

```
<!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 App</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background: #0f172a;
      color: #fff;
      margin: 0;
      display: flex;
      justify-content: center;
      align-items: flex-start;
      min-height: 100vh;
      padding: 30px;
    }

    .container {
      background: #1e293b;
      padding: 20px;
      border-radius: 12px;
      width: 100%;
      max-width: 420px;
      box-shadow: 0 4px 20px rgba(0,0,0,0.3);
    }

    h1 {
      text-align: center;
      margin-bottom: 20px;
      color: #38bdf8;
    }

    .input-row {
      display: flex;
      gap: 10px;
    }

    input[type="text"] {
      flex: 1;
      padding: 10px;
      border-radius: 6px;
      border: none;
```

```
outline: none;
background: #334155;
color: #fff;
}
```

```
button {
  background: #38bdf8;
  border: none;
  border-radius: 6px;
  padding: 10px 16px;
  color: #0f172a;
  font-weight: bold;
  cursor: pointer;
  transition: background 0.2s;
}
```

```
button:hover {
  background: #0ea5e9;
}
```

```
ul {
  list-style: none;
  padding: 0;
  margin-top: 20px;
}
```

```
li {
  background: #334155;
  margin-bottom: 10px;
  padding: 10px;
  border-radius: 6px;
  display: flex;
  justify-content: space-between;
  align-items: center;
  transition: background 0.3s;
}
```

```
li.completed {
  text-decoration: line-through;
  opacity: 0.6;
}
```

```
.btn-group {
  display: flex;
```

```

    gap: 6px;
}

.edit-btn {
    background: #22c55e;
    color: #fff;
    padding: 6px 10px;
    border-radius: 5px;
    cursor: pointer;
    border: none;
}

.delete-btn {
    background: #ef4444;
    color: #fff;
    padding: 6px 10px;
    border-radius: 5px;
    cursor: pointer;
    border: none;
}


.edit-btn:hover {
    background: #16a34a;
}

.delete-btn:hover {
    background: #dc2626;
}

.footer {
    text-align: center;
    font-size: 13px;
    color: #94a3b8;
    margin-top: 20px;
}
</style>
</head>
<body>
<div class="container">
    <h1>To-Do List</h1>
    <div class="input-row">
        <input type="text" id="taskInput" placeholder="Enter a new task..." />
        <button id="addBtn">Add</button>
    </div>

```

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

```
<div class="footer">Your tasks are saved locally </div>
</div>
```

```
<script>
```

```
const taskInput = document.getElementById('taskInput');
const addBtn = document.getElementById('addBtn');
const taskList = document.getElementById('taskList');
```

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

```
function saveTasks() {
  localStorage.setItem('tasks', JSON.stringify(tasks));
}
```

```
function renderTasks() {
  taskList.innerHTML = '';
  tasks.forEach((task, index) => {
    const li = document.createElement('li');
    li.textContent = task.text;
    if (task.completed) li.classList.add('completed');

    li.addEventListener('click', () => toggleComplete(index));
```

```
const btnGroup = document.createElement('div');
btnGroup.className = 'btn-group';
```

```
const editBtn = document.createElement('button');
editBtn.textContent = 'Edit';
editBtn.className = 'edit-btn';
editBtn.addEventListener('click', (e) => {
  e.stopPropagation();
  editTask(index);
});
```

```
const delBtn = document.createElement('button');
delBtn.textContent = 'Delete';
delBtn.className = 'delete-btn';
delBtn.addEventListener('click', (e) => {
  e.stopPropagation();
  deleteTask(index);
});
```

```

    btnGroup.appendChild(editBtn);
    btnGroup.appendChild(delBtn);
    li.appendChild(btnGroup);
    taskList.appendChild(li);
  });
}

```

```

function addTask() {
  const text = taskInput.value.trim();
  if (text === "") return;
  tasks.push({ text, completed: false });
  saveTasks();
  renderTasks();
  taskInput.value = "";
}

```

```

function toggleComplete(index) {
  tasks[index].completed = !tasks[index].completed;
  saveTasks();
  renderTasks();
}

```

```

function deleteTask(index) {
  tasks.splice(index, 1);
  saveTasks();
  renderTasks();
}

```

```

function editTask(index) {
  const newText = prompt('Edit your task:', tasks[index].text);
  if (newText !== null && newText.trim() !== "") {
    tasks[index].text = newText.trim();
    saveTasks();
    renderTasks();
  }
}

```

```

addBtn.addEventListener('click', addTask);
taskInput.addEventListener('keypress', (e) => {
  if (e.key === 'Enter') addTask();
});

```

```

renderTasks();

```

```
</script>  
</body>  
</html>
```