<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta name="description" content="A modern, responsive, and SEO-friendly todo list web application.">

<meta name="keywords" content="todo list, task manager, productivity, online todo list">

<meta name="author" content="Your Name">

<title>Online Todo List - Manage Your Tasks Efficiently</title>

<!-- Bootstrap CSS -->

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet">

<!-- Custom CSS -->

<style>

body {

background-color: #f8f9fa;

font-family: 'Arial', sans-serif;

}

.todo-container {

max-width: 600px;

margin: 50px auto;

padding: 20px;

background: #fff;

border-radius: 10px;

box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);

}

.todo-header {

text-align: center;

margin-bottom: 20px;

}

.todo-header h1 {

font-size: 2.5rem;

color: #333;

}

.todo-input {

display: flex;

gap: 10px;

margin-bottom: 20px;

}

.todo-input input {

flex: 1;

padding: 10px;

border: 1px solid #ddd;

border-radius: 5px;

}

.todo-input button {

padding: 10px 20px;

background-color: #28a745;

color: #fff;

border: none;

border-radius: 5px;

cursor: pointer;

}

.todo-input button:hover {

background-color: #218838;

}

.todo-list {

list-style: none;

padding: 0;

}

.todo-item {

display: flex;

justify-content: space-between;

align-items: center;

padding: 10px;

border: 1px solid #ddd;

border-radius: 5px;

margin-bottom: 10px;

background: #f9f9f9;

}

.todo-item.completed {

text-decoration: line-through;

opacity: 0.7;

}

.todo-item .actions {

display: flex;

gap: 10px;

}

.todo-item .actions button {

background: none;

border: none;

cursor: pointer;

font-size: 1rem;

}

.todo-item .actions .edit-btn {

color: #ffc107;

}

.todo-item .actions .delete-btn {

color: #dc3545;

}

.todo-item .actions .complete-btn {

color: #28a745;

}

.footer {

text-align: center;

margin-top: 20px;

font-size: 0.9rem;

color: #666;

}

</style>

</head>

<body>

<div class="todo-container">

<div class="todo-header">

<h1>Online Todo List</h1>

</div>

<div class="todo-input">

<input type="text" id="taskInput" placeholder="Add a new task...">

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

</div>

<ul class="todo-list" id="todoList"></ul>

<div class="footer">

<p>© 2025 Online Todo List. All rights reserved.</p>

</div>

</div>

<!-- Bootstrap JS and Popper.js -->

<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.6/dist/umd/popper.min.js"></script>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.min.js"></script>

<!-- Custom JS -->

<script>

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

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

const addTaskBtn = document.getElementById('addTaskBtn');

const todoList = document.getElementById('todoList');

// Load tasks from local storage

let tasks = JSON.parse(localStorage.getItem('tasks')) || [];

// Render tasks

function renderTasks() {

todoList.innerHTML = '';

tasks.forEach((task, index) => {

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

li.className = `todo-item ${task.completed ? 'completed' : ''}`;

li.innerHTML = `

<span>${task.text}</span>

<div class="actions">

<button class="complete-btn" onclick="toggleComplete(${index})">✔️</button>

<button class="edit-btn" onclick="editTask(${index})">✏️</button>

<button class="delete-btn" onclick="deleteTask(${index})">❌</button>

</div>

`;

todoList.appendChild(li);

});

}

// Add task

addTaskBtn.addEventListener('click', () => {

const text = taskInput.value.trim();

if (text !== '') {

tasks.push({ text, completed: false });

taskInput.value = '';

saveTasks();

renderTasks();

}

});

// Toggle task completion

window.toggleComplete = (index) => {

tasks[index].completed = !tasks[index].completed;

saveTasks();

renderTasks();

};

// Edit task

window.editTask = (index) => {

const newText = prompt('Edit your task:', tasks[index].text);

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

tasks[index].text = newText.trim();

saveTasks();

renderTasks();

}

};

// Delete task

window.deleteTask = (index) => {

tasks.splice(index, 1);

saveTasks();

renderTasks();

};

// Save tasks to local storage

function saveTasks() {

localStorage.setItem('tasks', JSON.stringify(tasks));

}

// Initial render

renderTasks();

});

</script>

</body>

</html>