**Rendering in React: Assignment**

**Instructions**

Your task is to build a React App for a profile page. It should contain a profile picture, name and birthday at the top. Below it should have a list of hobbies and a table of the daily schedule. Here is an example of what it might look like.

1:1

Use the starter code provided in the src folder. As you can see, the following component files are provided for you to add your solution:

* Header.js: This file contains the Header component. You should write code here so that Header renders the image of the kitten, its name, and birthday. It should also accept name, imageSrc, and birthday as props.
* HobbyList.js: This file contains the HobbyList component. You should write code here so that HobbyList renders the "Hobbies" heading and a list of the kitten's hobbies (e.g., "watching birds", "napping"). It should accept a hobbies array as a prop.
* Activity.js: This file contains the Activity component. You should write code here so that Activity renders the time and description for each activity (e.g., "8:00 am wake up"). It should accept time and description as props.
* ActivityList.js: This file contains the ActivityList component. You should write code here so that ActivityList renders a list of Activity components. It should accept an activities array as a prop.
* App.js: This file contains the App component, which is a wrapper for the other components. You will have to complete this component so that it renders the app properly.

The tests will be looking for the specific variable names and values defined in App.js.

**Specific Instructions**

* Do not assume anything about the list lengths. Your app should accommodate different values for the variables (name, birthday, imageSrc, hobbies, dailyActivities)
* If the hobbies list is empty, you should display nothing for hobbies.
* If the dailyActivities list is empty, you should display nothing for the schedule.
* Style the components using inline styling and/or CSS files. It does not need to match the appearance of the example image. Note that the automated tests **will NOT** be assessing the styling aspect of the assignment, so just try to achieve a reasonable layout.

**Success Criteria**

* Functionality:
  + Image is displayed
  + Name & birthday are displayed
  + Hobbies are displayed in a list labeled "Hobbies"
  + "Hobbies" does not appear when hobbies list is empty
  + Schedule is displayed in a list
* React Code Organization:
  + Uses the single responsibility components
  + Each component is in its own file
  + App component is just a wrapper for the other components
  + Data needed for the components is passed through props