🌱 Case Study: Building a Simple Nursery Landing Page in React
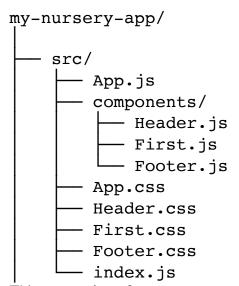
🎯 Objective

The goal of this case study is to teach beginners how to **structure a React application using components** and style them with CSS.
We will build a **Nursery Landing Page** that has:

- **Header Component** → Displays the nursery name.

- **First Component (Main Content)** → Shows a welcome message, a dropdown for selecting location, and a list of flowers.

- **Footer Component** → Displays copyright.

🏗️ Project Structure

Your React project will be organized like this:

```
my-nursery-app/
│
├── src/
│   ├── App.js
│   ├── components/
│   │   ├── Header.js
│   │   ├── First.js
│   │   └── Footer.js
│   ├── App.css
│   ├── Header.css
│   ├── First.css
│   ├── Footer.css
│   └── index.js
```
This **separation of components and CSS files** makes the project clean and beginner-friendly.

🔷 Step 1: Header Component

- This component is responsible for showing the **title of the nursery application**.

- The heading will be displayed at the top in a **bold, centered style**.

- CSS should include:

- A **dark green background** to give a natural feel.

- **White-colored text** for contrast.

- Padding for spacing.

- Center alignment so the heading looks neat.

🔷 Step 2: First Component (Main Content)

This is the most important part where we display the **Nursery Page details**.

It should contain:

1. **Message** → "Creating the Nursery Page" in a bold and colorful heading.

2. **Paragraph** → A welcome message to the nursery landing page.

3. **Dropdown (Location Selection)** → A list of cities such as *Chennai, Bangalore, Mumbai, Delhi*.

   - The dropdown should be styled with padding, margin, and rounded corners.

4. **List of Flowers** → Flowers like *Rose, Lily, Tulip, Sunflower, Jasmine*.

   - The list should use **circle or square bullets**.

   - Each item should be styled in **dark green color** to match the nursery theme.

CSS Implementation Details:

- Give the main container a **light green background** to create a nursery feel.

- Add a **border with rounded corners** for a soft look.

- Center-align all the content for readability.

- Use **different font colors**:

   - Heading → Dark green

   - Paragraph → Grey or black

   - List items → Dark green or bold for emphasis.

- Add spacing (margin and padding) so the content doesn't look cramped.

🔷 Step 3: Footer Component

- The footer should display **© 2025 My Nursery App | All Rights Reserved**.

- CSS styling for the footer:

  - Use a **dark background** (similar to the header for consistency).

  - Keep the text color **white**.

  - Add padding and center alignment.

  - Place it at the **bottom of the page**.

🔷 Step 4: App Component (Combining Everything)

- The `App.js` file should **import and render Header, First, and Footer** in order.

- The layout will look like this:

  - **Header at the top**

  - **First Component (main content) in the middle**

  - **Footer at the bottom**

🖼️ Final Output

1. **Header Section**

   - A dark green bar with white heading text → "My Nursery Application".

2. **Main Content (First Component)**

   - A light green background box with:

     - Heading: "Creating the Nursery Page"

     - A short welcome message

     - A dropdown to select a location

     - A flower list (Rose, Lily, Tulip, Sunflower, Jasmine)

3. **Footer Section**

   - A dark bar at the bottom with © 2025 My Nursery App.