Don Bosco Institute of Technology
Kurla (W), Mumbai 400 070

Internet Programming

Name: Parth Pravin Shikhare

Class: TE-IT

Roll No.: 53

Date: 19/09/25

Experiment No.: 8

Title:

To Design a web page using React Props and State

Problem Definition:

To design a web page using React by implementing Props and State after installation and configuration of React JS, including JSX, Components, Props, and State concepts.

Pre-requisites:

1. Basic understanding of HTML, CSS, and JavaScript
2. Familiarity with ES6 features (let/const, arrow functions, import/export)
3. Node.js and npm installed
4. Basic knowledge of React project structure and JSX syntax

Theory:

React is a JavaScript library for building user interfaces using reusable components. It allows developers to create dynamic web applications efficiently. Key concepts involved in this experiment include:

- Installation & Configuration: React can be set up using Create React App (CRA) or Vite, enabling a ready development environment.
- JSX: JavaScript XML syntax allows HTML-like structures to be written inside JavaScript code.
- Components: The building blocks of React applications that can be functional or class-based.
- Props: Used to pass data from parent to child components for reusability.

- State: Represents mutable data within components. In functional components, it is managed using the useState hook.

Procedure:

1. Install Node.js and verify npm installation.
2. Create a new React project using Create React App or Vite.
3. Start the development server and open the project in a code editor.
4. Create functional or class-based components.
5. Pass data between components using Props.
6. Use the useState hook to manage and update State.
7. Render data dynamically using JSX expressions.
8. Use event handlers to update State and trigger re-renders.
9. Test the web page in a browser and verify the functionality.

Program:

UserCard.jsx

```
import "./UserCard.css";

function UserCard({ name, age, location }) {
 return (
  <div className="user-card">
   <div className="avatar">
    <span>{name.charAt(0)}</span>
   </div>
   <h2 className="user-name">{name}</h2>
   <p className="user-detail">
    <strong>Age:</strong> {age}
   </p>
   <p className="user-detail">
    <strong>Location:</strong> {location}
   </p>
  </div>
 );
}

export default UserCard;
```

UserCard.css

```
.user-card {
 background: linear-gradient(145deg, #ffffff, #f3f3f3);
 border-radius: 16px;
 width: 230px;
 padding: 25px 20px;
 box-shadow: 0 5px 15px rgba(0, 0, 0, 0.1);
 text-align: center;
```

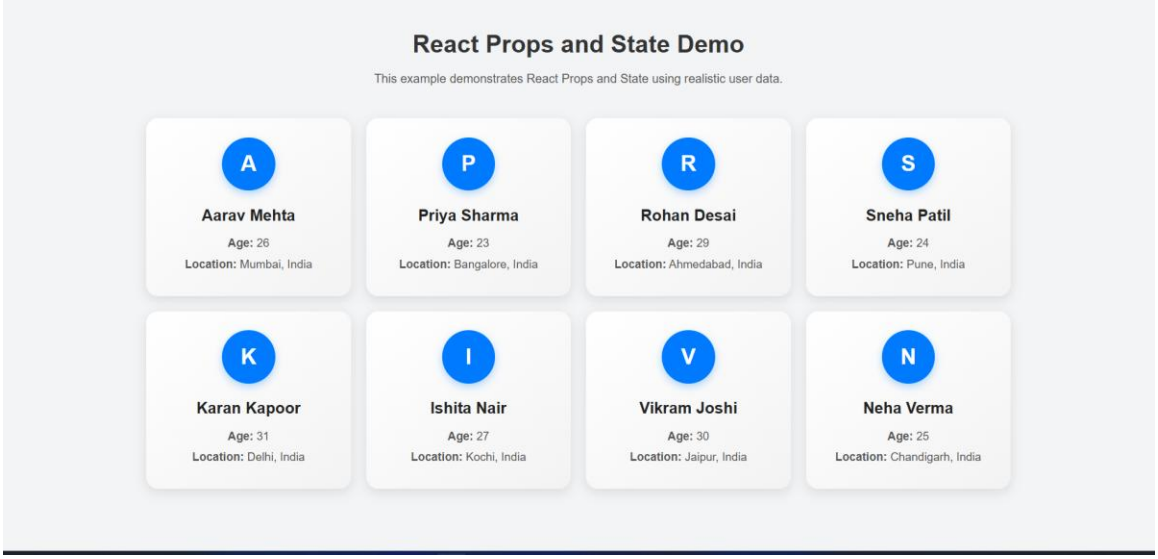
```
  transition: all 0.3s ease;
  cursor: pointer;
}

.user-card:hover {
  transform: translateY(-8px);
  box-shadow: 0 10px 25px rgba(0, 0, 0, 0.15);
  background: linear-gradient(145deg, #f0f4ff, #ffffff);
}

.avatar {
  background-color: #007bff;
  color: white;
  font-size: 1.8rem;
  font-weight: bold;
  width: 70px;
  height: 70px;
  margin: 0 auto 15px auto;
  border-radius: 50%;
  display: flex;
  align-items: center;
  justify-content: center;
  box-shadow: 0 4px 10px rgba(0, 123, 255, 0.3);
}

.user-name {
  font-size: 1.3rem;
  color: #222;
  margin-bottom: 10px;
  font-weight: 600;
}

.user-detail {
  color: #555;
  font-size: 0.95rem;
  margin: 5px 0;
}

@media (max-width: 600px) {
  .user-card {
    width: 85%;
  }
}
```

App.jsx

```
import { useState } from "react";
import UserCard from "./components/UserCard";
import "./App.css";
```

```
function App() {
  const [count, setCount] = useState(0);

  const users = [
    { name: "Aarav Mehta", age: 26, location: "Mumbai, India" },
    { name: "Priya Sharma", age: 23, location: "Bangalore, India" },
    { name: "Rohan Desai", age: 29, location: "Ahmedabad, India" },
    { name: "Sneha Patil", age: 24, location: "Pune, India" },
    { name: "Karan Kapoor", age: 31, location: "Delhi, India" },
    { name: "Ishita Nair", age: 27, location: "Kochi, India" },
    { name: "Vikram Joshi", age: 30, location: "Jaipur, India" },
    { name: "Neha Verma", age: 25, location: "Chandigarh, India" },
  ];

  return (
    <div className="app-container">
      <h1>React Props and State Demo</h1>
      <p className="description">
        This example demonstrates React Props and State using realistic user data.
      </p>

      <div className="cards-container">
        {users.map((user, index) => (
          <UserCard
            key={index}
            name={user.name}
            age={user.age}
            location={user.location}
          />
        ))}
      </div>
    </div>
  );
}

export default App;
```

Output:



Results:

Successfully designed a web page using React JS by implementing Props and State with proper component structuring.

References:

1. https://react.dev/
2. https://www.w3schools.com/react/
3. https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference