Aman Raj

Superset ID: 6358186

1. **ReactJS**

**Features of ES6 :-**

* let and const keywords (block-scoped variables)
* Arrow functions: () => {}
* Template literals: `Hello ${name}`
* Default parameters in functions
* Destructuring assignment
* Rest ...args and Spread ...array operators
* Enhanced object literals
* Classes and Inheritance
* Promises
* Modules (import, export)
* Map, Set, WeakMap, WeakSet
* For...of loop
* Symbol data type

**JavaScript let**

* let allows you to declare block-scoped variables.
* Unlike var, it doesn't hoist to the top of the function scope.
* You can reassign values but cannot redeclare within the same scope.

**JavaScript const**

* const is block-scoped, like let.
* Must be initialized at the time of declaration.
* The reference cannot be changed, but the content (for objects/arrays) can be modified.

**ES6 Class Fundamentals**

* Syntactic sugar over JavaScript prototype-based inheritance.
* Defines constructors and methods.

**ES6 Arrow Functions**

* Shorter syntax: () => {}
* No own this, arguments, or super
* Great for callbacks and array methods

**Create a React Application named “cricketapp”**

**App.js**

import React from 'react';

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';

const App = () => {

const flag = true;

const players = [

{ name: "Virat", score: 85 },

{ name: "Kohli", score: 65 },

{ name: "Rohit", score: 40 },

{ name: "Dhoni", score: 90 },

{ name: "Gill", score: 60 },

{ name: "Pant", score: 55 },

];

const IndianTeam = ["Virat", "Rohit", "Dhoni", "Pant", "Kohli", "Jadeja"];

const T20players = ["Surya", "Hardik"];

const RanjiPlayers = ["Pujara", "Ishant"];

const IndianPlayers = [...T20players, ...RanjiPlayers];

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>

<h1>Indian Team</h1>

<h2>Odd Players</h2>

<OddPlayers team={IndianTeam} />

<hr />

<h2>Even Players</h2>

<EvenPlayers team={IndianTeam} />

<hr />

<h2>List of Indian Players Merged</h2>

<ListofIndianPlayers players={IndianPlayers} />

</div>

);

}

};

export default App;

**ListofPlayers.js**

const ListofPlayers = ({ players }) => {

return (

<ul>

{players.map((p, i) => (

<li key={i}>{p.name} - {p.score}</li>

))}

</ul>

);

};

export default ListofPlayers;

**Scorebelow70.js**

import React from 'react';

const Scorebelow70 = ({ players }) => {

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

return (

<ul>

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

<li key={index}>

{player.name} - {player.score}

</li>

))}

</ul>

);

};

export default Scorebelow70;

**OddPlayers.js**

import React from 'react';

const OddPlayers = ({ team }) => {

const oddPlayers = team.filter((\_, index) => index % 2 !== 0);

return (

<ul>

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

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

))}

</ul>

);

};

export default OddPlayers;

**EvenPlayers.js**

import React from 'react';

const EvenPlayers = ({ team }) => {

const evenPlayers = team.filter((\_, index) => index % 2 === 0);

return (

<ul>

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

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

))}

</ul>

);

};

export default EvenPlayers;

**ListofIndianPlayers.js**

import React from 'react';

const ListofIndianPlayers = ({ players }) => {

return (

<ul>

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

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

))}

</ul>

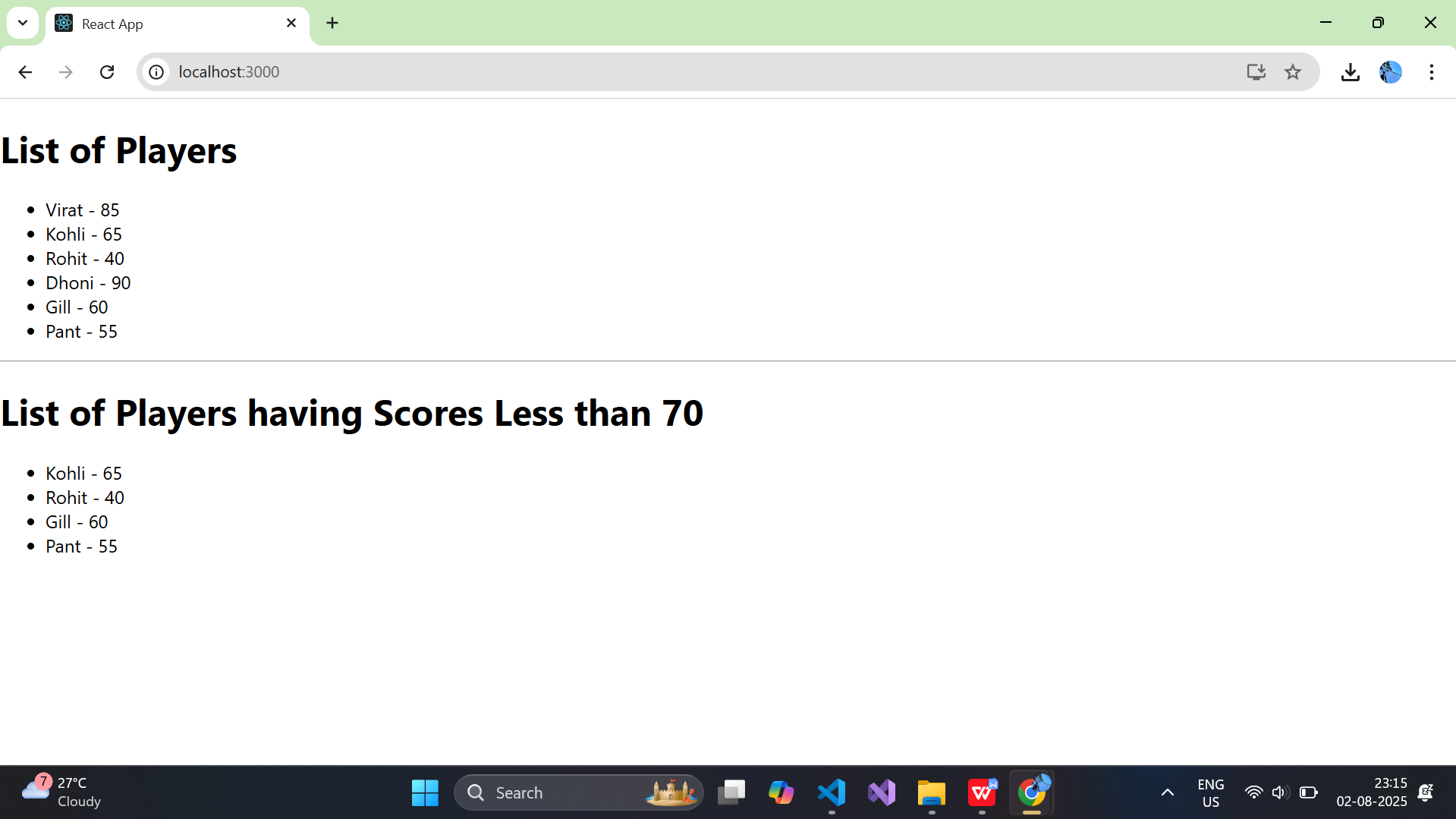
);

};

export default ListofIndianPlayers;

**Outputs:-**

**When Flag=true**



**When Flag=false**

