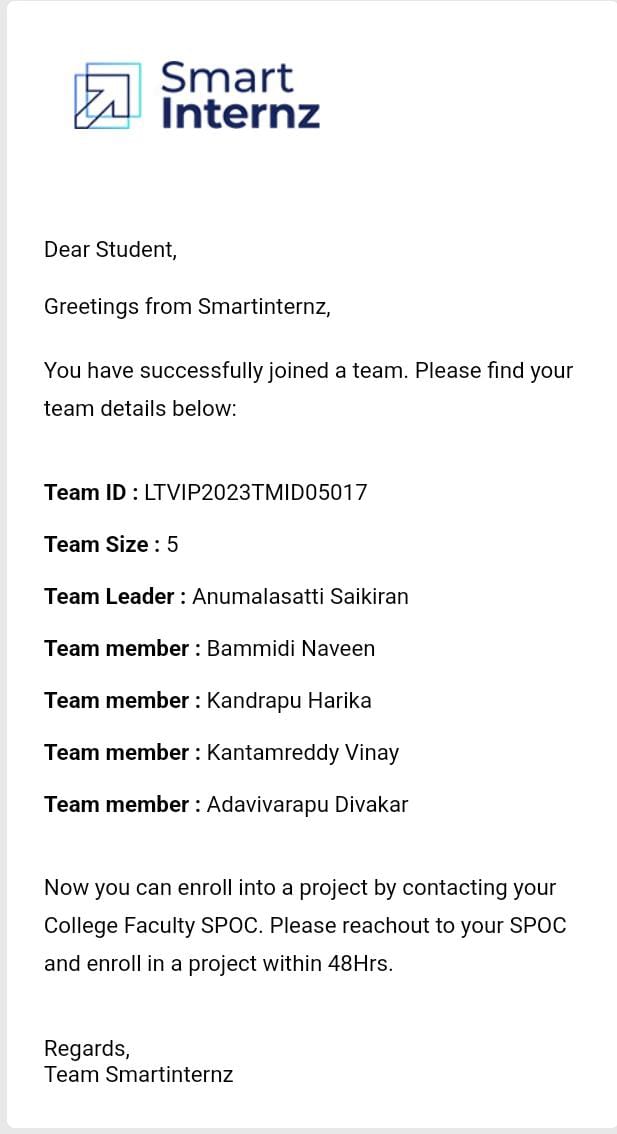
PROJECT

FRONT-END DEVELOPER

ADMINISTRATOR



**PROJECT DESCRIPTION:**

Android To-Do List Project :

Let’s work on a simple To Do List Project, this android project helps beginners to practice their learnings and gain confidence in android development.

### To Do List Project Details:

To-Do List project is an application specially built to keep track of errands or tasks that need to be done. This application will be like a task keeper where the user would be able to enter the tasks that they need to do. Once they are done with their tasks they can also remove them from the list. Let us see what is there in the application:

1. You can add the tasks that are to be done in a descriptive way.
2. You will be able to add as many tasks as you have.
3. Once the task is completed, you will be able to remove it by clicking on “Task Completed” button.

Following are the user interface components that will be shown in our application-

1. There will be an add task button on the right top of the application
2. There will be a button “Task Completed” for each task added, so you will be able to delete it.
3. The tasks that will be added by the users are stored in the list view.

**HTML:**

<!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>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<div class="container">

<h1>To-Do List</h1>

<div class="input-container">

<input type="text" id="taskInput" placeholder="Enter your task here...">

<button onclick="addTask()">Add Task</button>

</div>

<ul id="taskList"></ul>

</div>

<script src="script.js"></script>

</body>

</html>

**CSS :**

body {

font-family: Arial, sans-serif;

background-color: #f5f5f5;

margin: 0;

padding: 0;

}

.container {

max-width: 500px;

margin: 30px auto;

padding: 20px;

background-color: #fff;

border-radius: 5px;

box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);

}

h1 {

text-align: center;

}

.input-container {

display: flex;

margin-bottom: 20px;

}

input[type="text"] {

flex: 1;

padding: 10px;

font-size: 16px;

}

button {

padding: 10px 20px;

font-size: 16px;

background-color: #4CAF50;

color: #fff;

border: none;

cursor: pointer;

border-radius: 5px;

}

button:hover {

background-color: #45a049;

}

ul {

list-style: none;

padding: 0;

}

li {

display: flex;

align-items: center;

padding: 10px;

border-bottom: 1px solid #ddd;

}

li.completed {

text-decoration: line-through;

}

li .tick {

margin-right: 10px;

