

Name:MUHAMMAD IBRAHIM S S

Roll Number:11221029

Create a Todo app 1. Add and complete a task on a single page 2. Uses the Express Framework

Step 1: Create the Project Directory

1. Open **Terminal**.
2. Navigate to the folder where you want to create your project:

```
cd ~/Documents
```

3. Create a folder named todo-app and navigate into it:

```
mkdir todo-app
```

```
cd todo-app
```

Step 2: Initialize the Project

1. Run the following command to create a basic package.json file:

```
npm init -y
```

Step 3: Create Project Files

You'll create three files: server.js, views/index.ejs, and public/style.css.

a. Create server.js

1. Create the file:

```
touch server.js
```

2. Open it in **Visual Studio Code** (if installed):

```
code server.js
```

Or use the built-in TextEdit:

```
open -e server.js
```

3. Add the following code to server.js and save:

```
const express = require("express");
```

```
const bodyParser = require("body-parser");
```

```
const app = express();
```

```
const port = 3000;
```

```
app.use(bodyParser.urlencoded({ extended: true }));
```

```
app.use(express.static("public"));
```

```
app.set("view engine", "ejs");
```

```
let tasks = [];
```

```
app.get("/", (req, res) => {
```

```
  res.render("index", { tasks });
```

```
});
```

```
app.post("/add", (req, res) => {
```

```
  const task = req.body.task.trim();
```

```
  if (task) {
```

```
    tasks.push({ text: task, completed: false });  
  
  }  
  
  res.redirect("/");  
});
```

```
app.post("/complete", (req, res) => {  
  const taskIndex = parseInt(req.body.index);  
  
  if (tasks[taskIndex]) {  
    tasks[taskIndex].completed = true;  
  }  
  
  res.redirect("/");  
});
```

```
app.listen(port, () => {  
  console.log(`Todo app running at http://localhost:${port}`);  
});
```

b. Create views/index.ejs

1. Create the folder views:

```
mkdir views
```

2. Create the file:

```
touch index.ejs
```

3. Open the file:

```
code index.ejs
```

Or:

```
open -e index.ejs
```

4. Add the following code and save:

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

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

  <title>Todo App</title>

  <link rel="stylesheet" href="/style.css">

</head>

<body>

  <div class="container">

    <h1>Todo App</h1>

    <form action="/add" method="POST">

      <input type="text" name="task" placeholder="Enter a task" required>

      <button type="submit">Add</button>

    </form>

    <ul>

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

        <li class="<%= task.completed ? 'completed' : " %>">

          <span><%= task.text %></span>

          <% if (!task.completed) { %>

            <form action="/complete" method="POST" style="display: inline;">

              <input type="hidden" name="index" value="<%= index %>">

              <button type="submit">Complete</button>

            </form>

          <% }
```

```
<% } else { %>

<span>(Done)</span>

<% } %>

</li>

<% }) %>

</ul>

</div>

</body>

</html>
```

c. Create public/style.css

1. Create the folder public:

```
mkdir public
```

2. Create the file:

```
touch style.css
```

3. Open the file:

```
code style.css
```

Or:

```
open -e style.css
```

4. Add the following code and save:

```
body {

font-family: Arial, sans-serif;

background-color: #f9f9f9;

margin: 0;

padding: 0;

}
```

```
.container {  
  
  max-width: 600px;  
  
  margin: 50px auto;  
  
  padding: 20px;  
  
  background: #fff;  
  
  border-radius: 8px;  
  
  box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);  
  
}
```

```
h1 {  
  
  text-align: center;  
  
  color: #333;  
  
}
```

```
form {  
  
  display: flex;  
  
  justify-content: space-between;  
  
  margin-bottom: 20px;  
  
}
```

```
input[type="text"] {  
  
  flex: 1;  
  
  padding: 10px;  
  
  border: 1px solid #ccc;
```

```
border-radius: 4px;  
}
```

```
button {  
  
    background-color: #007bff;  
  
    color: #fff;  
  
    border: none;  
  
    padding: 10px 20px;  
  
    border-radius: 4px;  
  
    cursor: pointer;  
}
```

```
ul {  
  
    list-style: none;  
  
    padding: 0;  
}
```

```
li {  
  
    padding: 10px;  
  
    border-bottom: 1px solid #ddd;  
}
```

```
li.completed span {  
  
    text-decoration: line-through;  
  
    color: #999;  
}
```

```
}
```

Step 4: Run Your App

1. Start the server:

```
node server.js
```

2. Open your browser and visit:

<http://localhost:3000>

Output:

