1. **Render Nested JSX Elements**
   - **Problem:**
     Create a JSX structure for a blog post that includes:
     - A `div` with a `className` of "blog-post".
     - An `h1` for the title, a `p` for the content, and a nested `div` with a list of tags.
     - Use an array of tags to dynamically render the list.

Input:

```
const title = "Understanding JSX";
const content = "JSX allows you to write HTML in React.";
const tags = ["React", "JavaScript", "JSX"];
```

2. **Dynamic Table Rendering**
   - **Problem:**
     Create a JSX component that renders a table dynamically using an array of objects. Each object should represent a row with `name`, `age`, and `location`.

`Input:`

```
const data = [
  { name: "Alice", age: 25, location: "New York" },
  { name: "Bob", age: 30, location: "San Francisco" },
  { name: "Charlie", age: 35, location: "Los Angeles" }
];
```

3. **Render a Grid Layout**

Problem:

Create a JSX component that dynamically renders a grid of items. Use an array of product names and render each name inside a `div` with a `className` of "grid-item". The parent `div` should have a `className` of "grid-container".

Input:

```
const products = ["Laptop", "Smartphone", "Tablet", "Smartwatch",
"Camera"];
```

4. **Render a Form Dynamically**
    - **Problem:**
      Create a JSX component that generates a form dynamically based on an array of field definitions. Each field object should have a `label`, `type`, and `name`. Render appropriate form elements for `text`, `email`, and `password`.

Input:

```
const fields = [
  { label: "Username", type: "text", name: "username" },
  { label: "Email", type: "email", name: "email" },
  { label: "Password", type: "password", name: "password" }
];
```

5. **Conditional Styling Based on Props**
    - **Problem:**
      Create a JSX component that accepts a `status` prop and renders a `div` with dynamic text and styles.
        - If `status` is "success", display "Operation Successful" with a green background.
        - If `status` is "error", display "Operation Failed" with a red background.
        - Otherwise, display "Pending..." with a yellow background.

Input:

```
<StatusMessage status="success" />
```

6 . Dynamic Theme Switcher

- **Problem:**
  Create a `ThemeSwitcher` component that toggles between "Light Mode" and "Dark Mode". Change the background color of the `div` accordingly.
- **Expected Output:**
  A button that switches between light and dark themes.

7. Search Filter Component

- **Problem:**
  Create a `SearchBar` component that accepts a list of names and a search input. Filter and display names based on the input.

  Input:

```
<SearchBar names={['Alice', 'Bob', 'Charlie']} />
```

8. Profile Card Component

- **Description:**
  A card that displays a user's name, profile picture, and a short bio.
- **Features:**
  - Accepts `name`, `bio`, and `profileImage` as props.

9. Product Card

- **Description:**
  Displays a product image, name, price, and an "Add to Cart" button.
- **Features:**
  - `productName`, `productPrice`, and `productImage`.

10. Image Gallery

- **Description:**
  Create a gallery component that displays a grid of images.
- **Features:**
  - Accepts an array of image URLs as props.

11. Todo List

- **Description:**
  A simple component to add and display a list of tasks.
- **Features:**
  - Input field to add tasks.
  - Displays tasks in a list format.
- **Expected Output:**
  A task list with functionality to add new tasks.

12. Login Form

- **Description:**
  A simple form with username and password fields.
- **Features:**
  - Submit button.
  - Displays entered username on submission.
- **Expected Output:**
  A form that validates and displays the username entered.

13. Rating Component

- **Description:**
  A component where users can click on stars to give a rating.
- **Features:**
  - Props: `maxStars`.
- **Expected Output:**
  A row of stars where users can select their rating.

14. News Article Component

- **Description:**
  Displays a news headline, short description, and an image.
- **Features:**
  - Props: `headline`, `description`, and `articleImage`.

15. Weather Card

- **Description:**
  Displays current weather details for a city.
- **Features:**
  - Props: `city`, `temperature`, and `weatherIcon`.

Instructions:

1.Try to create the components an show the output in ui for all questions

2.Try to achieve the look and feel layout by using css