}

**JAVASCRIPT :**

const taskList = document.getElementById("taskList");

const taskInput = document.getElementById("taskInput");

let taskCounter = 1;

function addTask() {

const taskText = taskInput.value.trim();

if (taskText !== "") {

const li = document.createElement("li");

const taskNumber = document.createElement("span");

taskNumber.classList.add("task-number");

taskNumber.textContent = taskCounter++;

const taskTextElement = document.createElement("span");

taskTextElement.textContent = taskText;

const checkbox = document.createElement("input");

checkbox.type = "checkbox";

checkbox.classList.add("task-checkbox");

li.appendChild(taskNumber);

li.appendChild(document.createTextNode("\u00A0")); // Add a non-breaking space

li.appendChild(taskTextElement);

li.appendChild(document.createTextNode("\u00A0")); // Add another non-breaking space

li.appendChild(checkbox);

taskList.appendChild(li);

taskInput.value = "";

taskInput.focus();

checkbox.addEventListener("change", () => {

li.classList.toggle("completed", checkbox.checked);

});

}

}

taskInput.addEventListener("keyup", (event) => {

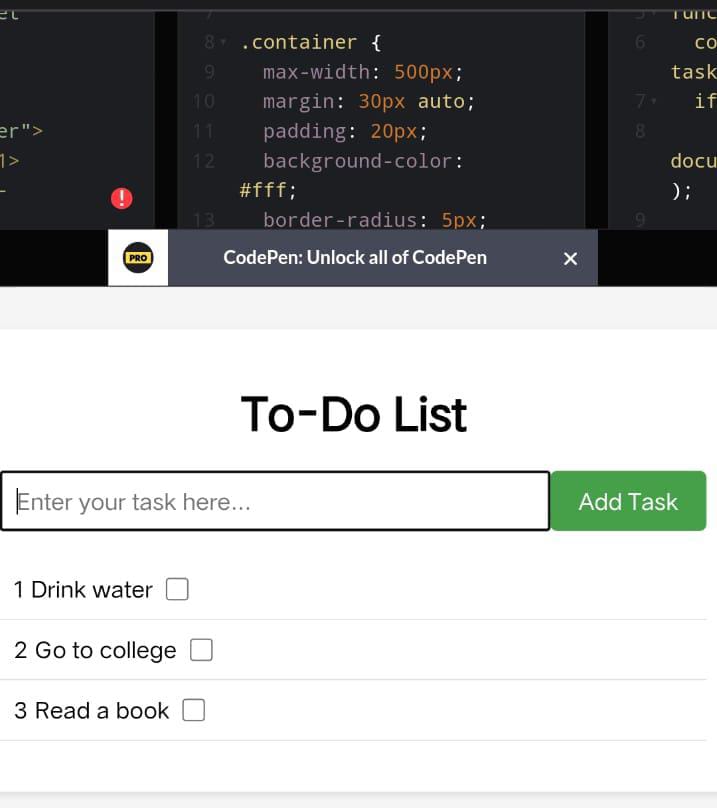
if (event.key === "Enter") {

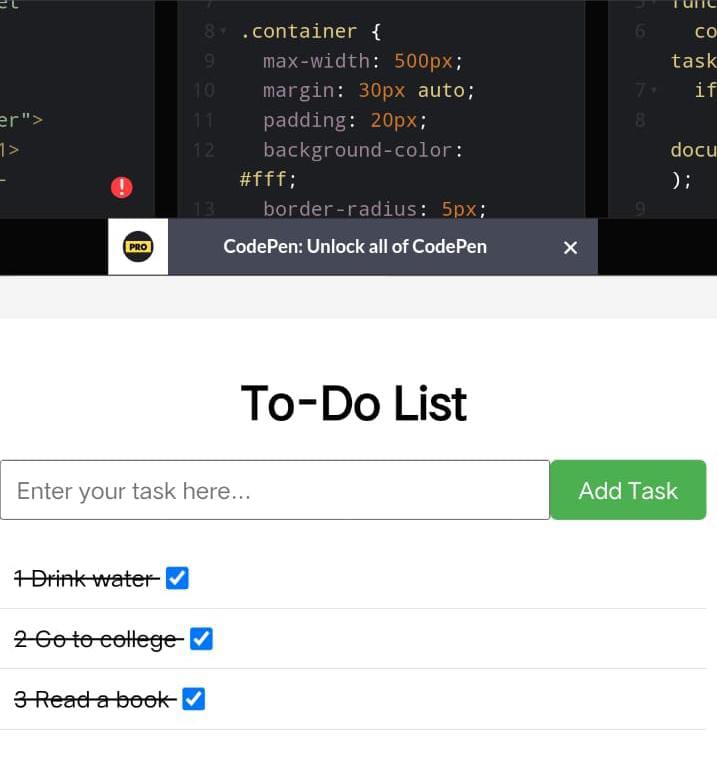
addTask();

}

});

**OUTPUT:**

****

****