# Classroom Activity: Demonstrate how to use the `useRef` hook to interact with the DOM directly, such as focusing an input field when the component mounts.

### InputFocusComponent.js:

This component will use `useRef` to focus the input field when the component mounts.

|  |
| --- |
| import React, { useRef, useEffect } from 'react';  const InputFocusComponent = () => {  const inputRef = useRef(null);   useEffect(() => {  // Focus the input field when the component mounts  inputRef.current.focus();  }, []);   return (  <div>  <h1>Focus on Input Field</h1>  <input ref={inputRef} type="text" placeholder="Focus me on mount" />  </div>  ); };  export default InputFocusComponent; |

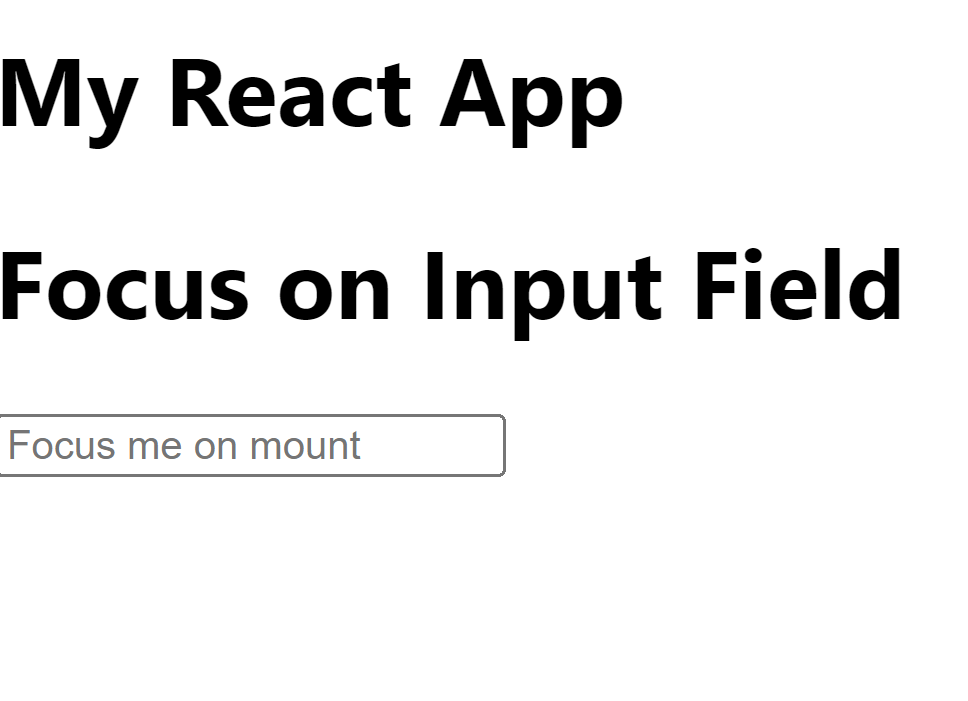
### App.js:

This file will import and use the `InputFocusComponent` to demonstrate the `useRef` functionality.

|  |
| --- |
| import React from 'react'; import InputFocusComponent from './InputFocusComponent';  function App() {  return (  <div className="App">  <header className="App-header">  <h1>My React App</h1>  <InputFocusComponent />  </header>  </div>  ); }  export default App; |

## Output:

**When you run the React app, the input field will be focused automatically when the component mounts.**

****

### Execution Steps:

1. Create a new React project if you don't have one already.
2. Create a file named InputFocusComponent.js in the src folder and write the code in the file.
3. Update App.js to import and use InputFocusComponent.
4. Save the files.
5. Run your React app using the terminal command: npm start.
6. Open your browser and go to http://localhost:3000. You should see the input field automatically focused when the page loads.