DEPARTMENT OF CSE
COURSE CODE: 23SDCS12A / 23SDCS12R
FULL STACK APPLICATION DEVELOPMENT

Date of the Session: __/__/____   **Time of The Session:_____to_____**

LAB – 3 ➔ Working with React props, state

Prerequisites:

Knowledge on the HTML elements and styling

Knowledge on the Java Script

Exercise

The TCS team lead need visited to KLU and they need User Profile Component as a real-time example, where we manage the user's personal information (like name, age, and location) in a parent component (App.js) and pass that data to a child component (UserProfile.js) using props. The user can also update this information, and we will manage the updates using state within the child component.finally TCS team need the following pages.

Pass user data (name, age, and location) to a child component as props.

Allow the user to update their profile, which will be handled via state in the child component. The updated profile information will be reflected back in the parent component using state lifting (passing the updated state back to the parent).

user-profile/

├── src/

  ├── App.js

  ├── UserProfile.js

  └── index.js

index.html

```html
<!doctype html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <link rel="icon" type="image/svg+xml" href="/vite.svg" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">

    <title>Vite + React</title>
  </head>
  <body>
    <div id="root"></div>
    <script type="module" src="/src/main.jsx"></script>
  </body>
</html>
```

❖ Watch The Video And Do In Visual Studio https://youtu.be/QiXjFroD4QI?si=r2w1FRcpQnzueCBn

Parent.jsx

```jsx
import { useState } from "react";
import Child from "./Child";

export default function Parent() {
  const [name, setName] = useState("n1");
  const [salary, setSalary] = useState(0);
  const [location, setLocation] = useState("11");

  const changeUser = (n, s, l) => {
    setName(n);
    setSalary(s);
    setLocation(l);
  };

  return (
```

```
      <div>
        <div>
          Name: {name}
          <br />
          Salary: <i className="fa fa-inr"></i> {salary}
          <br />
          Location: {location}
        </div>
        <br />
        <Child changeUser={changeUser} a={10} b={20}/>
      </div>
    );
}
```

Child.jsx

```
export default function Child(props){

  function changeFun() {
    var name = document.getElementById("idname").value;
    var salary = document.getElementsByName("tb")[0].value;
    var location = document.getElementsByName("tb")[1].value;
    props.changeUser(name, salary, location)
  }
  return(
    <div>
    Name: <input type="text" id="idname" />
    <br/>
    Salary: <input type="text" name="tb" />
    <br/>
    Location: <input type="text" name="tb" />
    <br/>
    <button  onClick={changeFun}> Update User </button>
    <br/>
    {props.a} - {props.b}
    </div>
  );
}
```

main.jsx

```
import { StrictMode } from 'react';
import { createRoot } from 'react-dom/client';
import './index.css';
import App from './App.jsx';
import Template from './components/ex1/Template.jsx';
import Parent from './components/lab 3/Parent.jsx';

createRoot(document.getElementById('root')).render(
  <StrictMode>
   <Parent />
  </StrictMode>,
);
```

Output

Name: THANOS
Salary: ₹ 10000
Location: VIJ

Name: THANOS
Salary: 10000
Location: VIJ
Update User
10 - 20

VIVA QUESTIONS:

1. What are props in React, and how are they different from state?

- **Props**: Read-only data passed from parent to child components.
- **State**: Mutable data managed within a component.

2. How would you pass props from a parent component to a child component? Provide an example.

Example:

```jsx
const Parent = () => <Child message="Hello!" />;
const Child = ({ message }) => <p>{message}</p>;
```

3. Explain how state is managed in React and how it differs from props.

- **State**: Local, mutable, managed with `setState` (class) or `useState` (functional).
- **Props**: Immutable, passed from parent to child.

4. What is the purpose of this.setState in class components?

To update the component's state and trigger a re-render.
Example:

```jsx
this.setState({ count: this.state.count + 1 });
```

5. How can you update state in functional components using hooks?

Use the `useState` hook.
Example:

```jsx
const [count, setCount] = useState(0);
setCount(count + 1);
```

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation
	Marks Secured _____ out of 50
	Full Name of the Evaluator:
	Signature of the Evaluator Date of Evaluation: