**Day3(29-Aug) React.js**

**================================================================**

**Session Agenda:**

Introduction to react

Create-react-app

Project structure

Understand components

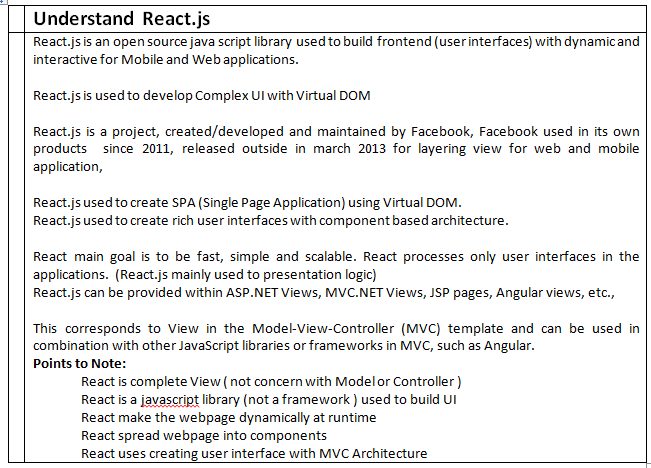
Functional components

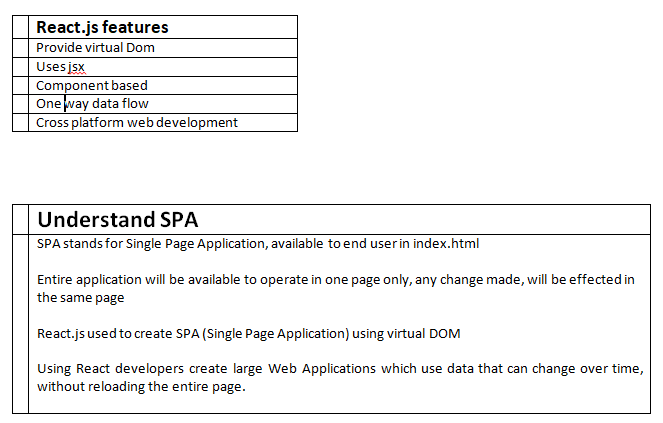
Class components

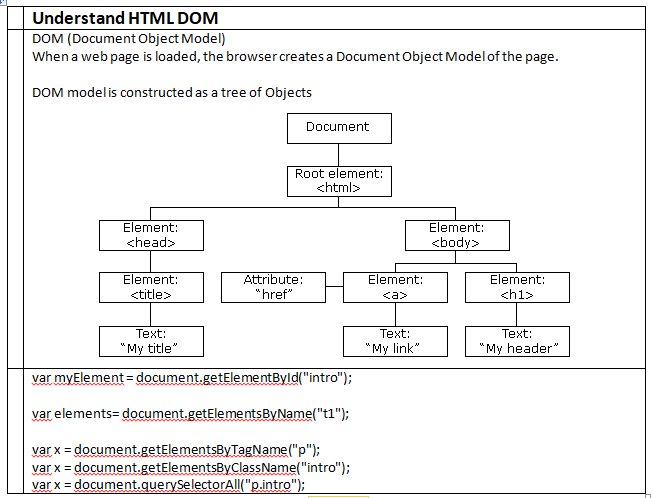
Pass data to components

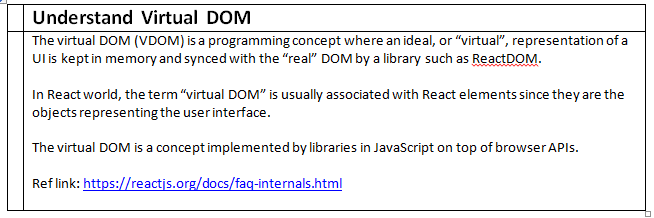
HTML5 Concepts

CSS concepts

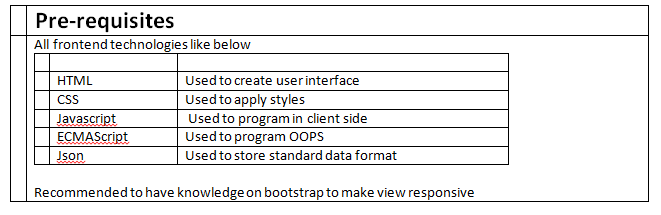


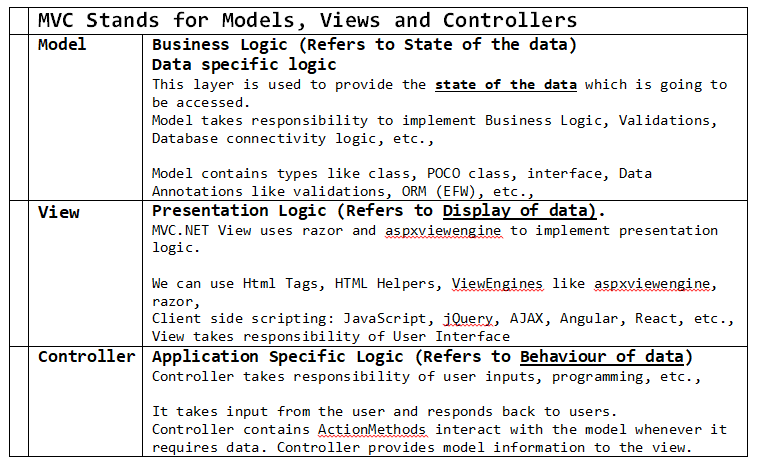


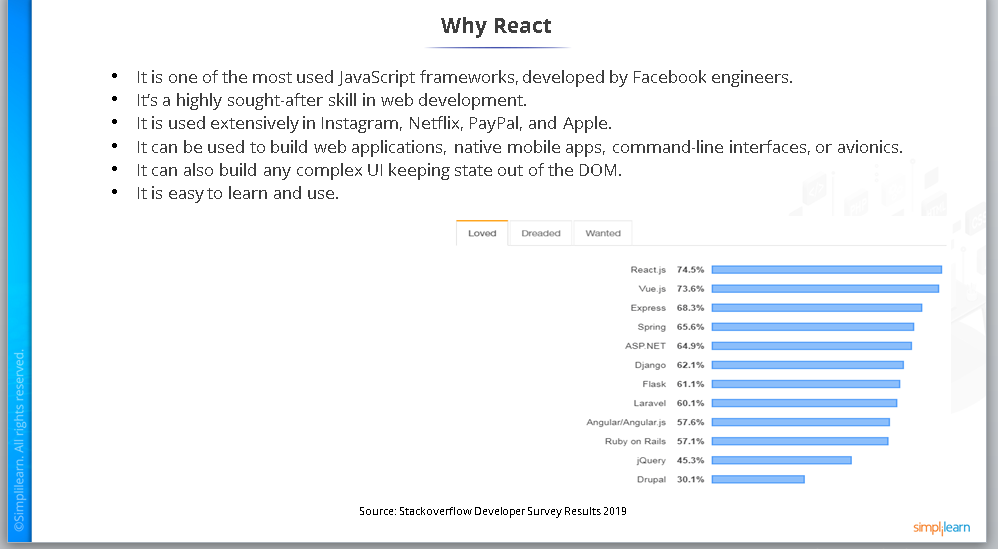


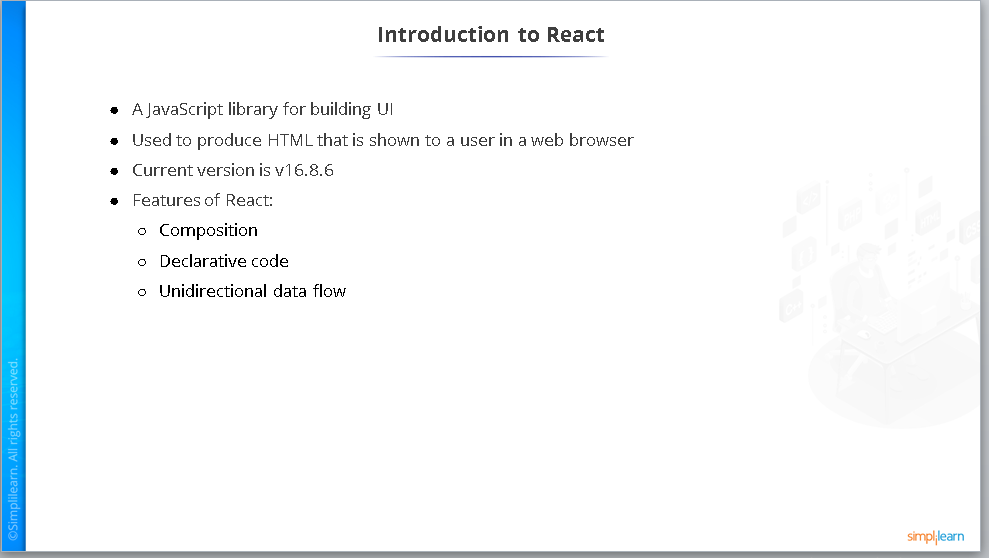


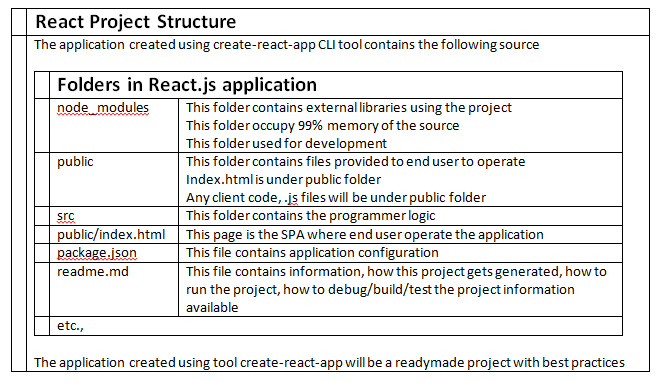
|  |  |  |
| --- | --- | --- |
|  |  |  |
|  | HTML | UI Designing |
|  | CSS | Style |
|  | Node.js | IT is a javascript runtime used to create Web Application and Web Services |
|  | React.js | It is a open source javascript library used to create Frontend Web Application, Complex UI using Virtual DOM |
|  | MongoDB | It is NoSql, used to store unstructured data permanently |







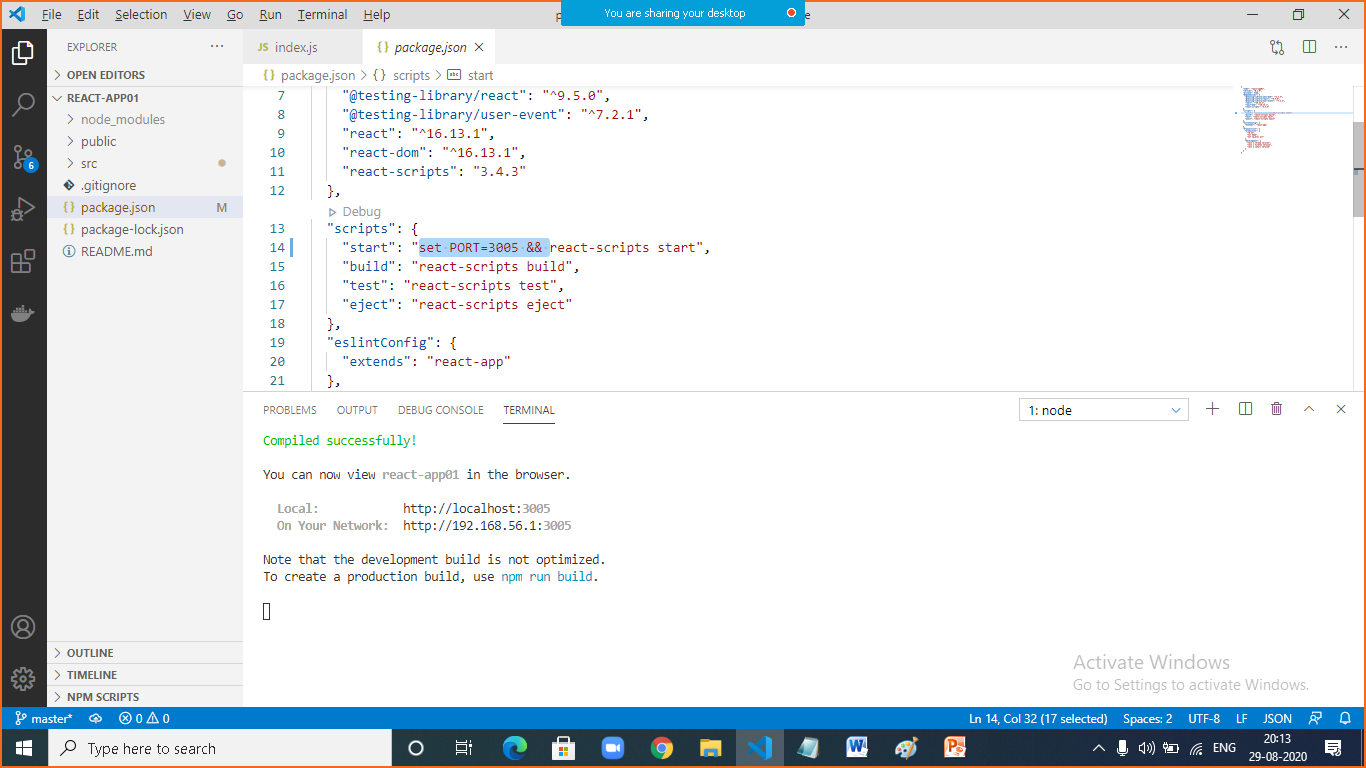




The default port is 3000

More applications are running, the next available port will be allotted

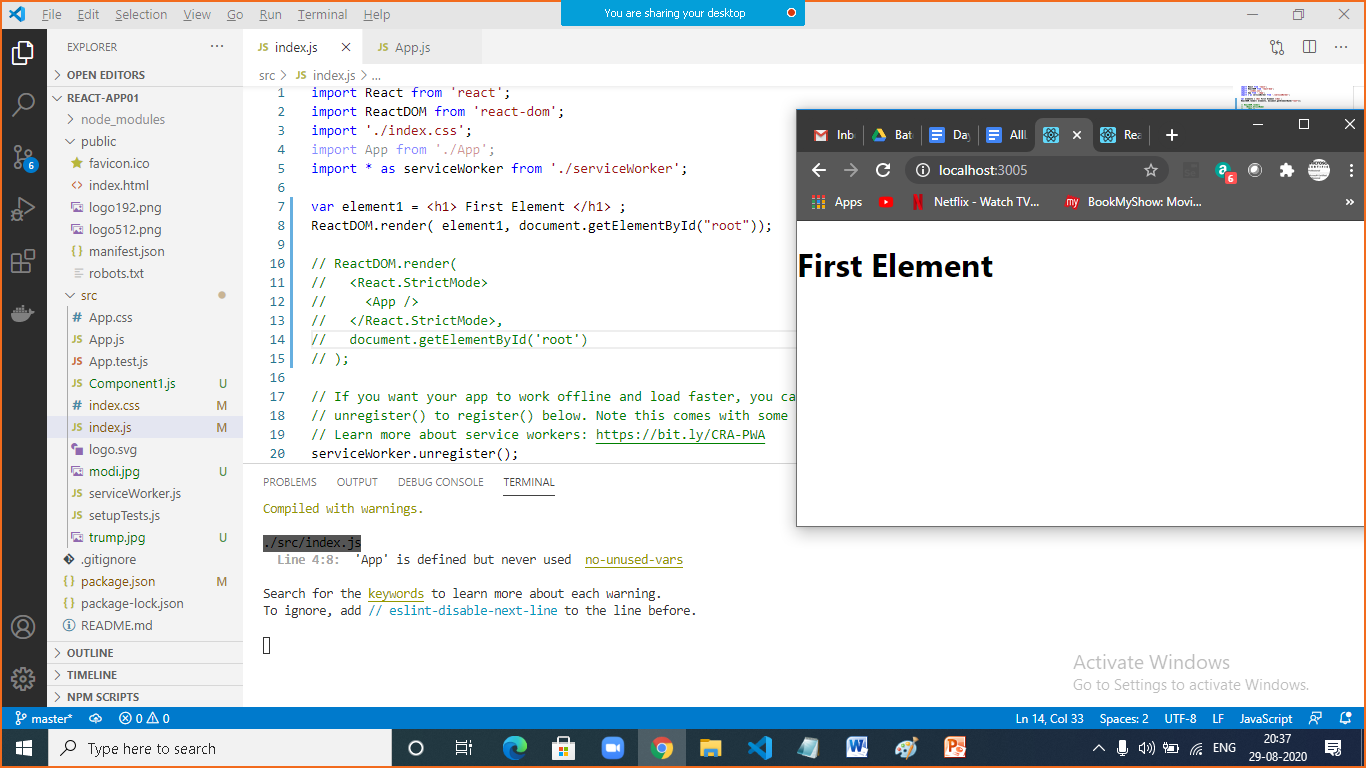
To change port number for react application

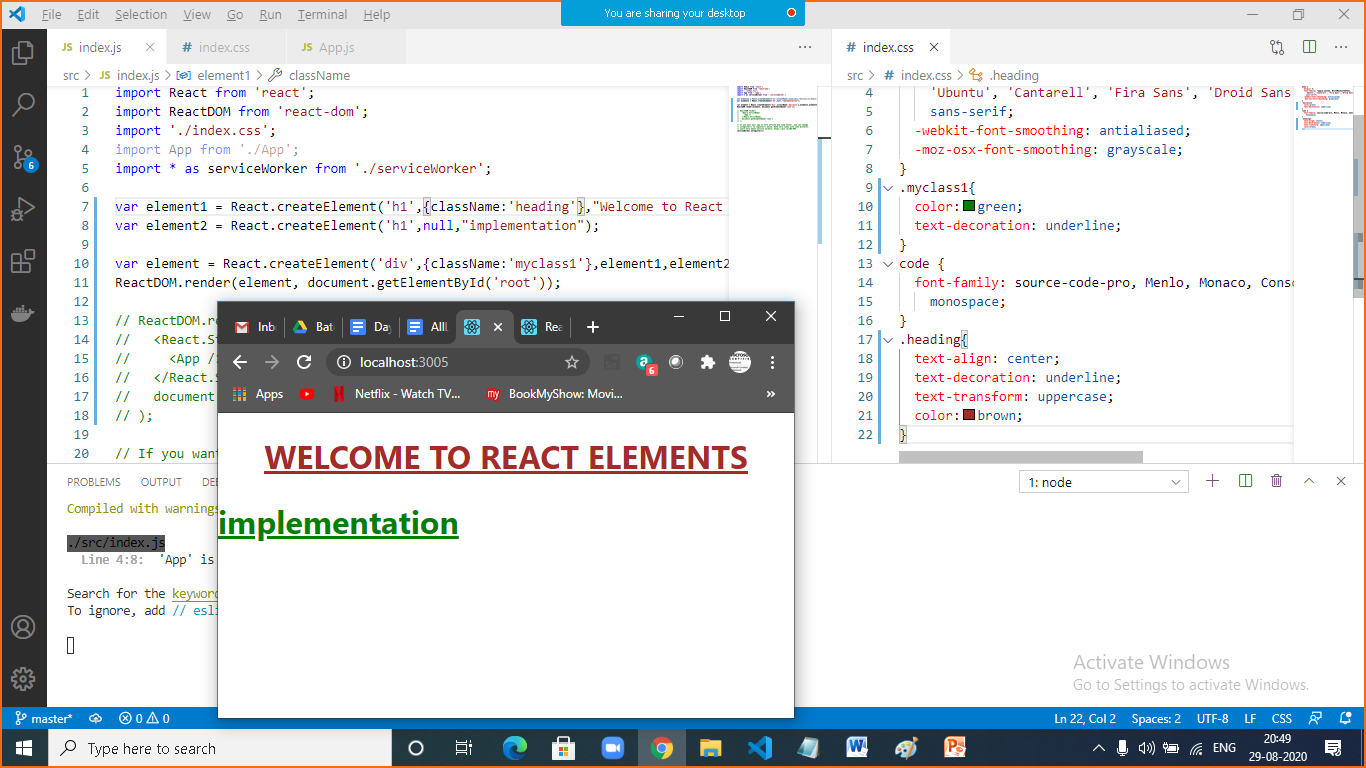


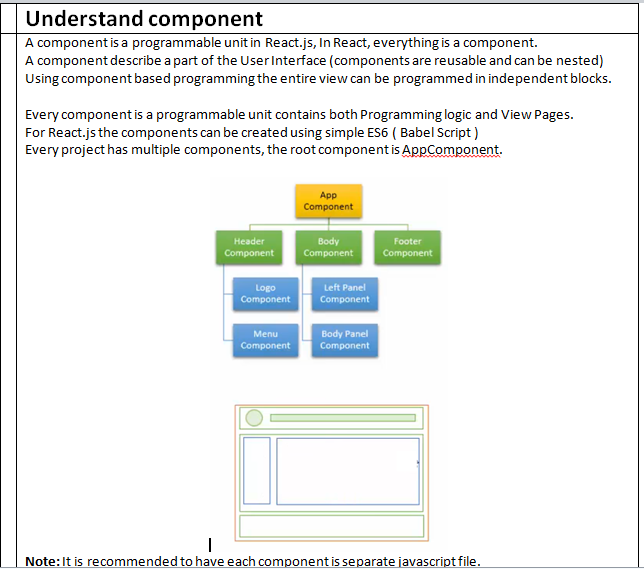
|  |  |
| --- | --- |
|  | **package.json file** |
|  | This file contains project configuration followed to depend libraries information with their version |
|  | Package-lock.json file  This file contains information about dependent libraries with their reference link  Using package-lock.json file, the dependent libraries available with their exact version |

|  |  |
| --- | --- |
|  | **Understand service worker** |
|  | The service worker is a web API that helps you cache your assets and other files so that when the user is offline or on slow network, he/she can still see results on the screen, as such, it helps you build a better user experience    **To use service worker:**  import \* as serviceWorker from './serviceWorker';      serviceWorker.unregister();  serviceWorker.register(); |

|  |  |
| --- | --- |
|  | **Understand StrictMode tool** |
|  | StrictMode is a tool, used as element in react, this tool is used for highlighting potential problems in an application. Like Fragment, StrictMode does not render any visible UI. It activates additional checks and warnings for its descendants.    **Example:**  Identifying components with unsafe lifecycles  Detecting unexpected side effects  Warning about deprecated findDOMNode usage  Detecting legacy context API      ReactDOM.render(  <React.StrictMode>  <App />  </React.StrictMode>,  document.getElementById('root')  ); |

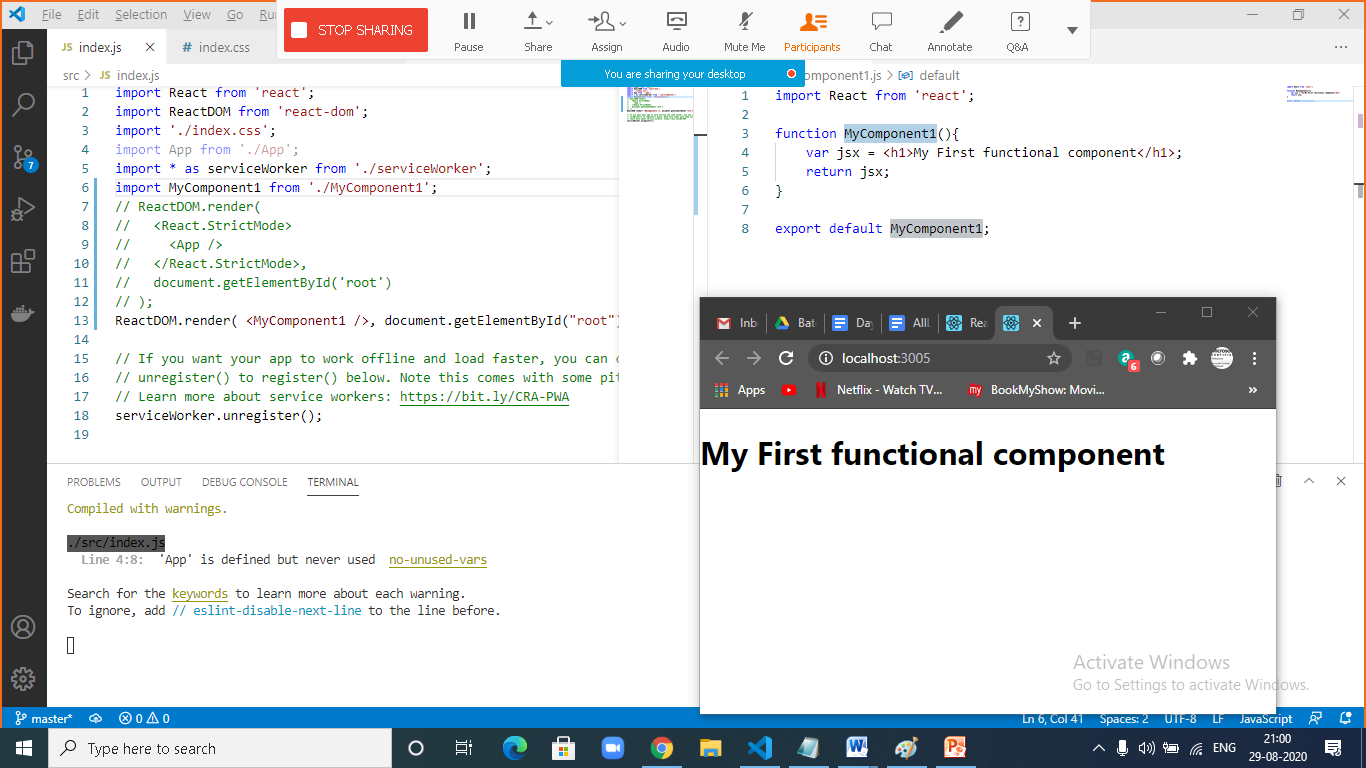






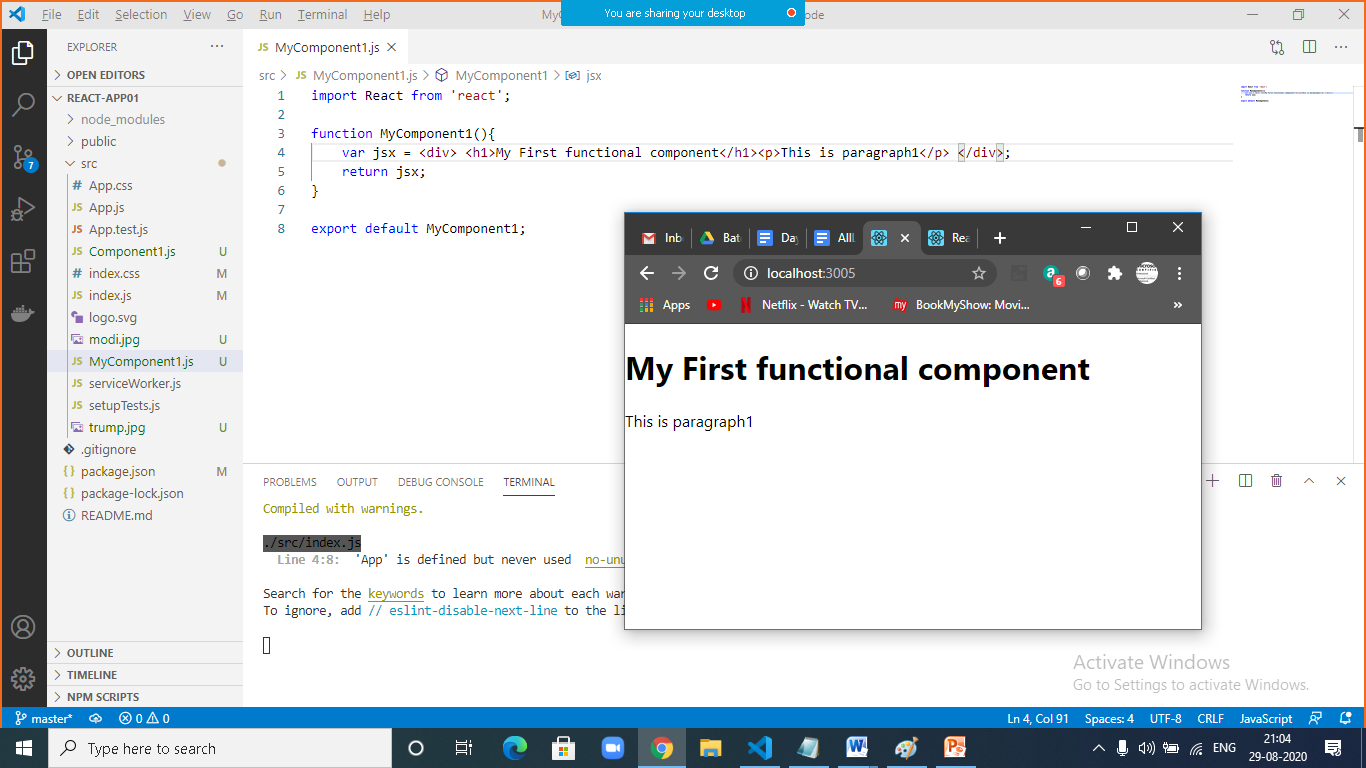
|  |  |
| --- | --- |
|  | **Understand component** |
|  | A components is a javascript file, which contains both application logic and presentation logic  The default component is App.js file  The default component injected in division root within index.html page  How to create component?  React allow to create component using 2 ways within .js file  Functional components  Class components |

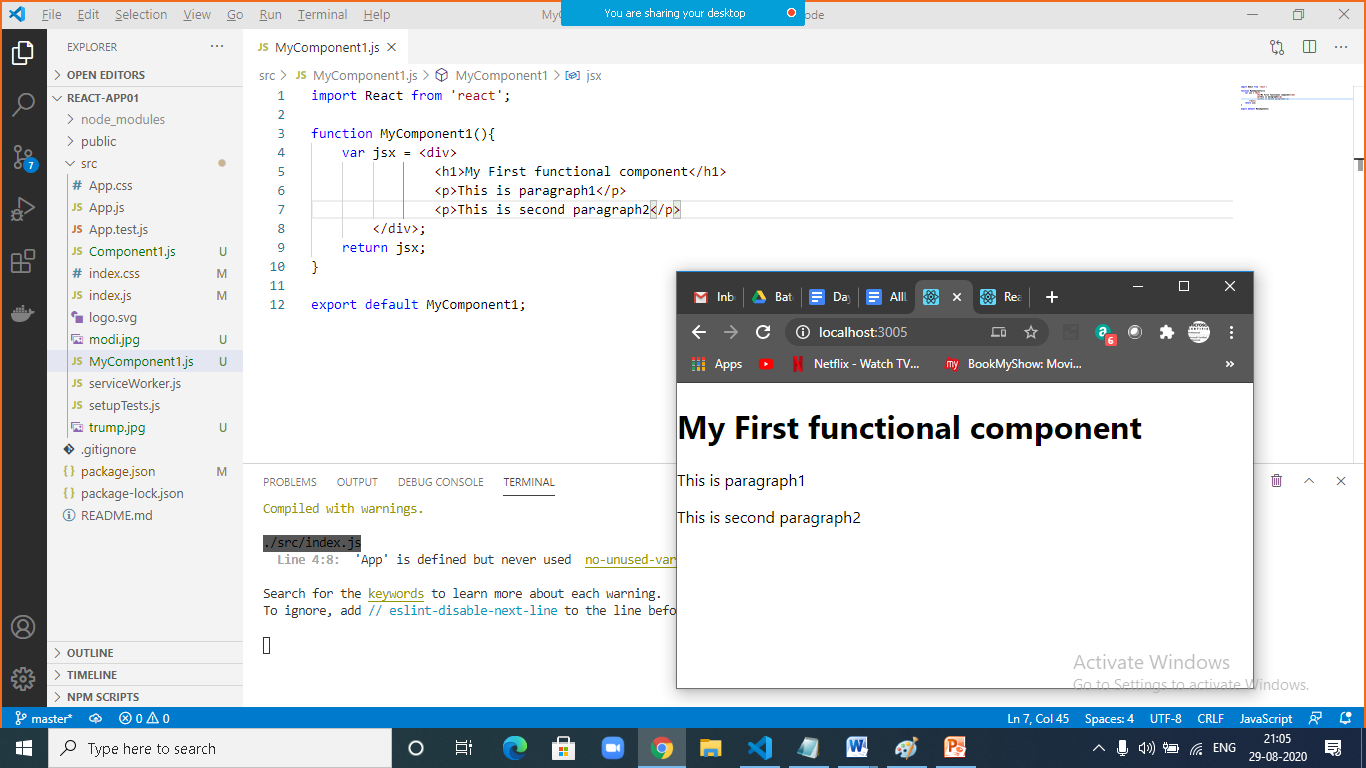
|  |  |
| --- | --- |
|  | **Functional components** |
|  | These are used to define components using java script functions    To import React:  import React from 'react';    function ComponentName(){  return jsx;  }    To export our component  export default ComponentName |



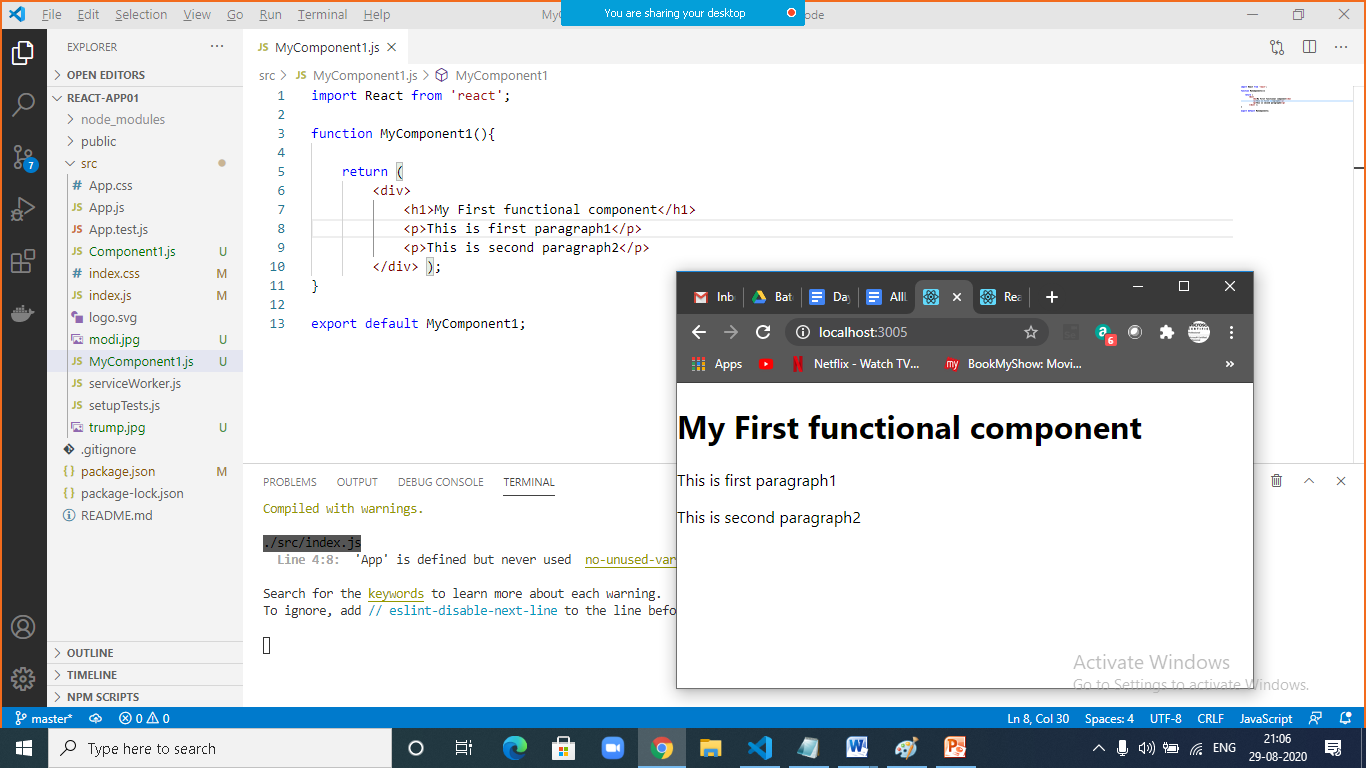
How to provide multiple elements in jsx?

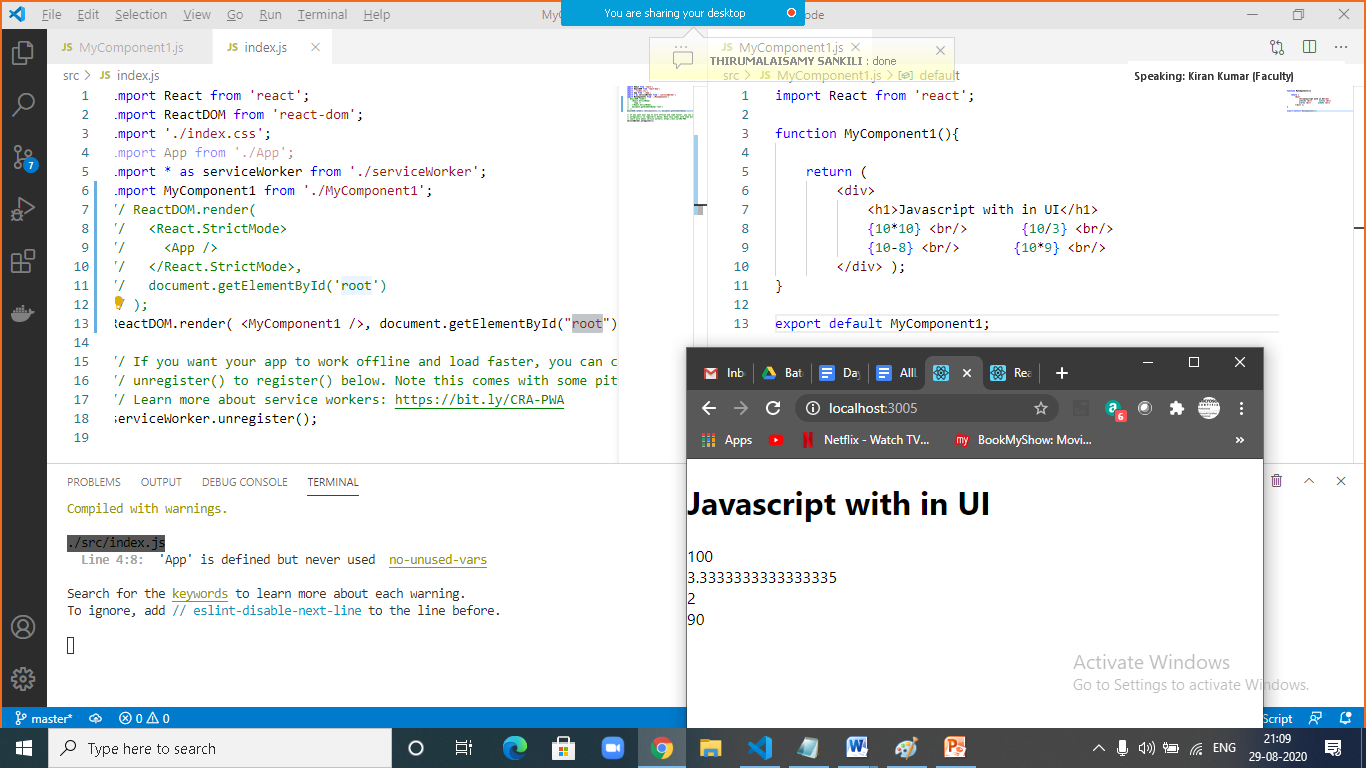
By providing <div></div>

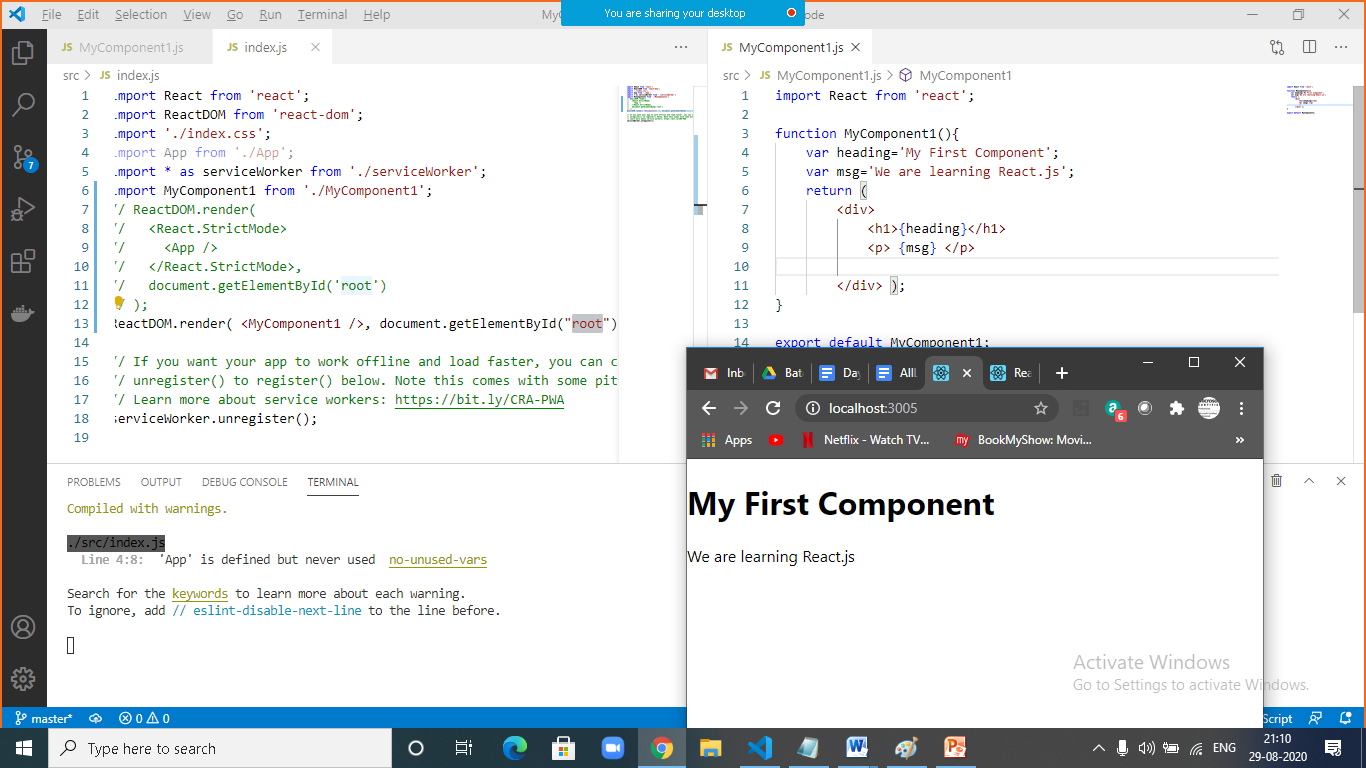


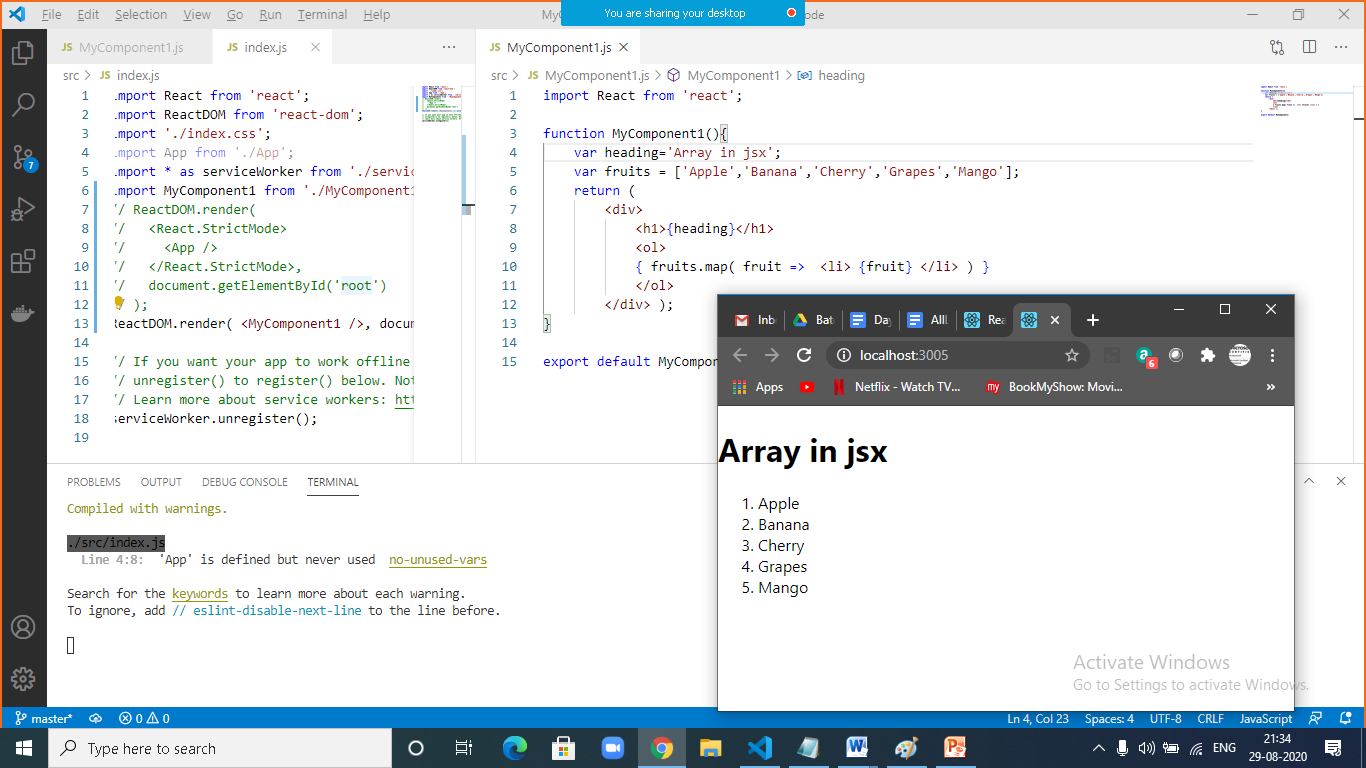


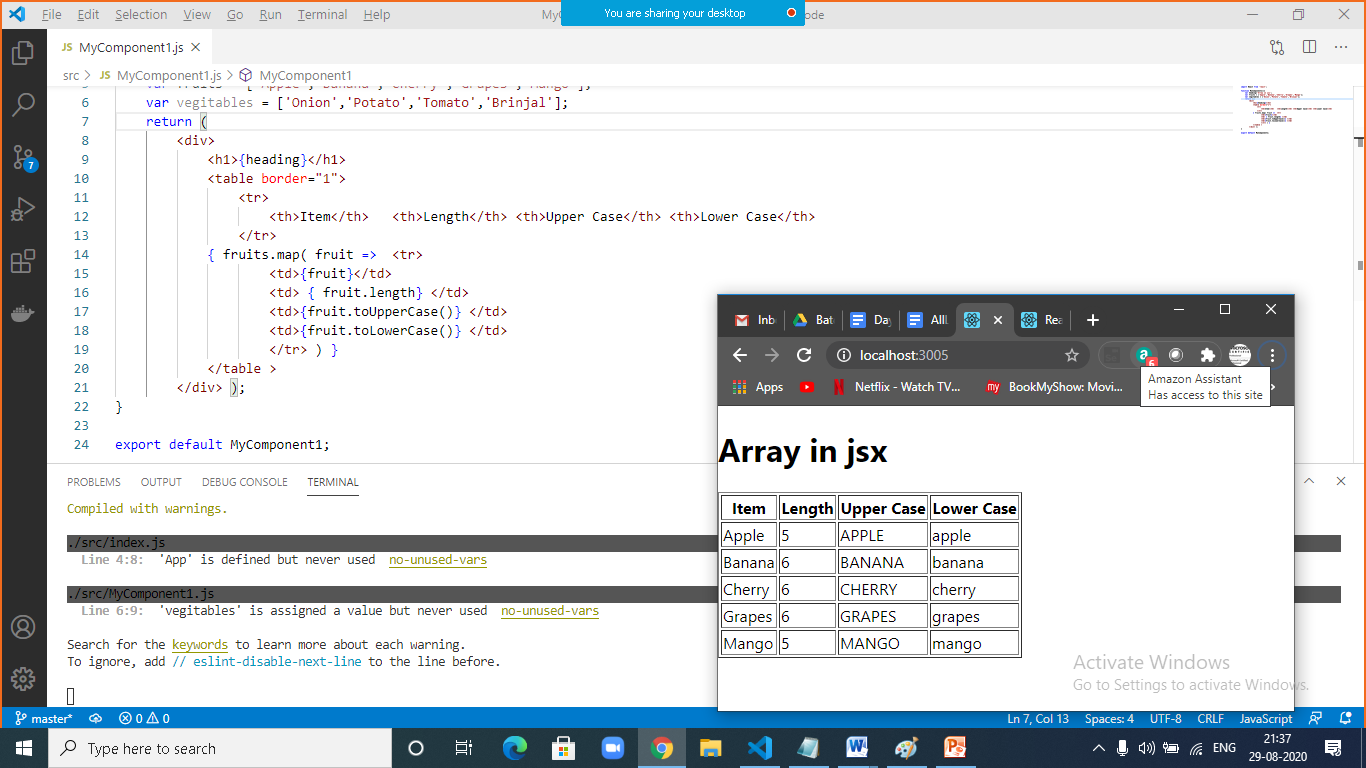
Return jsx in functional component

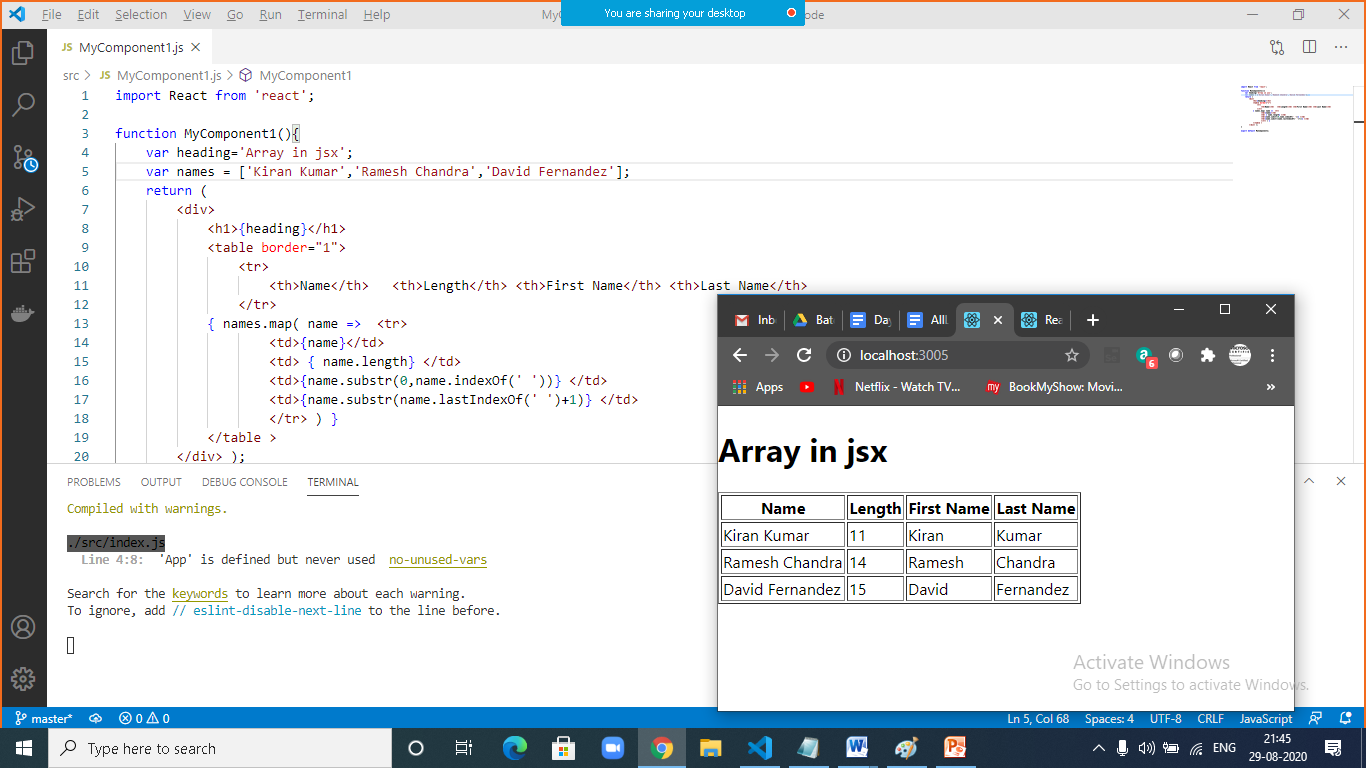








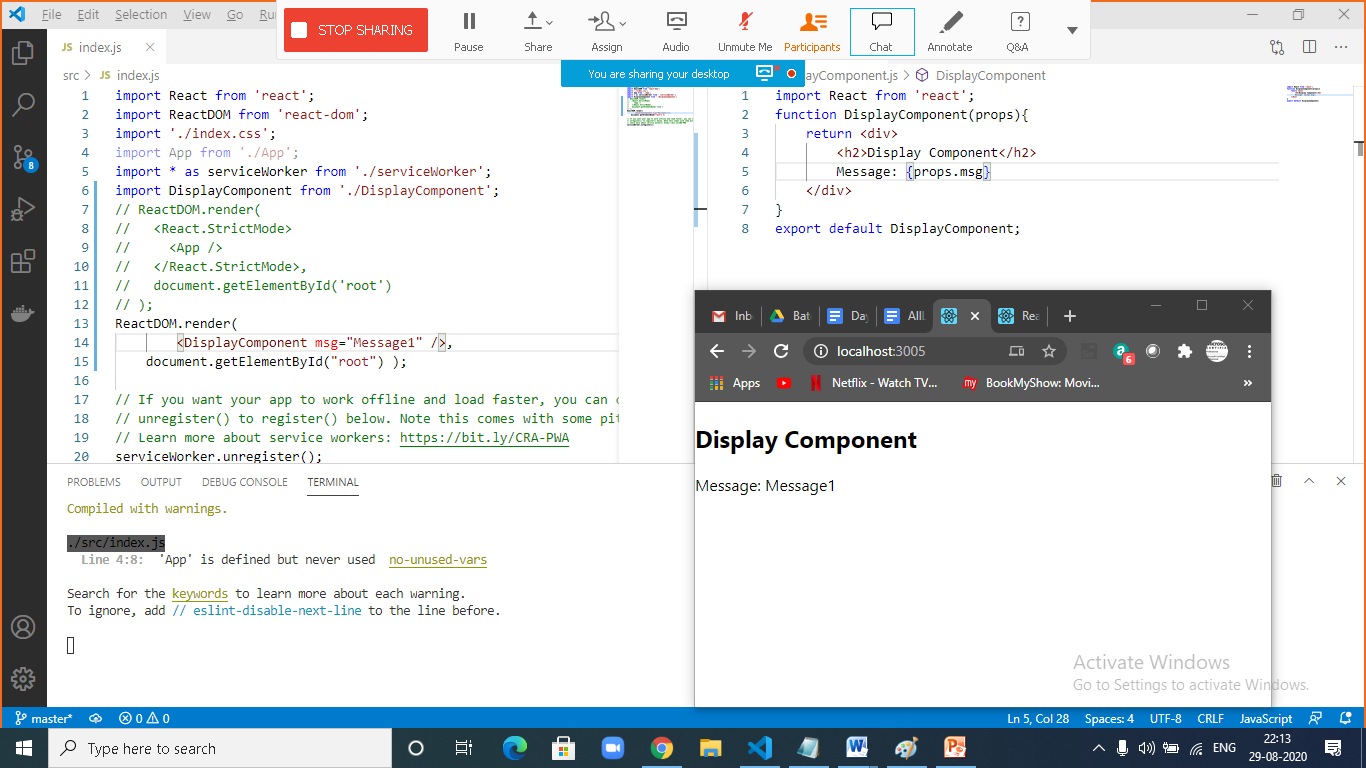




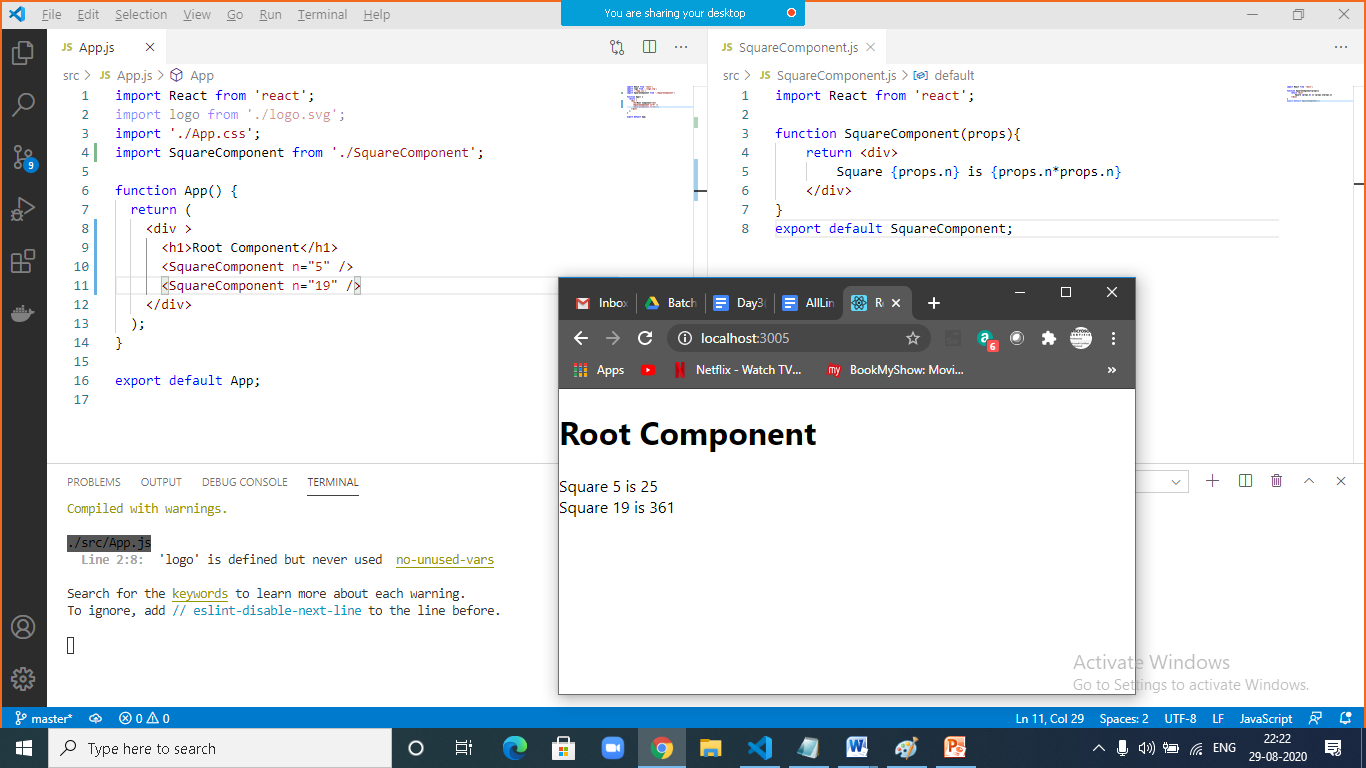
|  |  |
| --- | --- |
|  |  |
|  | import React from 'react';  function MyComponent1(){  var heading='Array in jsx';  var names = ['Kiran Kumar','Ramesh Chandra','David Fernandez'];  return (  <div>  <h1>{heading}</h1>  <table border="1">  <tr>  <th>Name</th> <th>Length</th> <th>First Name</th> <th>Last Name</th>  </tr>  { names.map( name => <tr>  <td>{name}</td>  <td> { name.length} </td>  <td>{name.substr(0,name.indexOf(' '))} </td>  <td>{name.substr(name.lastIndexOf(' ')+1)} </td>  </tr> ) }  </table >  </div> );  }  export default MyComponent1; |

|  |  |
| --- | --- |
|  | Task: Display employee records in jsx |
|  |  |

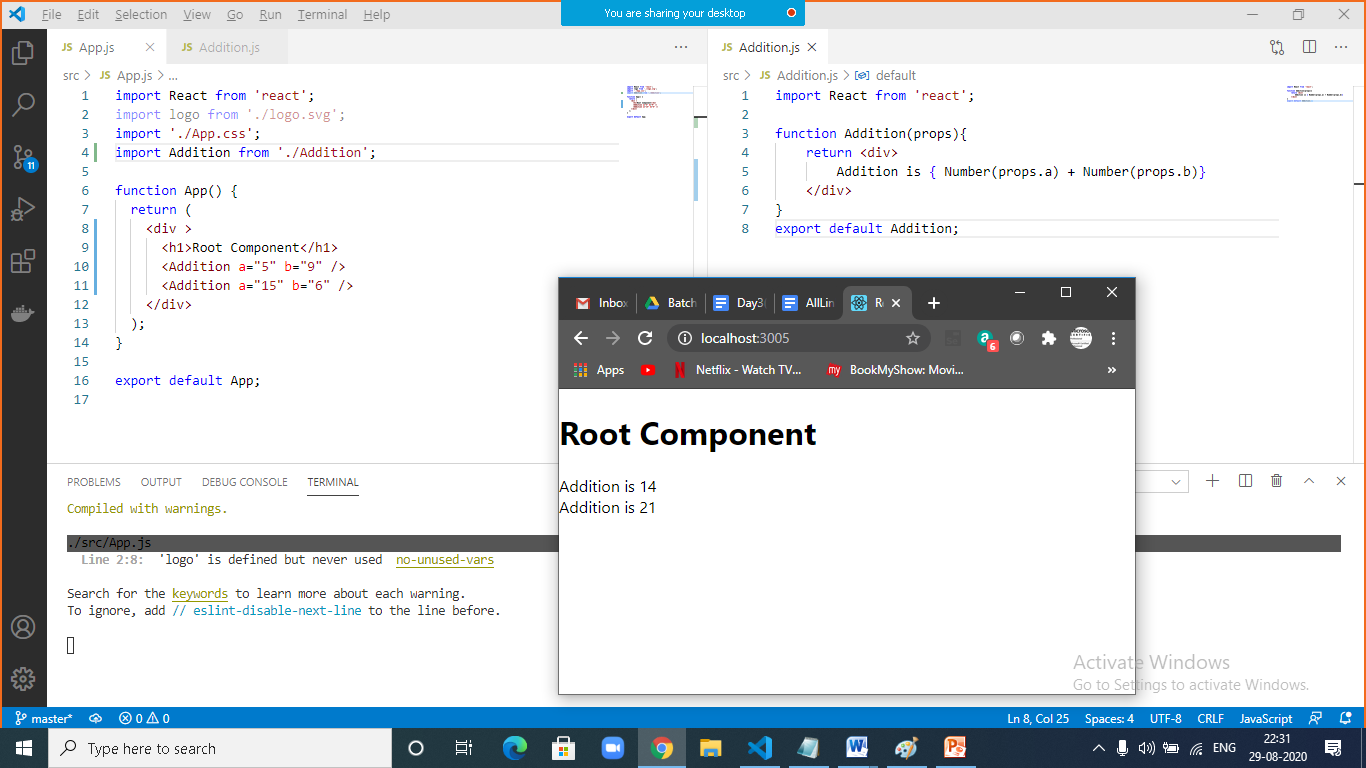
|  |  |
| --- | --- |
|  |  |
|  | import React from 'react';  function showEmployees(heading,employees){  return (  <div>  <h1>{heading}</h1>  <table border="1">  <tr>  <th>ID</th> <th>Name of the Employee</th> <th>Job</th> <th>Salary</th>  </tr>  { employees.map( e => <tr>  <td>{e.id}</td> <td> { e.ename} </td> <td>{e.job} </td> <td>{e.salary} </td>  </tr> ) }  </table >  </div> );  }  function MyComponent1(){  var employees = [  {"id":1001,"ename":"Anil","job":"Admin","salary":11000},  {"id":1002,"ename":"Bobby","job":"Clerk","salary":12000},  {"id":1003,"ename":"Cathe","job":"Clerd","salary":14000},  {"id":1004,"ename":"David","job":"Developer","salary":14000}  ];  var heading='List of Employees';  return showEmployees(heading,employees);  }  export default MyComponent1; |

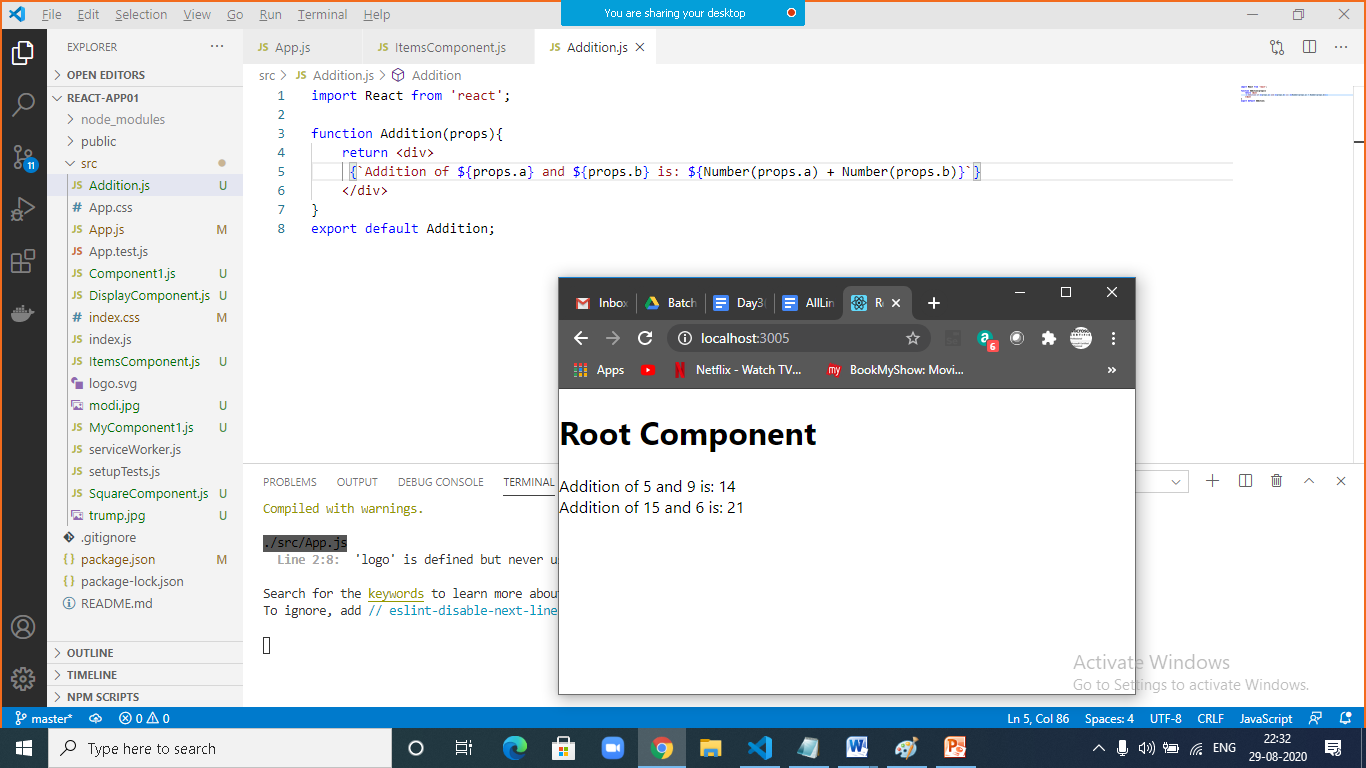






Define a component and name it as AdditionComponent takes two values in props and display sum of given numbers;





|  |  |
| --- | --- |
|  |  |
|  | import React from 'react';  function Addition(props){  return <div>  {`Addition of ${props.a} and ${props.b} is: ${Number(props.a) + Number(props.b)}`}  </div>  }  export default Addition; |

