React JS Stateless Component Assignment - Rendering Arrays and Array of Objects

---

Objective:
In this assignment, you will create a React functional (stateless) component that receives an array of objects and renders them in both a table format and a list format.

Requirements:
1. Create a new React project using create-react-app.
2. Create a JavaScript array of objects. Each object should contain the following properties:
   - **name** (string)
   - **email** (string)
   - **age** (integer)

You can define this array in a separate **data.js** file and import it into your main application file.

3. Create a functional component called **UserTable**. This component should receive the array of objects as a prop and render it as a table. Each object in the array should correspond to a row in the table.
4. Create a functional component called **UserList**. This component should also receive the array of objects as a prop, but it should render it as an unordered list. Each object in the array should correspond to a list item.
5. Use CSS or a CSS-in-JS library to style your components. You should at least change the default font, add some padding and margins to your elements, and alternate the background color of your table rows.
6. Finally, render both **UserTable** and **UserList** in your main application component (**App**). Pass the array of objects to them as a prop.

What to Submit:
Please submit all your source code files, including your **package.json** file. If you have used any additional libraries or tools, please include a **README.md** file that explains how to install and run your project.

Rubric:
1. Correctness of code (Does the code fulfill the assignment requirements?): 60%
2. Code quality (Is the code readable and well-organized?): 20%
3. Design (Is the output well-designed and aesthetically pleasing?): 20%

Note:
Remember that functional components in React are also known as stateless components, as they do not maintain their own state. Instead, they receive all of their data via props.

Hints:
- To create a table in React, you can use the **<table>**, **<thead>**, **<tbody>**, **<tr>**, **<th>**, and **<td>** elements.
- To create a list in React, you can use the **<ul>** and **<li>** elements.
- To pass an array as a prop in React, you can use syntax like **<UserTable users={users} />** or **<UserList users={users} />**.

- To access a prop in a functional component, you can use syntax like **const UserTable = ({ users }) => { /* ... */ }**.
- Remember to add a **key** prop to each child in a list. This is a unique identifier that helps React maintain performance when updating the list.
- For styling, you can use either regular CSS files, inline styles, or a CSS-in-JS library like styled-components or emotion.

example of the type of data you could use:

```
export const users = [
  {
    id: 1,
    name: 'John Doe',
    email: 'john.doe@example.com',
    age: 34,
  },
  {
    id: 2,
    name: 'Jane Smith',
    email: 'jane.smith@example.com',
    age: 28,
  },
  {
    id: 3,
    name: 'Harry Johnson',
    email: 'harry.johnson@example.com',
    age: 46,
  },
  {
    id: 4,
    name: 'Emma Wilson',
    email: 'emma.wilson@example.com',
    age: 38,
  },
];
```

In this array, each object represents a user, and has four properties:
1. **id**: a unique identifier for each user (used for React's **key** prop in lists)
2. **name**: the user's name
3. **email**: the user's email
4. **age**: the user's age

You can put this array in a file (e.g., **data.js**) and import it in your main application file or wherever you need to use it.