## **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:**



**PROGRAM:**

**App.js**

import React from 'react';

// Component to display all players

export const ListofPlayers = ({ players }) => {

  return (

    <ul>

      {players.map(({ name, score }, index) => (

        <li key={index}>

          {name} - {score}

        </li>

      ))}

    </ul>

  );

};

// Component to display players with score < 70

export const Scorebelow70 = ({ players }) => {

  const filtered = players.filter(player => player.score < 70);

  return (

    <ul>

      {filtered.map(({ name, score }, index) => (

        <li key={index}>

          {name} - {score}

        </li>

      ))}

    </ul>

  );

};

// Function to return odd-indexed Indian players

export const OddPlayers = (IndianTeam) => {

  return (

    <ul>

      {IndianTeam.filter((\_, index) => index % 2 !== 0).map((player, index) => (

        <li key={index}>{player}</li>

      ))}

    </ul>

  );

};

// Function to return even-indexed Indian players

export const EvenPlayers = (IndianTeam) => {

  return (

    <ul>

      {IndianTeam.filter((\_, index) => index % 2 === 0).map((player, index) => (

        <li key={index}>{player}</li>

      ))}

    </ul>

  );

};

// Component to display merged Indian players

export const ListofIndianPlayers = ({ IndianPlayers }) => {

  return (

    <ul>

      {IndianPlayers.map((player, index) => (

        <li key={index}>{player}</li>

      ))}

    </ul>

  );

};

// Main App component

const App = () => {

  const flag = true;

  const players = [

    { name: 'Player 1', score: 85 },

    { name: 'Player 2', score: 65 },

    { name: 'Player 3', score: 90 },

    { name: 'Player 4', score: 55 }

  ];

  const IndianTeam = ['Sachin', 'Dhoni', 'Kohli', 'Rohit', 'Yuvi', 'Raina'];

  const IndianPlayers = [...IndianTeam]; // Using spread operator to clone

  if (flag === true) {

    return (

      <div>

        <h1>List of Players</h1>

        <ListofPlayers players={players} />

        <hr />

        <h1>List of Players having Scores Less than 70</h1>

        <Scorebelow70 players={players} />

      </div>

    );

  } else {

    return (

      <div>

        <div>

          <h1>Indian Team</h1>

          <h1>Odd Players</h1>

          {OddPlayers(IndianTeam)}

          <hr />

          <h1>Even Players</h1>

          {EvenPlayers(IndianTeam)}

        </div>

        <hr />

        <div>

          <h1>List of Indian Players Merged:</h1>

          <ListofIndianPlayers IndianPlayers={IndianPlayers} />

        </div>

      </div>

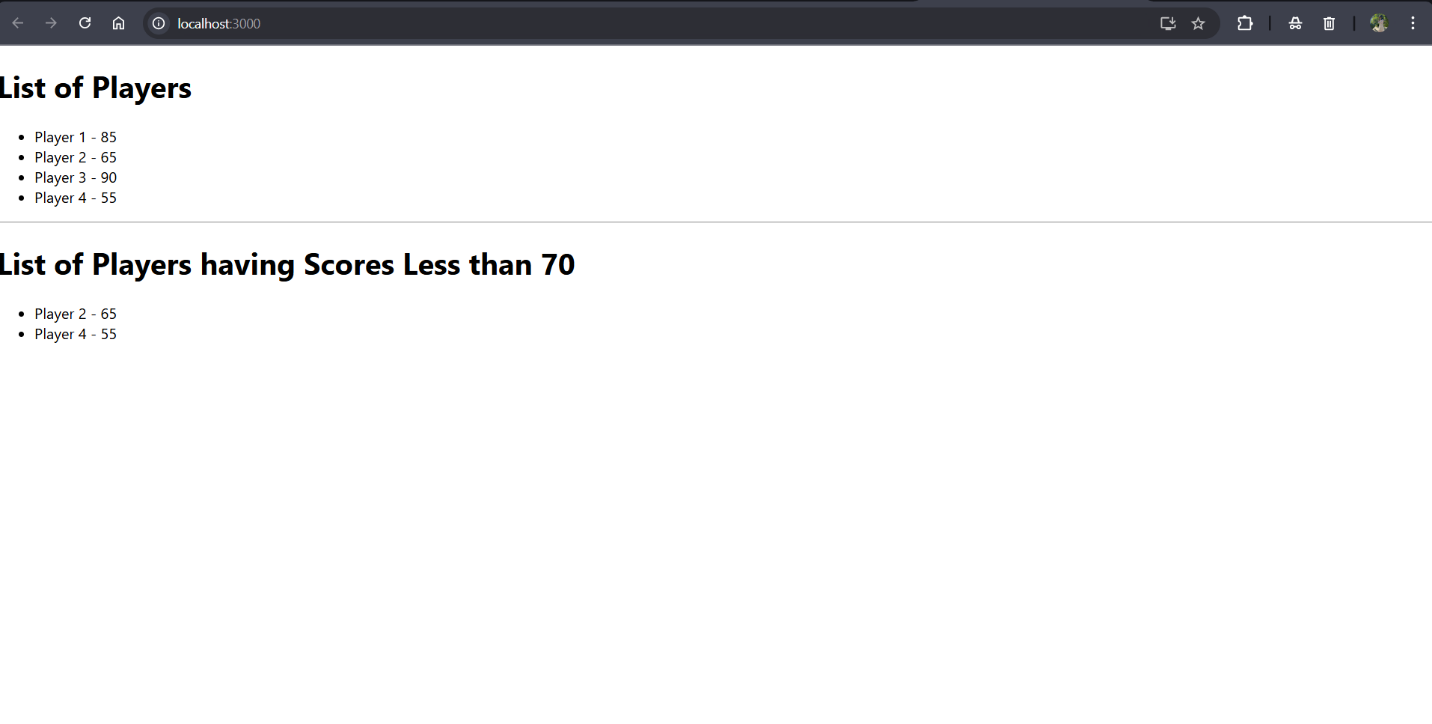
    );

  }

};

export default App;

**OUTPUT:**

****