REACT JS

WEEK 7

## **Objectives**

* List the features of ES6
* Explain JavaScript let
* Identify the differences between var and let
* Explain JavaScript const
* Explain ES6 class fundamentals
* Explain ES6 class inheritance
* Define ES6 arrow functions
* Identify set(), map()

In this hands-on lab, you will learn how to:

* Use map() method of ES6
* Apply arrow functions of ES6
* Implement Destructuring features of ES6

## **Prerequisites**

The following is required to complete this hands-on lab:

* Node.js
* NPM
* Visual Studio Code

## **Notes**

Estimated time to complete this lab: **60 minutes.**

Create a React Application named “cricketapp” with the following components:

1. ListofPlayers

* Declare an array with 11 players and store details of their names and scores using the map feature of ES6



* Filter the players with scores below 70 using arrow functions of ES6.



1. IndianPlayers
   1. Display the Odd Team Player and Even Team players using the Destructuring features of ES6



* 1. Declare two arrays T20players and RanjiTrophy players and merge the two arrays and display them using the Merge feature of ES6



Display these two components in the same home page using a simple if else in the flag variable.

**Output:**

When Flag=true



When Flag=false



**Hint:**



Index.js

import React from 'react';import ReactDOM from 'react-dom/client';import './index.css';import App from './App';import reportWebVitals from './reportWebVitals';const root = ReactDOM.createRoot(document.getElementById('root'));root.render( <React.StrictMode> <App /> </React.StrictMode>);// If you want to start measuring performance in your app, pass a function// to log results (for example: reportWebVitals(console.log))// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitalsreportWebVitals();Index.css:

body { margin: 0; font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto', 'Oxygen', 'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue', sans-serif; -webkit-font-smoothing: antialiased; -moz-osx-font-smoothing: grayscale;}code { font-family: source-code-pro, Menlo, Monaco, Consolas, 'Courier New', monospace;}

App.js

import React, { useState } from "react";import "./App.css";import ListofPlayers from "./components/ListofPlayers";import Scorebelow70 from "./components/Scorebelow70";import OddPlayers from "./components/OddPlayers";import EvenPlayers from "./components/EvenPlayers";import ListofIndianPlayers from "./components/ListofIndianPlayers";// Player list with scoresconst players = [  { name: "David Warner", score: 50 }, // Australian  { name: "Steve Smith", score: 70 }, // Australian  { name: "Pat Cummins", score: 40 }, // Australian  { name: "Glenn Maxwell", score: 61 }, // Australian  { name: "Mitchell Starc", score: 61 }, // Australian  { name: "Sachin Tendulkar", score: 95 }, // Indian  { name: "MS Dhoni", score: 100 }, // Indian  { name: "Virat Kohli", score: 84 }, // Indian  { name: "Ravindra Jadeja", score: 64 }, // Indian  { name: "Suresh Raina", score: 75 }, // Indian  { name: "Rohit Sharma", score: 80 }, // Indian];// Actual Indian player names with numbered suffixconst IndianPlayers = [ "Sachin 1", "Dhoni 2", "Virat 3", "Rohit 4", "Yuvarajs 5", "Raina 6"];export default function App() { const [flag, setFlag] = useState(true); const handleToggle = () => { setFlag((prev) => !prev); }; return ( <div style={{ padding: "20px", fontFamily: "Arial, sans-serif" }}> <button onClick={handleToggle} className="toggle-button"> Toggle View </button> <p>Current flag value: <strong>{flag.toString()}</strong></p> {flag ? ( <> <h1>List of Players</h1> <ListofPlayers players={players} /> <hr /> <h1>List of Players having Scores Less than 70</h1> <Scorebelow70 players={players} /> </> ) : ( <> <h1>Indian Team</h1> <h2>Odd Players</h2> <OddPlayers players={IndianPlayers} /> <hr /> <h2>Even Players</h2> <EvenPlayers players={IndianPlayers} /> <hr /> <h2>List of Indian Players Merged:</h2> <ListofIndianPlayers IndianPlayers={IndianPlayers} /> </> )} </div> );}

Report Web Vitals:

const reportWebVitals = onPerfEntry => { if (onPerfEntry && onPerfEntry instanceof Function) { import('web-vitals').then(({ getCLS, getFID, getFCP, getLCP, getTTFB }) => { getCLS(onPerfEntry); getFID(onPerfEntry); getFCP(onPerfEntry); getLCP(onPerfEntry); getTTFB(onPerfEntry); }); }};export default reportWebVitals;

Package.json:

{ "name": "cricketapp", "version": "0.1.0", "private": true, "dependencies": { "@testing-library/dom": "^10.4.0", "@testing-library/jest-dom": "^6.6.3", "@testing-library/react": "^16.3.0", "@testing-library/user-event": "^13.5.0", "react": "^19.1.0", "react-dom": "^19.1.0", "react-scripts": "5.0.1", "web-vitals": "^2.1.4" }, "scripts": { "start": "react-scripts start", "build": "react-scripts build", "test": "react-scripts test", "eject": "react-scripts eject" }, "eslintConfig": { "extends": [ "react-app", "react-app/jest" ] }, "browserslist": { "production": [ ">0.2%", "not dead", "not op\_mini all" ], "development": [ "last 1 chrome version", "last 1 firefox version", "last 1 safari version" ] }}

Output:





