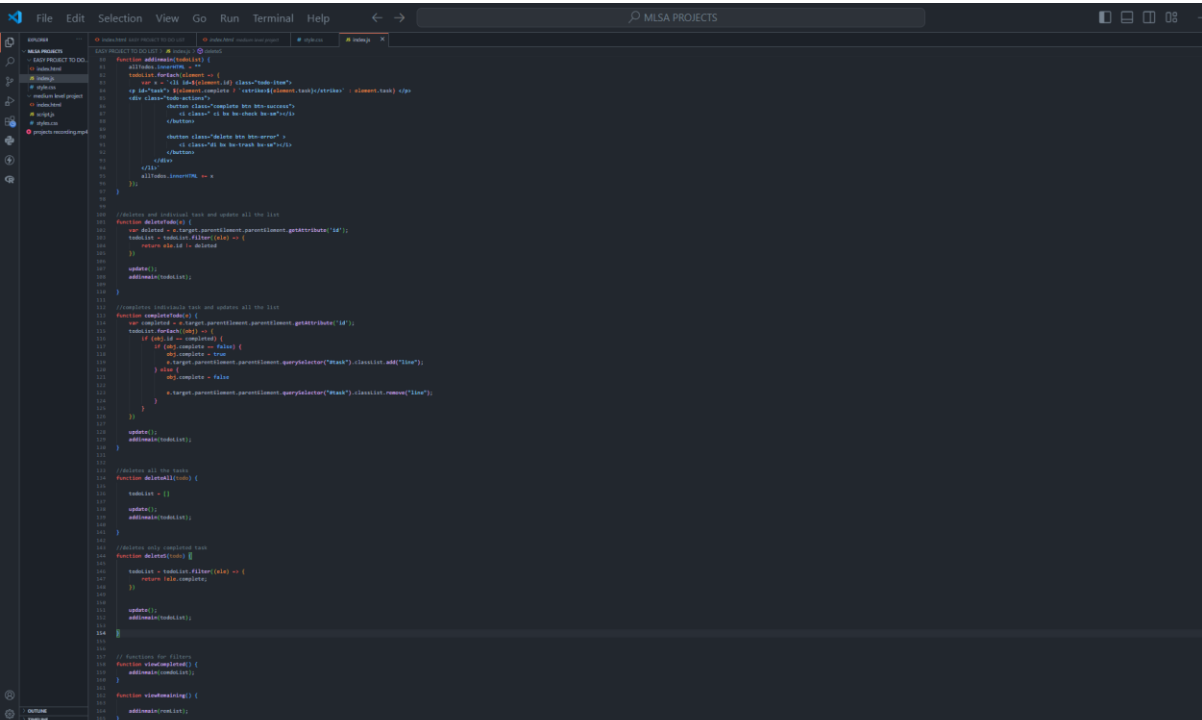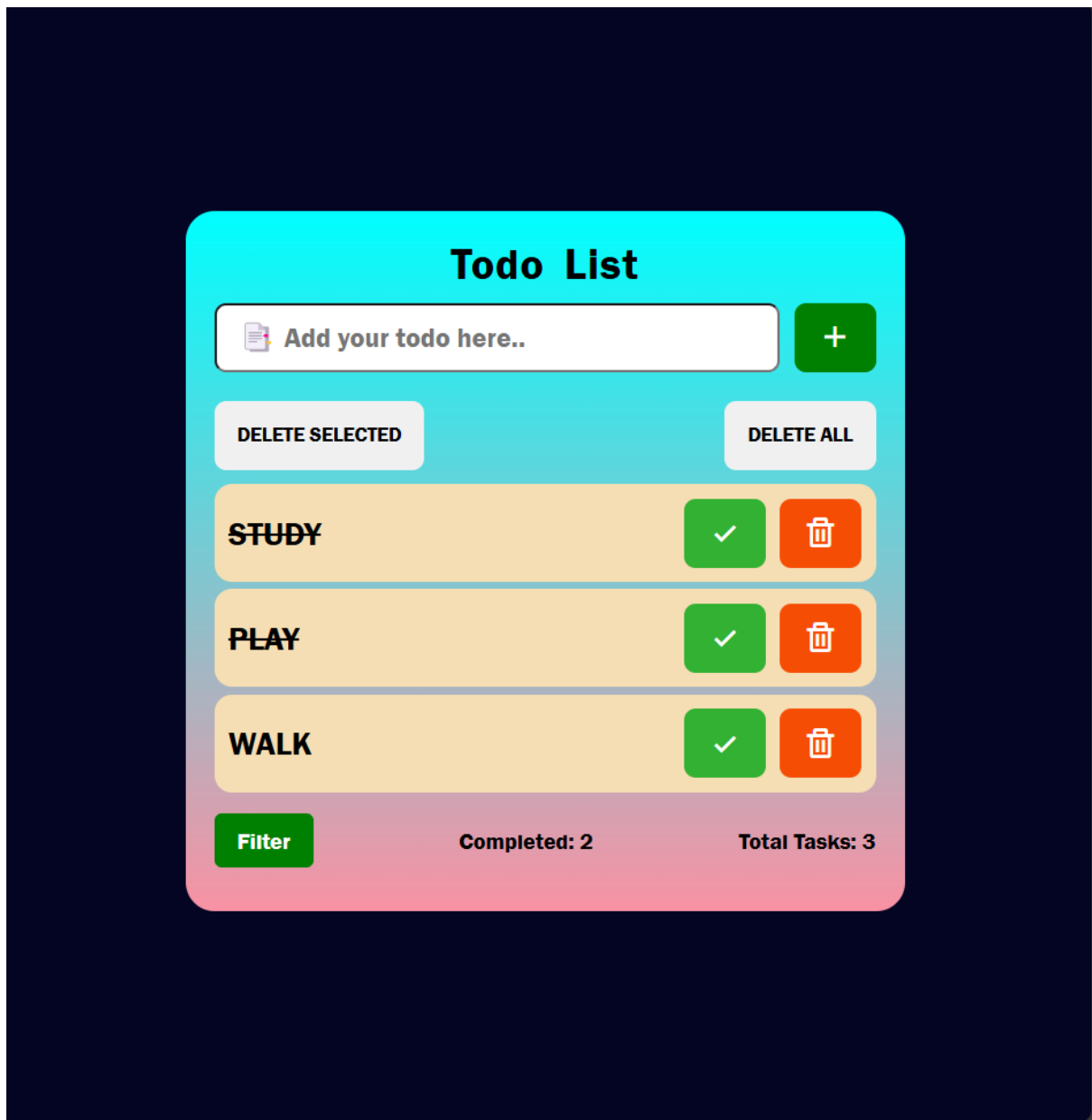
CSS:

JS:

```javascript
var todoList = [];
var comoList = [];
var remList = [];
var addButton = document.getElementById("add-button");
var todoInput = document.getElementById("todo-input");
var deleteAllButton = document.getElementById("delete-all");
var allTodos = document.getElementById("all-todos");
var deleteSButton = document.getElementById("delete-selected");

//event listners for add and delete
addButton.addEventListener("click", add);
deleteAllButton.addEventListener("click", deleteAll);
deleteSButton.addEventListener("click", deleteS);

//event listners for filters
document.addEventListener('click', (e) => {
    if (e.target.className.split(' ')[0] == 'complete' || e.target.className.split(' ')[0] == 'ci') {
        completeTodo(e);
    }
    if (e.target.className.split(' ')[0] == 'delete' || e.target.className.split(' ')[0] == 'di') {
        deleteTodo(e)
    }
    if (e.target.id == "all") {
        viewAll();
    }
    if (e.target.id == "rem") {
        viewRemaining();
    }
    if (e.target.id == "com") {
        viewCompleted();
    }

})
//event listner for enter key
todoInput.addEventListener('keypress', (e) => {
    if (e.key === 'Enter') {
        add();
    }
});

//updates the all the remaining, completed and main list
function update() {
    comoList = todoList.filter((ele) => {
        return ele.complete
    })
    remList = todoList.filter((ele) => {
        return !ele.complete
    })
    document.getElementById("r-count").innerText = todoList.length.toString();
    document.getElementById("c-count").innerText = comoList.length.toString();

}

//adds the task in main list
function add() {
    var value = todoInput.value;
    if (value === '') {
        alert("⚠ Task cannot be empty")
        return;
    }
    todoList.push({
        task: value,
        id: Date.now().toString(),
        complete: false,
    });

    todoInput.value = "";
    update();
    addInmain(todoList);
}

//renders the main list and views on the main content
function addInmain(todoList) {
    allTodos.innerHTML = ""
    todoList.forEach(element => {
        var x = `<li id=${element.id} class="todo-item">
        <p id="task">${element.complete ? `<strike>${element.task}</strike>` : element.task} </p>
        <div class="todo-actions">
            <button class="complete btn btn-success">
```

OUTPUT:

Todo List

📝 Add your todo here..   [+]

[DELETE SELECTED]   [DELETE ALL]

~~STUDY~~   ✓   🗑

~~PLAY~~   ✓   🗑

WALK   ✓   🗑

[Filter]   Completed: 2   Total Tasks: 3

PROJECT 2:

CODE:

HTML:

```html
<html lang="en">
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Advanced Calculator</title>
    <link rel="stylesheet" href="styles.css">
</head>
<body>
    <div class="calculator">
        <div class="display">
            <input type="text" id="result" disabled>
        </div>
        <div class="buttons">
            <button class="span-two" onclick="clearDisplay()">C</button>
            <button onclick="deleteLast()">DEL</button>
            <button onclick="appendToDisplay('/')">÷</button>
            <button onclick="appendToDisplay('7')">7</button>
            <button onclick="appendToDisplay('8')">8</button>
            <button onclick="appendToDisplay('9')">9</button>
            <button onclick="appendToDisplay('*')">x</button>
            <button onclick="appendToDisplay('4')">4</button>
            <button onclick="appendToDisplay('5')">5</button>
            <button onclick="appendToDisplay('6')">6</button>
            <button onclick="appendToDisplay('-')">-</button>
            <button onclick="appendToDisplay('1')">1</button>
            <button onclick="appendToDisplay('2')">2</button>
            <button onclick="appendToDisplay('3')">3</button>
            <button onclick="appendToDisplay('+')">+</button>
            <button onclick="appendToDisplay('0')">0</button>
            <button onclick="appendToDisplay('.')">.</button>
            <button class="span-two" onclick="calculateResult()">=</button>
        </div>
    </div>
    <script src="script.js"></script>
</body>
</html>
```
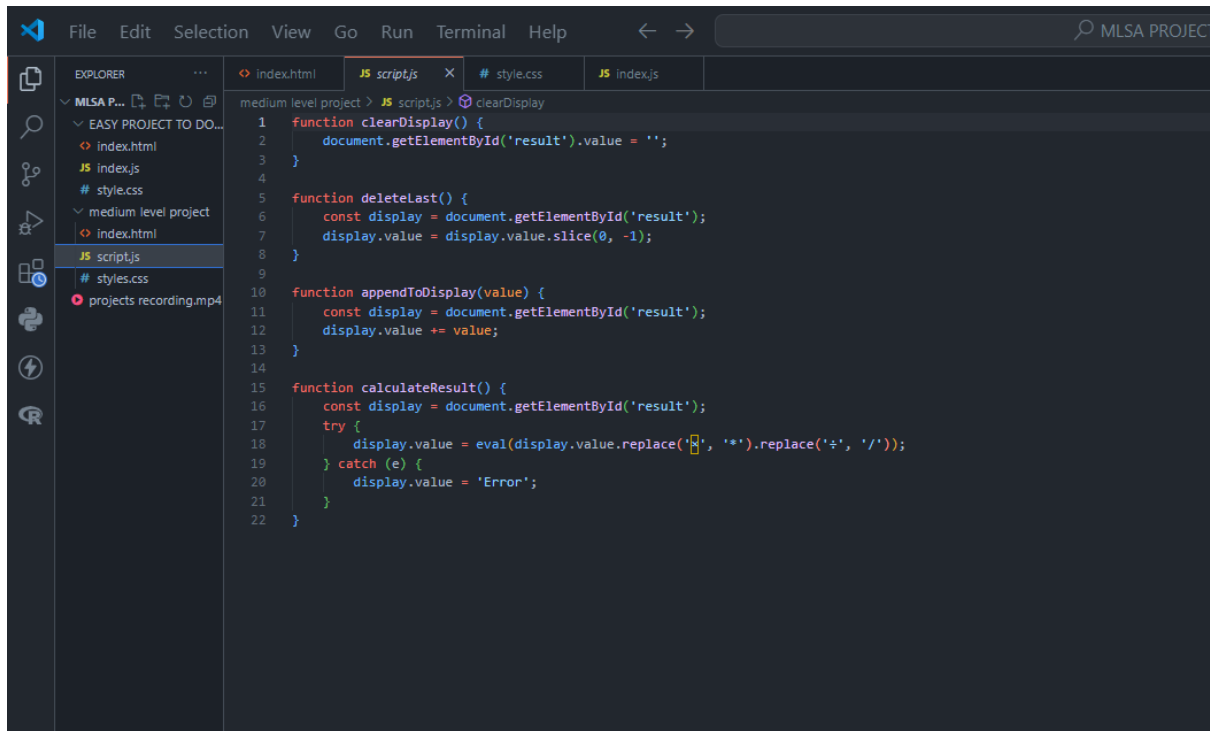
CSS:

```css
.buttons {
    display: grid;
    grid-template-columns: repeat(4, 1fr);
    gap: 1px;
    background: #333;
}

button {
    padding: 20px;
    font-size: 1.5em;
    border: none;
    background: #fff;
    cursor: pointer;
    transition: background 0.2s;
}

button:hover {
    background: #f0f0f0;
}

button:active {
    background: #ddd;
}

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

JS:

```javascript
function clearDisplay() {
    document.getElementById('result').value = '';
}

function deleteLast() {
    const display = document.getElementById('result');
    display.value = display.value.slice(0, -1);
}

function appendToDisplay(value) {
    const display = document.getElementById('result');
    display.value += value;
}

function calculateResult() {
    const display = document.getElementById('result');
    try {
        display.value = eval(display.value.replace('×', '*').replace('÷', '/'));
    } catch (e) {
        display.value = 'Error';
    }
}
```