

Mannat Nandi Se2/B3/45

Experiment 3

Aim:

Create a dynamic To-Do List where users can add, mark as complete, and remove tasks using JavaScript & DOM.

Code:

```
<!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>
  <style>
    * {
      margin: 0;
      padding: 0;
      box-sizing: border-box;
    }

    body {
      font-family: Arial, sans-serif;
      background-color: #f0e6f6;
      display: flex;
      justify-content: center;
      align-items: center;
      min-height: 100vh;
      padding: 20px;
    }

    .container {
      background-color: #fff5f8;
      padding: 40px;
      border-radius: 15px;
      box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
      width: 100%;
      max-width: 500px;
    }

    h1 {
      text-align: center;
      color: #9b59b6;
      margin-bottom: 30px;
      font-size: 32px;
```

Mannat Nandi Se2/B3/45

```
}

.input-section {
  display: flex;
  gap: 10px;
  margin-bottom: 25px;
}

input[type="text"] {
  flex: 1;
  padding: 12px 15px;
  border: 2px solid #d4a5d4;
  border-radius: 8px;
  font-size: 16px;
  outline: none;
  background-color: #fef9ff;
}

input[type="text"]:focus {
  border-color: #9b59b6;
}

button {
  padding: 12px 24px;
  border: none;
  border-radius: 8px;
  cursor: pointer;
  font-size: 16px;
  transition: background-color 0.3s;
}

.add-btn {
  background-color: #b4a7d6;
  color: white;
}

.add-btn:hover {
  background-color: #9b8bc4;
}

.task-list {
  list-style: none;
}

.task-item {
  background-color: #e8f4f8;
```

Mannat Nandi Se2/B3/45

```
padding: 15px;
margin-bottom: 10px;
border-radius: 8px;
display: flex;
justify-content: space-between;
align-items: center;
border: 2px solid #c8e3f0;
}

.task-item.completed {
  background-color: #d4f1d4;
  border-color: #a8d8a8;
}

.task-item.completed .task-text {
  text-decoration: line-through;
  color: #7a9b7a;
}

.task-text {
  flex: 1;
  font-size: 16px;
  color: #333;
}

.task-buttons {
  display: flex;
  gap: 8px;
}

.complete-btn {
  background-color: #a8d8a8;
  color: white;
  padding: 8px 16px;
}

.complete-btn:hover {
  background-color: #8ec68e;
}

.delete-btn {
  background-color: #f5b7b1;
  color: white;
  padding: 8px 16px;
}
```

Mannat Nandi Se2/B3/45

```
.delete-btn:hover {
  background-color: #e89e98;
}

.empty-message {
  text-align: center;
  color: #999;
  padding: 20px;
  font-style: italic;
}
</style>
</head>
<body>
  <div class="container">
    <h1>My To-Do List</h1>

    <div class="input-section">
      <input type="text" id="taskInput" placeholder="Enter a new task...">
      <button class="add-btn" onclick="addTask()">Add Task</button>
    </div>

    <ul class="task-list" id="taskList">
      <li class="empty-message">No tasks yet. Add one above!</li>
    </ul>
  </div>

  <script>
    let tasks = [];

    function addTask() {
      const input = document.getElementById('taskInput');
      const taskText = input.value.trim();

      if (taskText === "") {
        alert('Please enter a task!');
        return;
      }

      const task = {
        id: Date.now(),
        text: taskText,
        completed: false
      };

      tasks.push(task);
      input.value = "";
    }
  </script>
</body>
</html>
```

Mannat Nandi Se2/B3/45

```
    renderTasks();
  }

  function toggleComplete(id) {
    tasks = tasks.map(task =>
      task.id === id ? { ...task, completed: !task.completed } : task
    );
    renderTasks();
  }

  function deleteTask(id) {
    tasks = tasks.filter(task => task.id !== id);
    renderTasks();
  }

  function renderTasks() {
    const taskList = document.getElementById('taskList');

    if (tasks.length === 0) {
      taskList.innerHTML = '<li class="empty-message">No tasks yet. Add one
above!</li>';
      return;
    }

    taskList.innerHTML = tasks.map(task => `
      <li class="task-item ${task.completed ? 'completed' : ''}>
        <span class="task-text">${task.text}</span>
        <div class="task-buttons">
          <button class="complete-btn" onclick="toggleComplete(${task.id})">
            ${task.completed ? 'Undo' : 'Complete'}
          </button>
          <button class="delete-btn" onclick="deleteTask(${task.id})">Delete</button>
        </div>
      </li>
    `).join("");
  }

  document.getElementById('taskInput').addEventListener('keypress', function(e) {
    if (e.key === 'Enter') {
      addTask();
    }
  });
</script>
</body>
</html>
```

Mannat Nandi Se2/B3/45

Output:

The image displays two screenshots of a web application titled "My To-Do List". The application has a light purple background and a central white card with rounded corners and a subtle shadow.

Top Screenshot: The card contains the title "My To-Do List" in bold purple text. Below the title is a text input field with the placeholder text "Enter a new task...". To the right of the input field is a purple button with the text "Add Task". Below the input field, the text "No tasks yet. Add one above!" is displayed in a small, italicized font.

Bottom Screenshot: The card is identical to the top one, but the text input field now contains the text "mannat nandi". The "Add Task" button and the text below it remain the same.

My To-Do List

Add Task

mannat nandi

Complete

Delete

My To-Do List

Add Task

mannat nandi

Undo

Delete