



Day 2: Components, Props, List, Keys, Events, Conditional rendering

Problem: React-Problem-With-Module-Css

- Your task is to create a React application that displays information about mobile devices, courses, and users.

Folder Structure

```

-src
|--->App.jsx
|--->App.module.css
|--->Components
    +---courses
    |       Courses.jsx
    |       Courses.module.css
    +---mobileInfo
    |       MobileInfo.jsx
    |       MobileInfo.module.css
    +---users
        UserCard.jsx
        Users.jsx
        Users.module.css

```

App

This app contains different components like MobileInfo, Courses, Users

Mobile Operating System

- Android
- Blackberry
- iPhone
- Windows Phone

Mobile Manufacturers

- Samsung
- HTC
- Micromax
- Apple

Full Time Courses

1. Full Stack Web Development
 - Eligibility : 18-28 yrs
 - Duration : 30 weeks

Part Time Courses

1. Full Stack Web Development
 - Eligibility : 18-28 yrs
 - Duration : 30 weeks
2. Data Analytics
 - Eligibility : 18-28 yrs
 - Duration : 30 weeks

- This component serves as the main wrapper for the entire application.
- It should display a heading and import three child components: MobileInfo, Courses, and Users.
- The heading should be in the `h1` tag with the text content `This app contains different components like MobileInfo, Courses, Users`.
 - After importing display the components also in the same order as above, ie. 1. MobileInfo, 2. Courses, 3. Users

- The heading should be styled using CSS modules with a `blue` color.
- While writing in the `module.css` file use `className` as `heading` to give `blue` color to the `heading`.

MobileInfo



- This component should display two headings in `h1` tags, `Mobile Operating System` and `Mobile Manufacturers` followed by unordered lists containing respective data.
- The unordered lists under `Mobile Operating System` should contain the `li` with the following text contents.
 - Android, Blackberry, iPhone, Windows Phone
- The unordered lists under `Mobile Manufacturers` should contain the `li` with the following text contents.
 - Samsung, HTC, Micromax, Apple
- The order of `li` tags should be the same as the above order in which text Contents are provided.
- The component should be styled using `CSS modules` with `color` as `red` for the headings(`h1` tags) and a `border` as `red` around the container. Use `className` as `heading` for headings and `container` for `div` with `data-testid="mobile_info"` (As this is a container for this component) as `classNames` when you writing css in the `module.css` file.

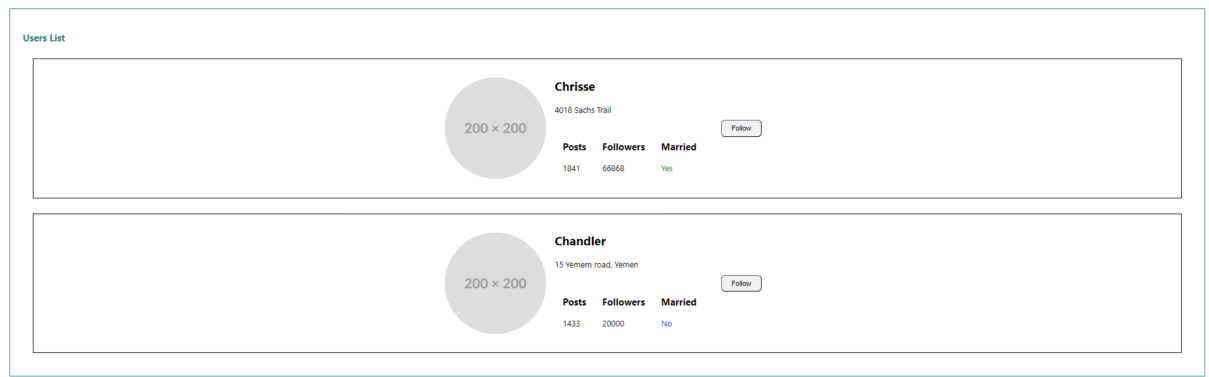
Courses



- This component should display information about full-time and part-time courses.
- For each type of course, display a heading(in h2 tag) and an ordered list with details about eligibility and duration.
 - There are two courses.
 - Full Time Courses and Part Time Courses in h2 tags.
 - Below h2 tag of Full Time Courses there should be ol tag.
 - The ol tag should contain a li tag with text content as Full Stack Web Development
 - Now the above li tag will also contain a child element ul tag.
 - the above ul tag should contain Eligibility and Duration of the course in li tags with the following text contents.
 - Eligibility : 18-28 yrs
 - Duration : 30 weeks
 - Below h2 tag of Part Time Courses there should be ol tag.
 - The ol tag should contain two li tags.
 - First li should contain the text Contain Full Stack Web Development
 - Now the above li tag will also contain a child element ul tag.
 - the above ul tag should contain Eligibility and Duration of the course in li tags with the following text contents.
 - Eligibility : 18-28 yrs
 - Duration : 30 weeks
 - Second li should contain text content Data Analytics
 - Now the above li tag will also contain a child element ul tag.

