Report Day: 6.5

Submitted By: Ayesha Abid

Submitted To: Mr. Ali Hyder Hidayat

Group: Frontend Development @ProSensia

Dated: 1/08/25

Tasks:

Forms in React & Lifting State Up

1. Introduction
Forms allow us to collect data from users, and lifting state up helps manage and share that data between components.

2. Forms in React
What are Forms?
A form is a section in a webpage where users enter information. Examples include:
- Login form (email and password)
- Registration form
- Contact form
- Search input

In React, forms are not handled the same way as in plain HTML. Instead, React uses state to control the form fields, which means each input is connected to the component's memory. This is called a controlled component.

Example of a React Form:

```
import { useState } from 'react';

function MyForm() {
 const [name, setName] = useState('');

 const handleChange = (e) => {
  setName(e.target.value);
 };
```

```
  const handleSubmit = (e) => {
   e.preventDefault();
   alert("Hello, " + name);
  };

  return (
   <form onSubmit={handleSubmit}>
    <input type="text" value={name} onChange={handleChange} />
    <button type="submit">Submit</button>
   </form>
  );
}
```

Key Features:
- value={name} keeps input value in sync with state.
- onChange={handleChange} updates the state on every keystroke.
- onSubmit={handleSubmit} handles form submission.

Why Use Forms in React?

- Allows real-time validation (e.g., password strength).
- You can store and process the input anywhere in the app.
- Ensures input fields are always synced with app state.

3. Lifting State Up

Lifting state up means moving a shared piece of state from child components to their common parent. This is needed when:

- Two or more components need access to the same data.
- One component's action should affect another component.

Instead of keeping local state in multiple places, we lift the state up to the parent and pass it down using props.

React Example:

Parent Component (App.js):

```
function App() {
 const [name, setName] = useState('');

 return (
  <div>
    <NameInput name={name} setName={setName} />
    <Greeting name={name} />
  </div>
 );
}
```

Child Component 1:

```
function NameInput({ name, setName }) {
 return <input value={name} onChange={(e) => setName(e.target.value)} />;
}
```

Child Component 2:

```
function Greeting({ name }) {
 return <h2>Hello, {name}</h2>;
}
```

Benefits of Lifting State Up:

- Centralized state = consistent data.
- Reduces bugs and confusion.
- Makes components reusable and clean.

4. Conclusion

Both forms and lifting state up are vital in building interactive React applications. Forms allow us to take user input in a dynamic, controlled way, while lifting state up lets multiple components work with shared data efficiently.

Understanding these concepts not only improves your code structure but also teaches how to think in terms of component communication and data flow in React.