**TASK 4**

***HTML***

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>To-Do List App</title>

    <link rel="stylesheet" href="task4.css">

</head>

<body>

    <div class="container">

    <h1>📝 To-Do List</h1>

<div class="task-input">

    <input type="text" id="taskText" placeholder="Enter your task">

    <input type="datetime-local" id="taskDateTime">

    <button id="addTaskBtn">Add Task</button>

    </div>

    <ul id="taskList"></ul>

</div>

<script src="task4.js"></script>

</body>

</html>

***CSS***

body {

    font-family: Arial, sans-serif;

    background: #f2f2f2;

    display: flex;

    justify-content: center;

    padding-top: 50px;

}

.container {

    background: white;

    padding: 30px;

    border-radius: 10px;

    box-shadow: 0 4px 10px rgba(0,0,0,0.1);

    width: 400px;

}

h1 {

    text-align: center;

}

.task-input {

    display: flex;

    flex-direction: column;

    gap: 10px;

}

.task-input input,

.task-input button {

    padding: 10px;

    font-size: 16px;

}

.task-input button {

    background-color: #007bff;

    color: white;

    border: none;

    cursor: pointer;

}

.task-input button:hover {

    background-color: #0056b3;

}

ul {

    list-style: none;

    padding: 0;

    margin-top: 20px;

}

li {

    background: #f9f9f9;

    margin-bottom: 10px;

    padding: 15px;

    border-radius: 8px;

    display: flex;

    flex-direction: column;

    gap: 5px;

    position: relative;

}

li.completed {

    text-decoration: line-through;

    background: #d4edda;

}

.task-actions {

    display: flex;

    gap: 10px;

    margin-top: 8px;

}

button.action {

    padding: 5px 10px;

    font-size: 14px;

    border: none;

    cursor: pointer;

}

button.edit { background-color: #ffc107; }

button.done { background-color: #28a745; color: white; }

button.delete { background-color: #dc3545; color: white; }

***JS***

let tasks = [];

document.addEventListener('DOMContentLoaded', () => {

    document.getElementById('addTaskBtn').addEventListener('click', addTask);

});

function addTask() {

    const taskText = document.getElementById('taskText').value.trim();

    const taskDateTime = document.getElementById('taskDateTime').value;

if (!taskText) {

    alert('Please enter a task');

    return;

}

const task = {

    id: Date.now(),

    text: taskText,

    datetime: taskDateTime,

    completed: false

};

    tasks.push(task);

    document.getElementById('taskText').value = '';

    document.getElementById('taskDateTime').value = '';

    renderTasks();

}

function renderTasks() {

    const list = document.getElementById('taskList');

    list.innerHTML = '';

tasks.forEach(task => {

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

    li.className = task.completed ? 'completed' : '';

    const taskText = document.createElement('span');

    taskText.textContent = `${task.text} (${task.datetime ? formatDateTime(task.datetime) : 'No date'})`;

    li.appendChild(taskText);

    const actions = document.createElement('div');

    actions.className = 'task-actions';

    const editBtn = document.createElement('button');

    editBtn.className = 'action edit';

    editBtn.textContent = 'Edit';

    editBtn.onclick = () => editTask(task.id);

    const doneBtn = document.createElement('button');

    doneBtn.className = 'action done';

    doneBtn.textContent = task.completed ? 'Undo' : 'Done';

    doneBtn.onclick = () => toggleComplete(task.id);

    const deleteBtn = document.createElement('button');

    deleteBtn.className = 'action delete';

    deleteBtn.textContent = 'Delete';

    deleteBtn.onclick = () => deleteTask(task.id);

    actions.appendChild(editBtn);

    actions.appendChild(doneBtn);

    actions.appendChild(deleteBtn);

    li.appendChild(actions);

    list.appendChild(li);

});

}

function editTask(id) {

    const task = tasks.find(t => t.id === id);

    const newText = prompt('Edit your task:', task.text);

if (newText !== null && newText.trim() !== '') {

    task.text = newText.trim();

    renderTasks();

}

}

function toggleComplete(id) {

    const task = tasks.find(t => t.id === id);

    task.completed = !task.completed;

    renderTasks();

}

function deleteTask(id) {

    tasks = tasks.filter(t => t.id !== id);

    renderTasks();

}

function formatDateTime(dateTimeStr) {

    const date = new Date(dateTimeStr);

    return date.toLocaleString();

}