- the above `ul` tag should contain `Eligibility` and `Duration` of the course in `li` tags with the following text contents.
 - `Eligibility : 18-28 yrs`
 - `Duration : 30 weeks`
- Style the component using CSS modules with a `green` color for the headings(`h2` tags) and a `border` as `green` around the container. And use `heading` for heading and `wrapper` for `div` with `data-testid="mobile_info"` (this is the container for this component) as `classNames` when you write css in the module.css file.

Users

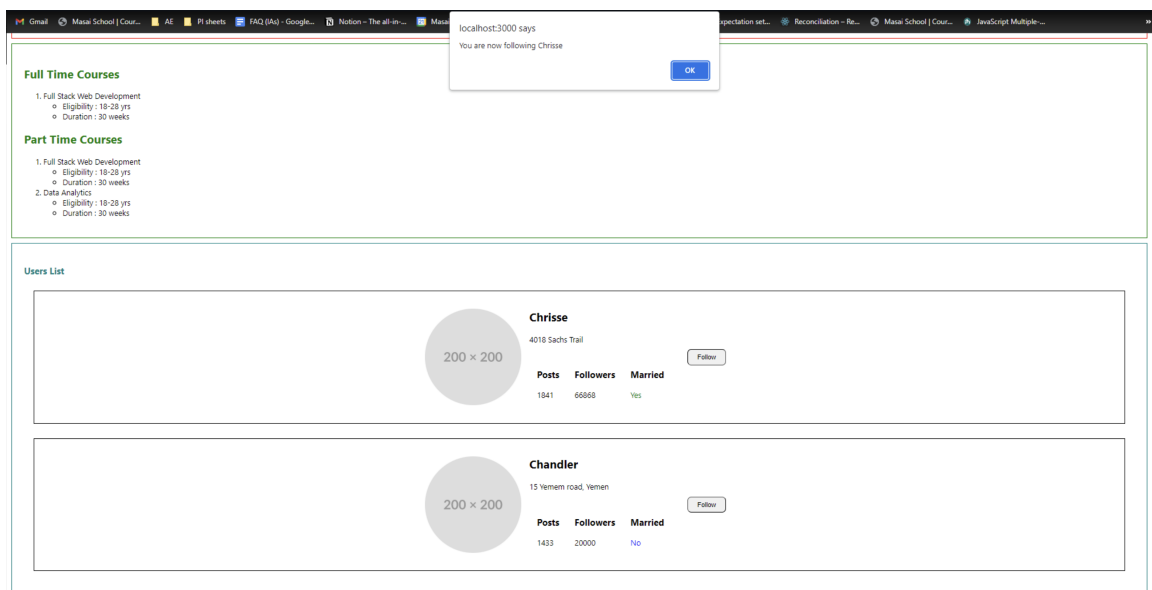


- This component will consist of a heading in `h3` tag and with text content as `Users List`
- This component should also display a list of `users` .(users list will be included in boilerplate)
- Each `user` should be represented by a `UserCard` component.
- The Users component should be styled using CSS modules with a `teal` color for the `heading` and a `teal border` around the container.
- use `heading` as a className for `heading` and `container` as a className for the `div` with `data-testid="users"` (this is the `container` of this component) while writing in module.css file.

UserCard

- This component should receive user data as props and display the details of the users with the help of the following HTML elements.

- avatar (in `img` tag)
- user's name (in `h2` tag with `data-testid="user_name"` as attribute)
- address (in `p` tag with `data-testid="user_address"` as attribute)
- There should be an `h3` with `Posts` as text content
- no of `posts` (in `p` tag with `data-testid="user_posts"` as an attribute)
- There should be an `h3` with `Followers` as text content
- no of `followers` (in `p` tag with `data-testid="user_followers"` as attribute)
- There should be an `h3` tag with the text content `Married`
- if the person is married there should be text content as `Yes` or else `No` in `p` tag with the attribute as `data-testid="is-married"`.
- there should be a `button` with text content as `Follow` on clicking on it we should get an alert `You are now following ${name}` where `name` is the name of the `user` you are following.



- Follow the requirements mentioned above to create the structure and styling of the components. Make sure to follow the best practices for using CSS modules, writing concise and maintainable code, and using React component composition.
- Ensure that your application is fully functional and responsive. Test your components thoroughly to validate their behavior and appearance.