**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:**



Solution:

My ListofPlayers.js code is:

import React from 'react';

function ListofPlayers({ players }) {

  return (

    <div>

      <h1>List of Players</h1>

      <ul>

        {players.map((item, index) => (

          <li key={index}>Mr. {item.name} <span>{item.score}</span></li>

        ))}

      </ul>

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

      <ul>

        {players.filter((item) => item.score < 70).map((item, index) => (

          <li key={index}>Mr. {item.name} <span>{item.score}</span></li>

        ))}

      </ul>

    </div>

  );

}

export default ListofPlayers;

**my IndianPlayers.js code is:**

import React from 'react';

export function OddPlayers({ players }) {

  const [first,, third,, fifth] = players;

  return (

    <div>

      <h1>Odd Players</h1>

      <ul>

        <li>First : {first}</li>

        <li>Third : {third}</li>

        <li>Fifth : {fifth}</li>

      </ul>

    </div>

  );

}

export function EvenPlayers({ players }) {

  const [, second,, fourth,, sixth] = players;

  return (

    <div>

      <h1>Even Players</h1>

      <ul>

        <li>Second : {second}</li>

        <li>Fourth : {fourth}</li>

        <li>Sixth : {sixth}</li>

      </ul>

    </div>

  );

}

const T20Players = ['Sachin1', 'Dhoni2', 'Virat3'];

const RanjiTrophyPlayers = ['Rohit4', 'YuvaraJ5', 'Raina6'];

export const IndianPlayers = [...T20Players, ...RanjiTrophyPlayers];

export default { OddPlayers, EvenPlayers, IndianPlayers };

**My app.js code is:**

import React from 'react';

import ListofPlayers from './components/ListofPlayers';

import { OddPlayers, EvenPlayers, IndianPlayers } from './components/IndianPlayers';

function App() {

  const flag = true; // change true/false to switch views

  const players = [

    { name: 'Jack', score: 50 },

    { name: 'Michael', score: 70 },

    { name: 'John', score: 40 },

    { name: 'Ann', score: 61 },

    { name: 'Elisabeth', score: 61 },

    { name: 'Sachin', score: 95 },

    { name: 'Dhoni', score: 100 },

    { name: 'Virat', score: 84 },

    { name: 'Jadeja', score: 64 },

    { name: 'Raina', score: 75 },

    { name: 'Rohit', score: 80 }

  ]

  if (flag === true) {

    return (

      <div>

        <ListofPlayers players={players} />

      </div>

    );

  } else {

    return (

      <div>

        <div>

          <h1>Indian Team</h1>

          <OddPlayers players={IndianPlayers} />

          <hr />

          <EvenPlayers players={IndianPlayers} />

        </div>

        <hr />

        <div>

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

          <ul>

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

              <li key={index}>{player}</li>

            ))}

          </ul>

        </div>

      </div>

    );

  }

}

export default App;

output:



