

Experiment No. 08

Title: Design a web page using React JS.

Batch: B-4

Roll No: 16010422234

Name: Chandana Ramesh Galgali

Experiment No: 08**Aim:** To design a web page using React JS.

Resources needed: Notepad, any Web Browser and Internet.

Theory:

React (also known as React.js or ReactJS) is an open-source, front end, JavaScript library for building user interfaces or UI components. It is maintained by Facebook and a community of individual developers and companies. React can be used as a base in the development of single-page or mobile applications. However, React is only concerned with state management and rendering that state to the DOM(Document Object Model), so creating React applications usually requires the use of additional libraries for routing, as well as certain client-side functionality. ReactJS is a JavaScript library used for building reusable UI components.

Features of React

- **JSX** – JSX is a JavaScript syntax extension. It isn't necessary to use JSX in React development, but it is recommended.
- **Components** – React is all about components. You need to think of everything as a component. This will help you maintain the code when working on larger scale projects.
- **Unidirectional data flow and Flux** – React implements one-way data flow which makes it easy to reason about your app. Flux is a pattern that helps keep your data unidirectional.
- **License** – React is licensed under Facebook Inc. Documentation is licensed under CC BY 4.0.

ReactJS - Environment Setup

1. First you need to install NodeJS
2. Second install ReactJS

Install NodeJS:**Step 1:**

Visit the website www.Nodejs.org/en/. For installation on Windows, you use the MSI file and follow the prompts to install the Node.js. By default, the installer uses the Node.js distribution in C:\Program Files\nodejs. The installer should set the C:\Program Files\nodejs\bin directory in the window's PATH environment variable. Restart any open command prompts for the change to take effect. The source code written in the source file is simply javascript. The Node.js interpreter will be used to interpret and execute your javascript code. Node.js distribution comes as a binary installable for SunOS, Linux, Mac OS X, and Windows operating systems with the 32-bit (386) and 64-bit (amd64) x86 processor architectures. Next step will be to install Node.js binary distribution on Windows OS.

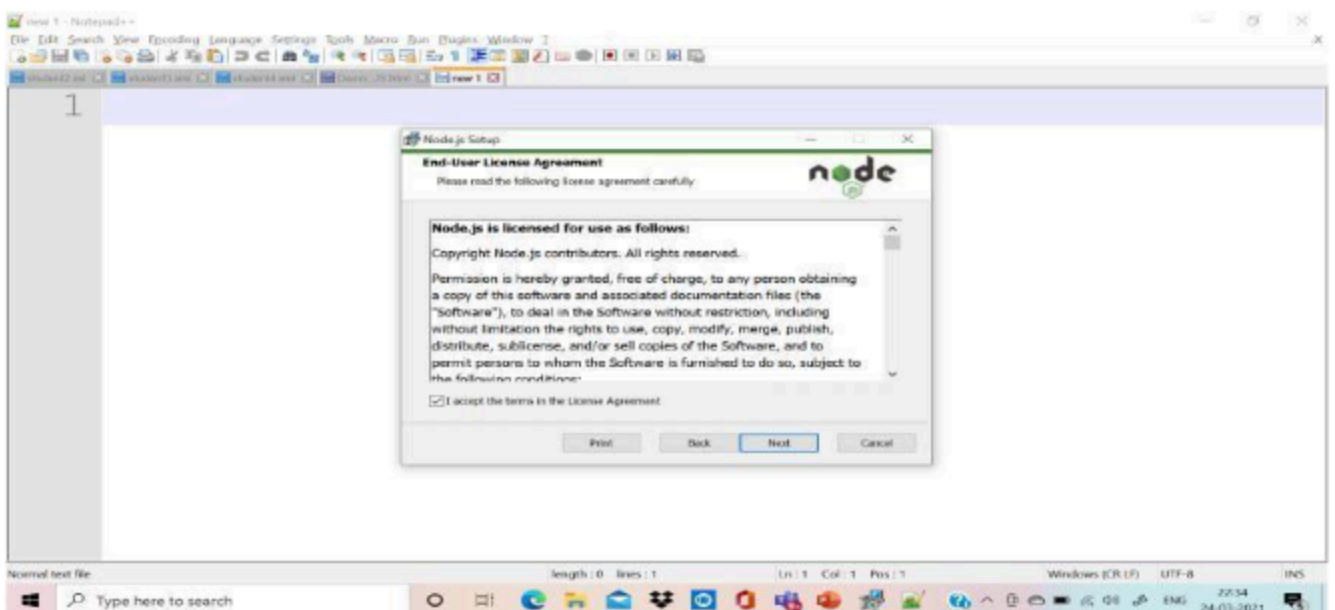
Step 2:



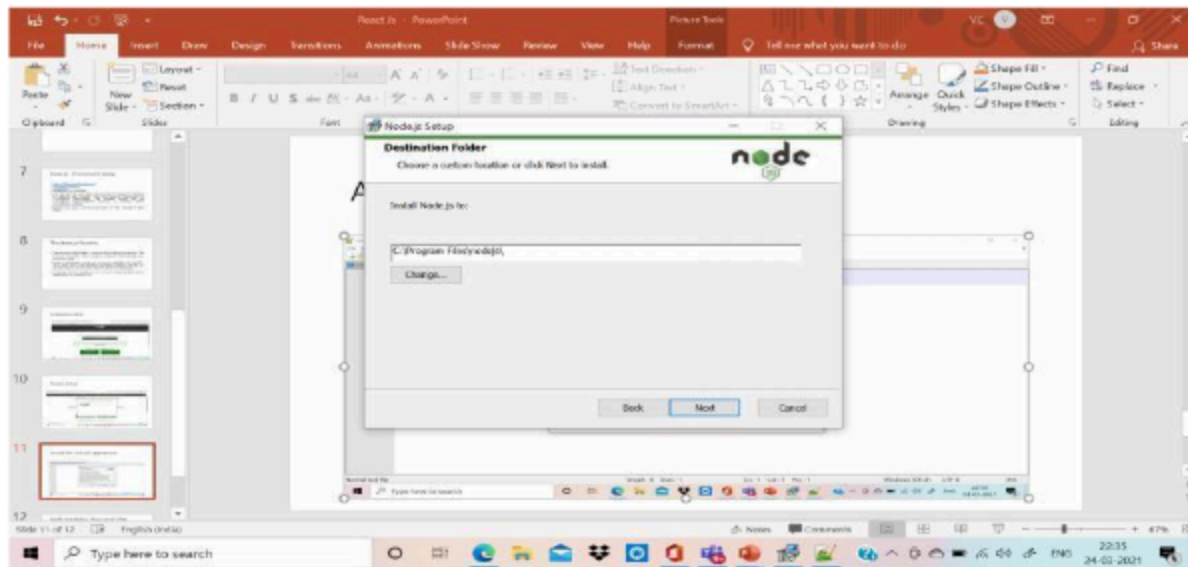
Step 3:



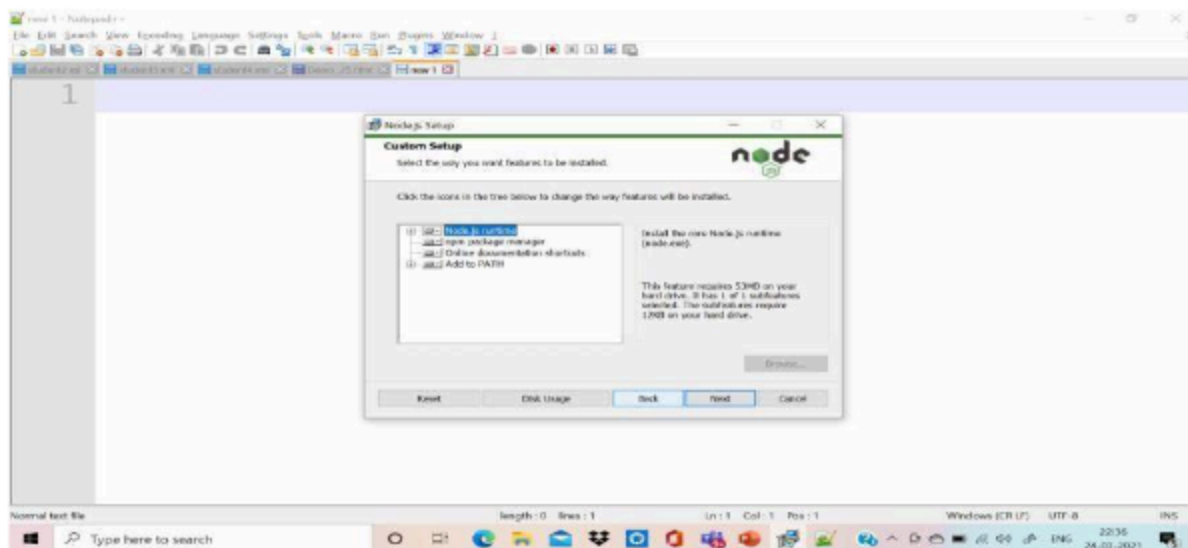
Step 4: Accept the Agreement



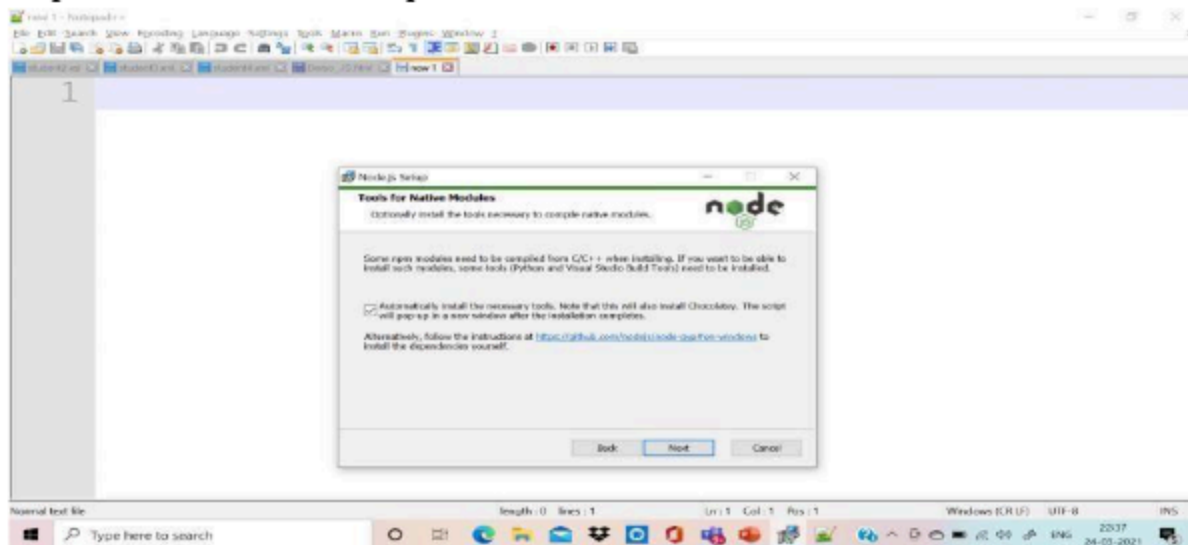
Step 5: Choose Destination Folder



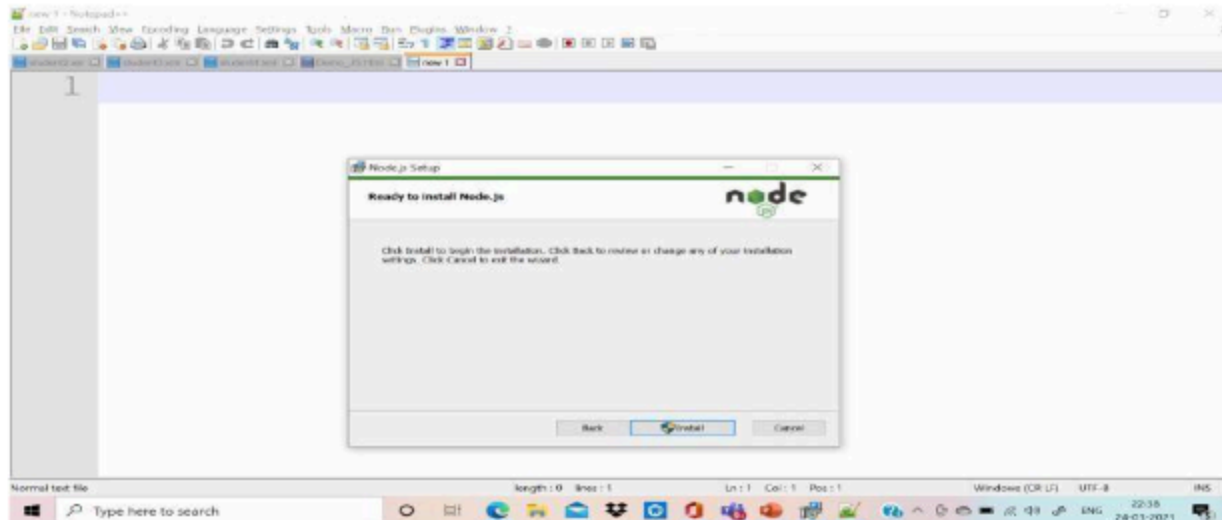
Step 6: Custom Installation setup



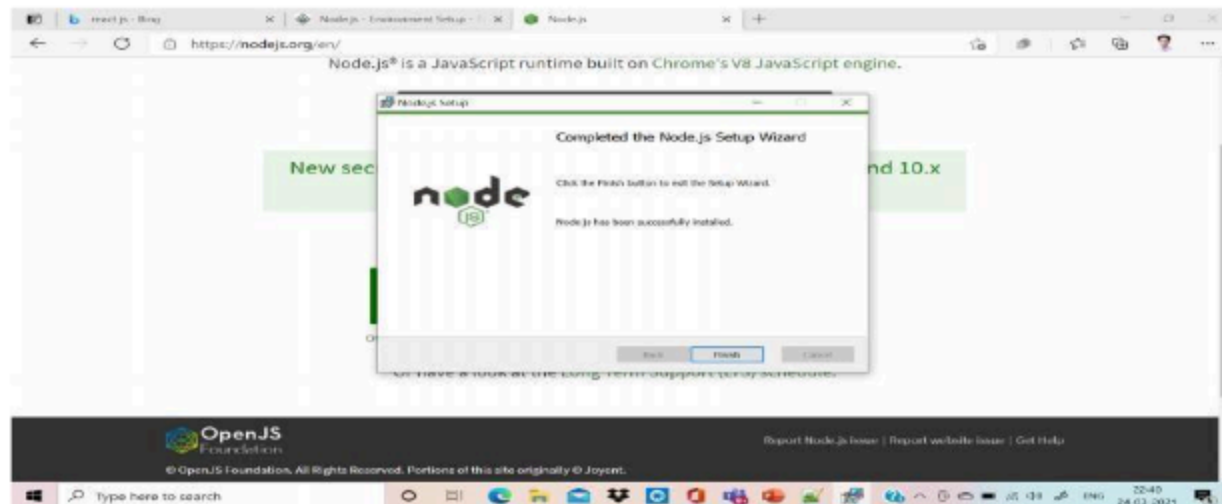
Step 7: Additional Tools Setup



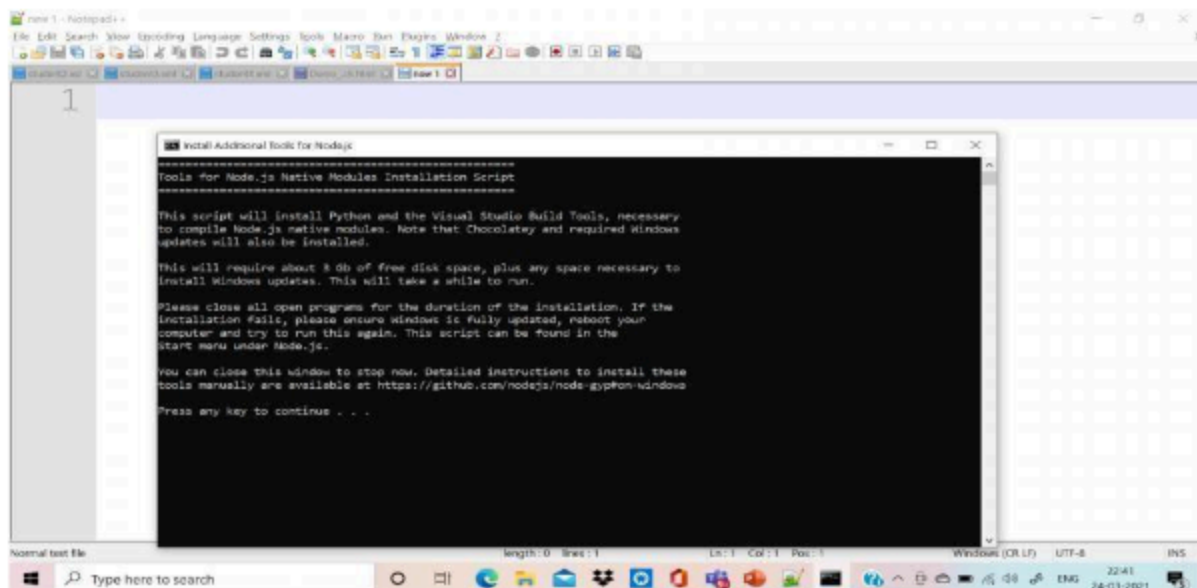
Step 8: Ready to install Nodejs



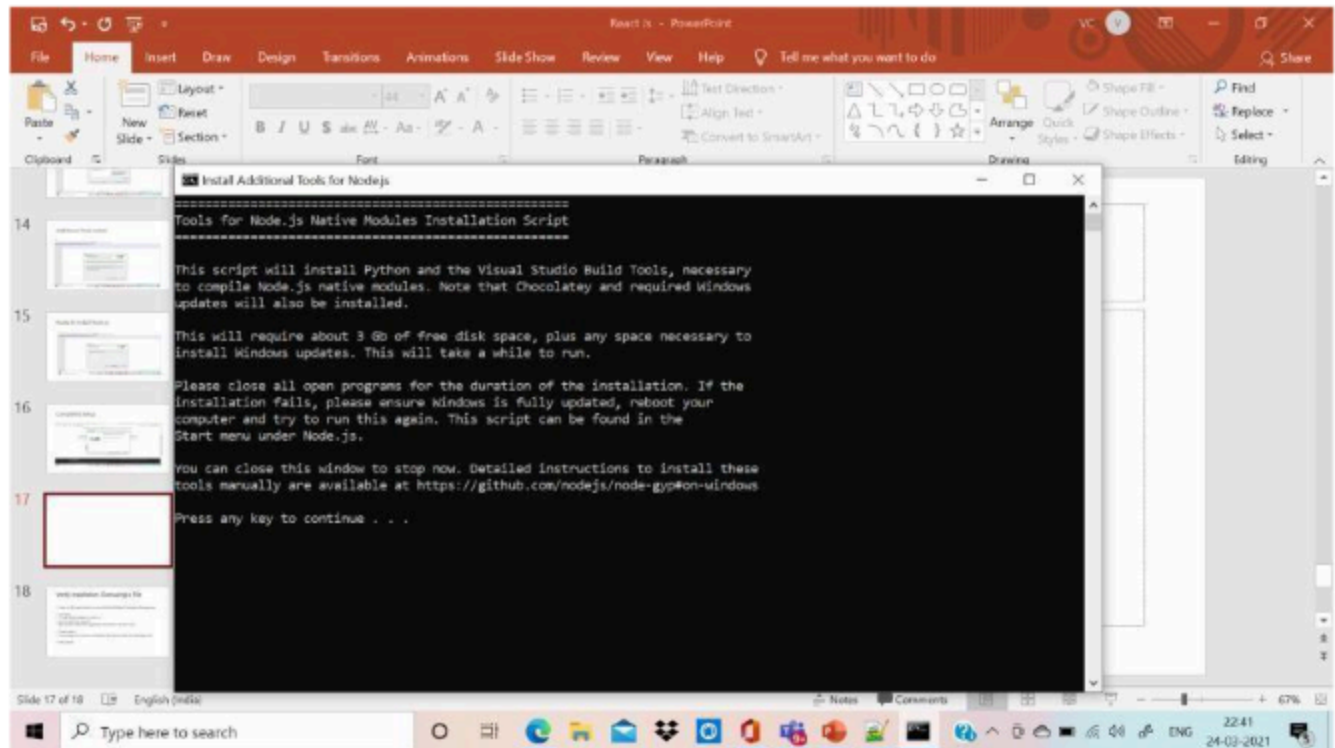
Step 9: Completed Setup



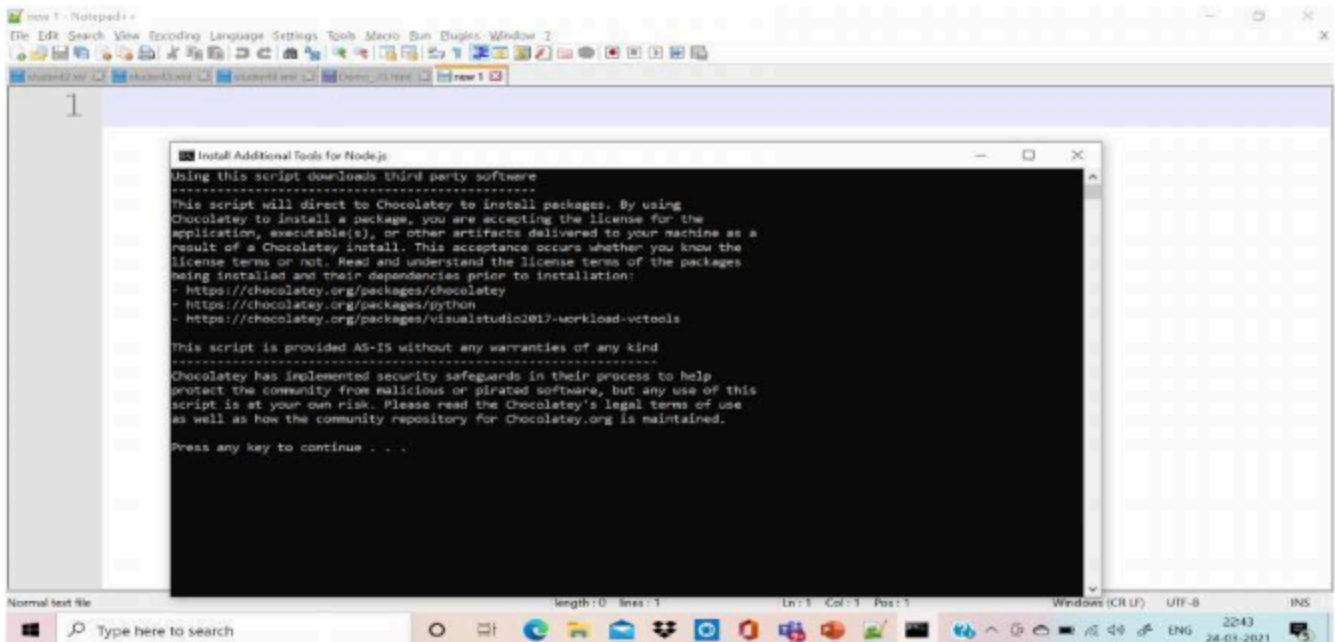
Step 10: Ready to start



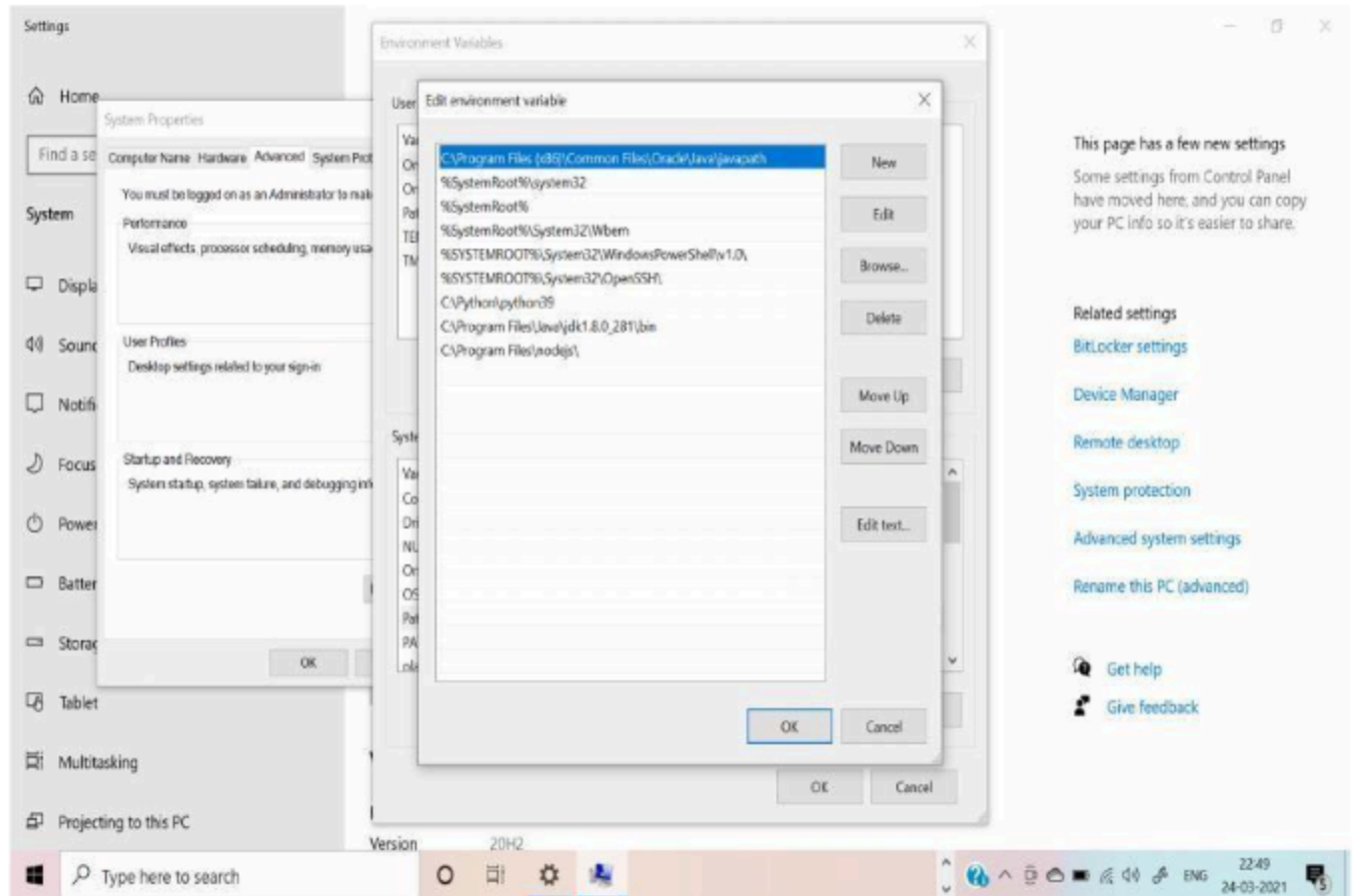
Step 11: Install Native Module



Step 12: Additional Installation

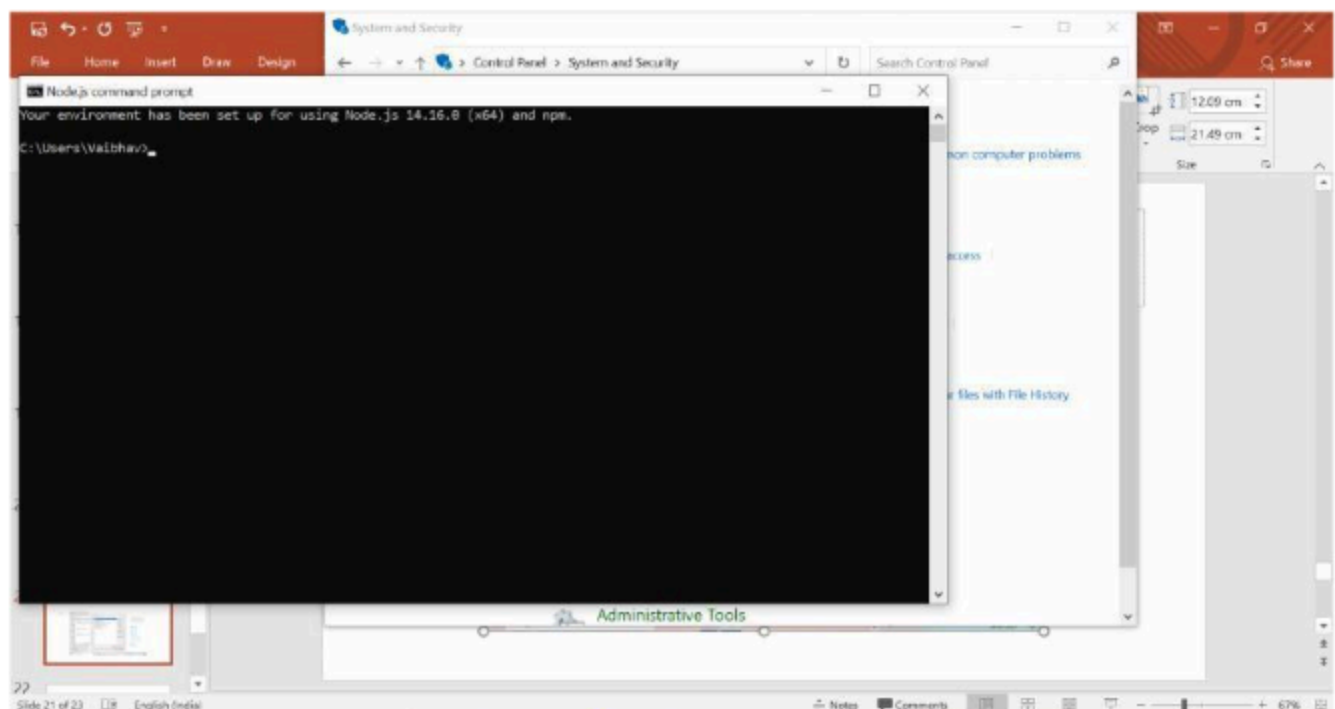


Step 13: Verify the setup



Step 14: Setup Message on Command Prompt

For this message go start menu and click the button you will find command prompt menu available with cmd.



Step 15: Check the version

```
C:\>node -v
```

```
→v14.16.0
```

Step 16: Type the Hello World and execute

Type the command and install gtypescript : `C:\>npm install -g typescript`

Create a js file named main.js on your machine (Windows) having the following code.

Live Demo

```
/* Hello, World! program in node.js */
```

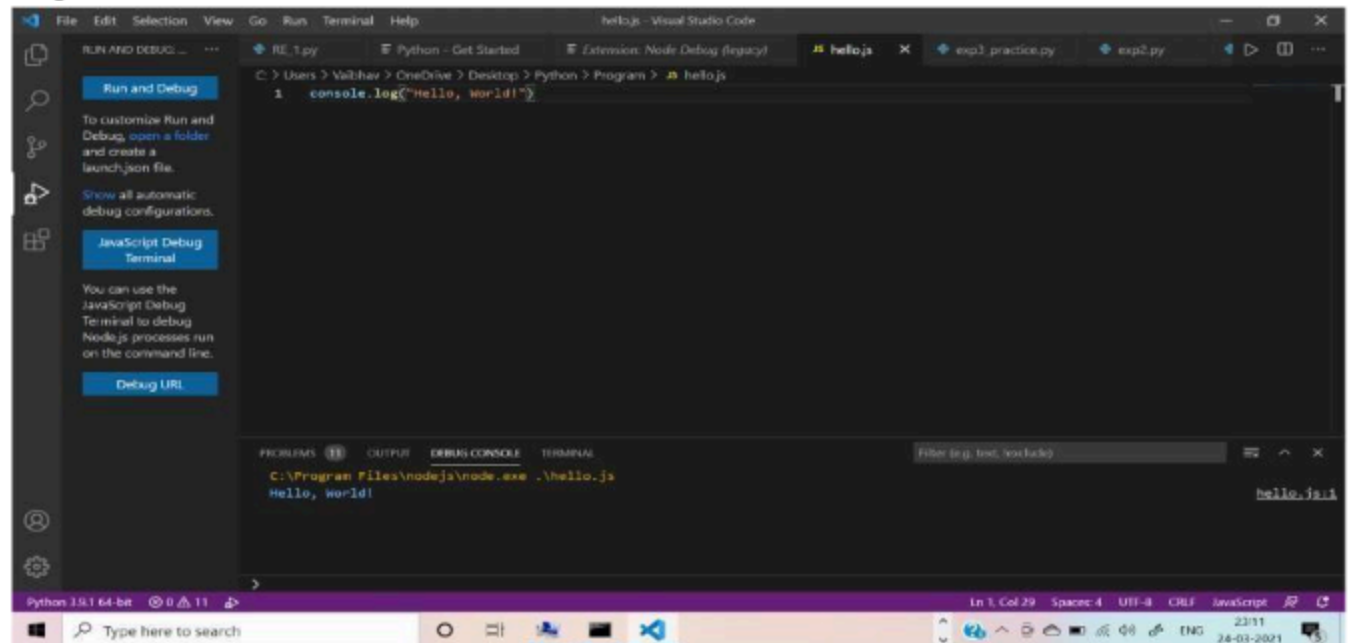
```
console.log("Hello, World!")
```

Now execute main.js file using Node.js interpreter to see the result –

```
$ node main.js
```

If everything is fine with your installation, this should produce the following result –

Hello, World!

Step 17: Install VS Code – Editor Tool**Install ReactJS:**

There are 3 ways to install ReactJS :

1. webpack
2. Babel
3. create-react-app

Webpack:

Webpack is a module bundler (manages and loads independent modules). It takes dependent modules and compiles them to a single (file) bundle. You can use this bundle while developing apps using command line or, by configuring it using the webpack.config file.

Steps:

1. Since we are using webpack to generate bundler install webpack, webpack-dev-server and webpack-cli.
2. C:\Users\username\Desktop\reactApp>npm install webpack --save
3. C:\Users\username\Desktop\reactApp>npm install webpack-dev-server --save
4. C:\Users\username\Desktop\reactApp>npm install webpack-cli --save

Or, you can install all of them in single command as –

C:\Users\username\Desktop\reactApp>npm install webpack webpack-dev-server --save

Babel:

Babel is a JavaScript compiler and transpiler. It is used to convert one source code to other. Using this you will be able to use the new ES6 features in your code where, babel converts it into plain old ES5 which can be run on all browsers.

Install babel, and its plugins babel-core, babel-loader, babel-preset-env, babel-preset-react and, html-webpack-plugin

1. C:\Users\username\Desktop\reactApp>npm install babel-core --save-dev
2. C:\Users\username\Desktop\reactApp>npm install babel-loader --save-dev
3. C:\Users\username\Desktop\reactApp>npm install babel-preset-env --save-dev
4. C:\Users\username\Desktop\reactApp>npm install babel-preset-react --save-dev
5. C:\Users\username\Desktop\reactApp>npm install html-webpack-plugin --save-dev

Or, you can install all of them in single command as –

```
C:\Users\username\Desktop\reactApp>npm install babel-core babel-loader babel-preset-env
babel-preset-react html-webpack-plugin --save-dev
```

Working with ReactJS

In this experiment , we are going to use this steps of create react app Steps are as follows:

1. Create a folder with name reactApp on the desktop to install all the required files, using the mkdir command.
C:\Users\username\Desktop>mkdir reactApp
2. Change the directory:
C:\Users\username\Desktop>cd reactApp C:\Users\vaibhav>cd C:\Users\vaibhav\reactapp\
3. Install ReactJS:
C:\Users\vaibhav\reactapp>npx create-react-app my-app

This will create a folder named my-app on the desktop and install all the required files in it.

npm (node package manager) is the dependency/package manager you get out of the box when you install Node.js. It provides a way for developers to install packages both globally and locally

npx: The npx stands for Node Package Execute and it comes with the npm, when you installed npm above 5.2.0 version then automatically npx will be installed. It is an npm package runner that can execute any package that you want from the npm registry without even installing that package. The npx is useful during a single time use package. If you have installed npm below 5.2.0 then npx is not installed in your system.

1. Delete all source files

Browse through the src folder in the generated my-app folder and remove all the files in it as shown below –

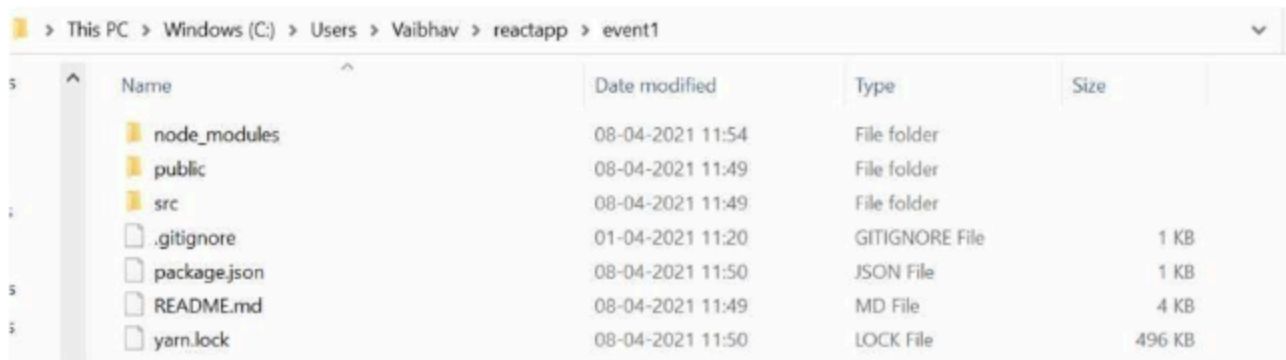
- C:\Users\Desktop>cd my-app/src
- C:\Users\Desktop\my-app\src>del *
- C:\Users\Desktop\my-app\src*, Are you sure (Y/N)? y

2. Add files with names index.css and index.js in the src folder as –

- C:\Users\Desktop\my-app\src>type nul > index.css
- C:\Users\Desktop\my-app\src>type nul > index.js

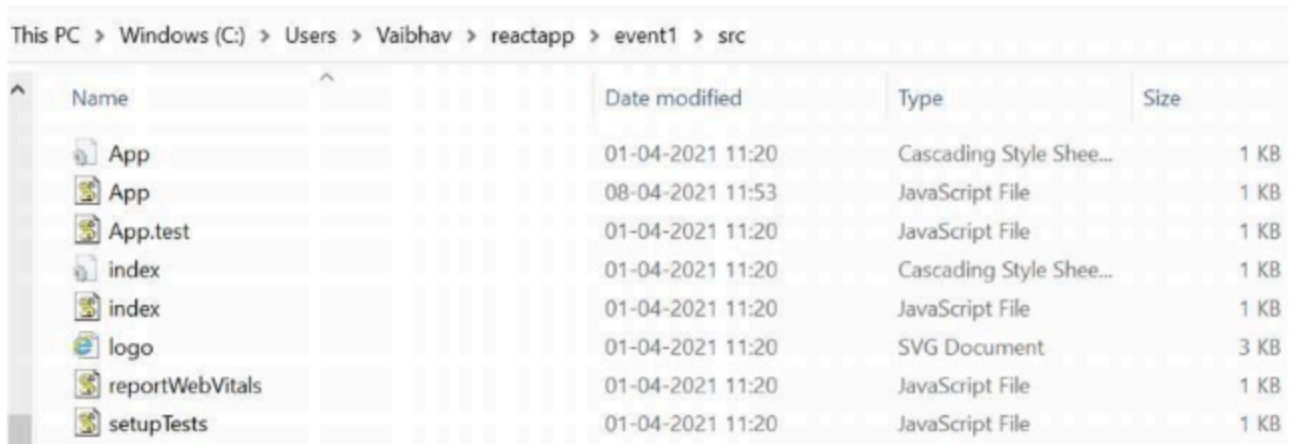
3. Skip step 4 and 5 and only delete the file called App.js

4. Locate your code into these folder as event1 as shown below:



Name	Date modified	Type	Size
node_modules	08-04-2021 11:54	File folder	
public	08-04-2021 11:49	File folder	
src	08-04-2021 11:49	File folder	
.gitignore	01-04-2021 11:20	GITIGNORE File	1 KB
package.json	08-04-2021 11:50	JSON File	1 KB
README.md	08-04-2021 11:49	MD File	4 KB
yarn.lock	08-04-2021 11:50	LOCK File	496 KB

5. Open the src folder as shown below:



6. Choose the App.js file and indite code which is given below.

```
import React, {Component} from 'react';

class App extends React.Component { constructor(props) {

super(props); this.state = {

companyName: "

};

}

changeText(event) { this.setState({

companyName: event.target.value

});

}

render() { return (

<div>

<h2>Simple Event Example</h2>

<label htmlFor="name">Enter company name: </label>

<input type="text" id="companyName" onChange={this.changeText.bind(this)}/>
```

```
<h4>You entered: { this.state.companyName }</h4>
```

```
</div>
```

```
);
```

```
}
```

```
} export default App;
```

7. Execute the React Code:

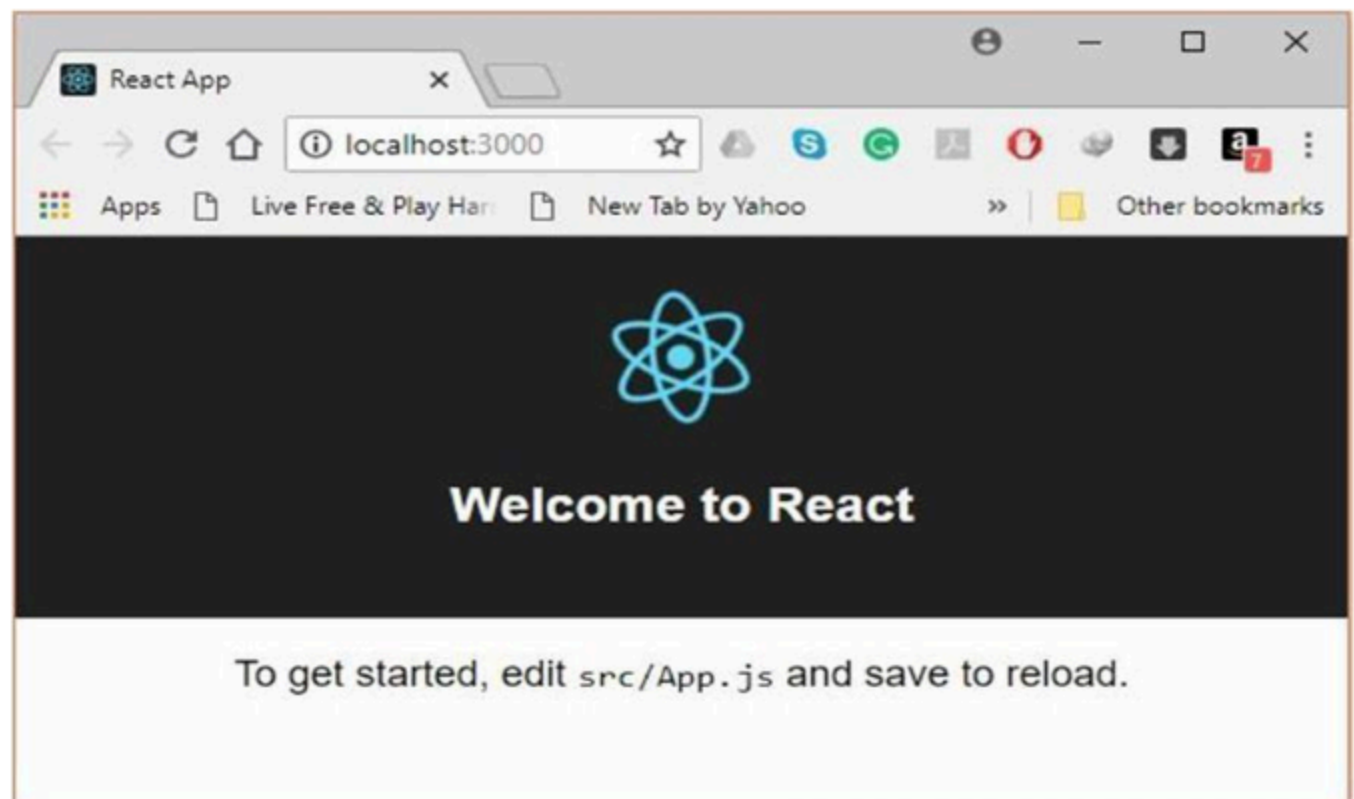
To run the react code you have to type the command on Vs Code Terminal as it is:

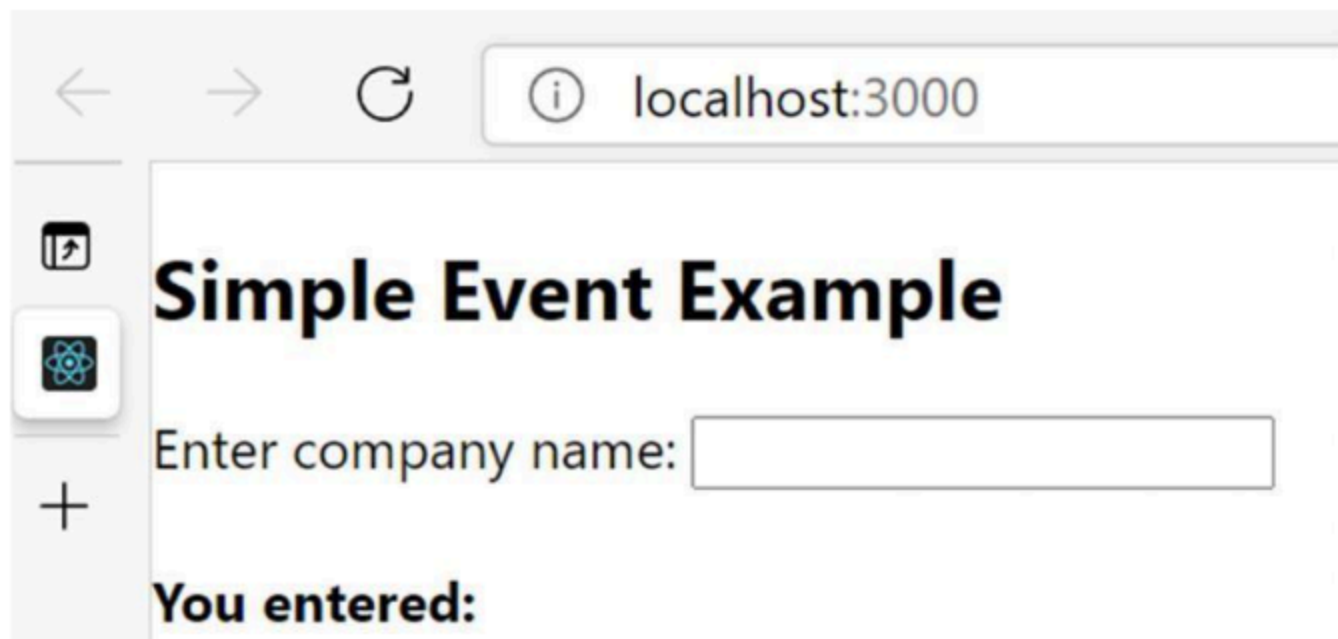
`npm start`

`yarn start`(if you install yarn utility)

To do this on, terminal of VS CODE,
`C:\Users\Vaibhav\reactapp\myevent\npm start.`

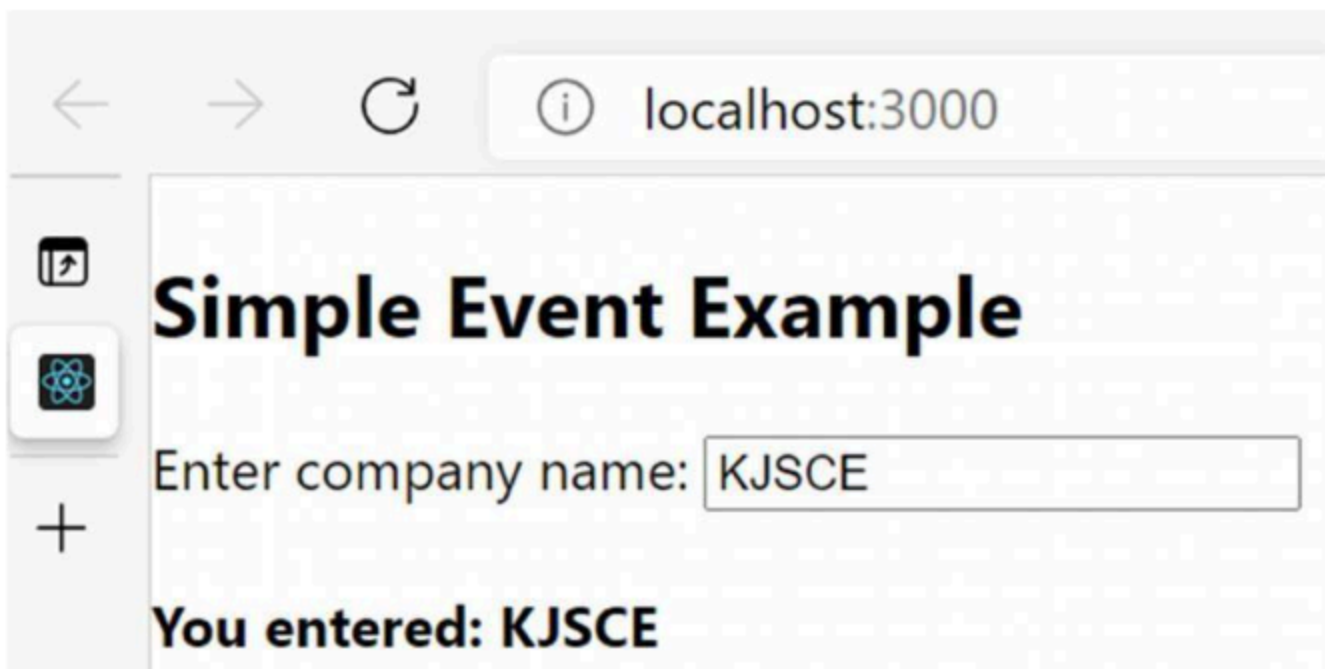
By default port number 3000 will start on the web browser with the code execution.





A screenshot of a web browser window. The address bar shows 'localhost:3000'. The page has a sidebar on the left with three icons: a document with an arrow, a network diagram, and a plus sign. The main content area has the title 'Simple Event Example' in large bold black font. Below the title is a text input field with the placeholder text 'Enter company name:'. Below the input field, the text 'You entered:' is displayed in bold black font.

8. After running the event the outlook:

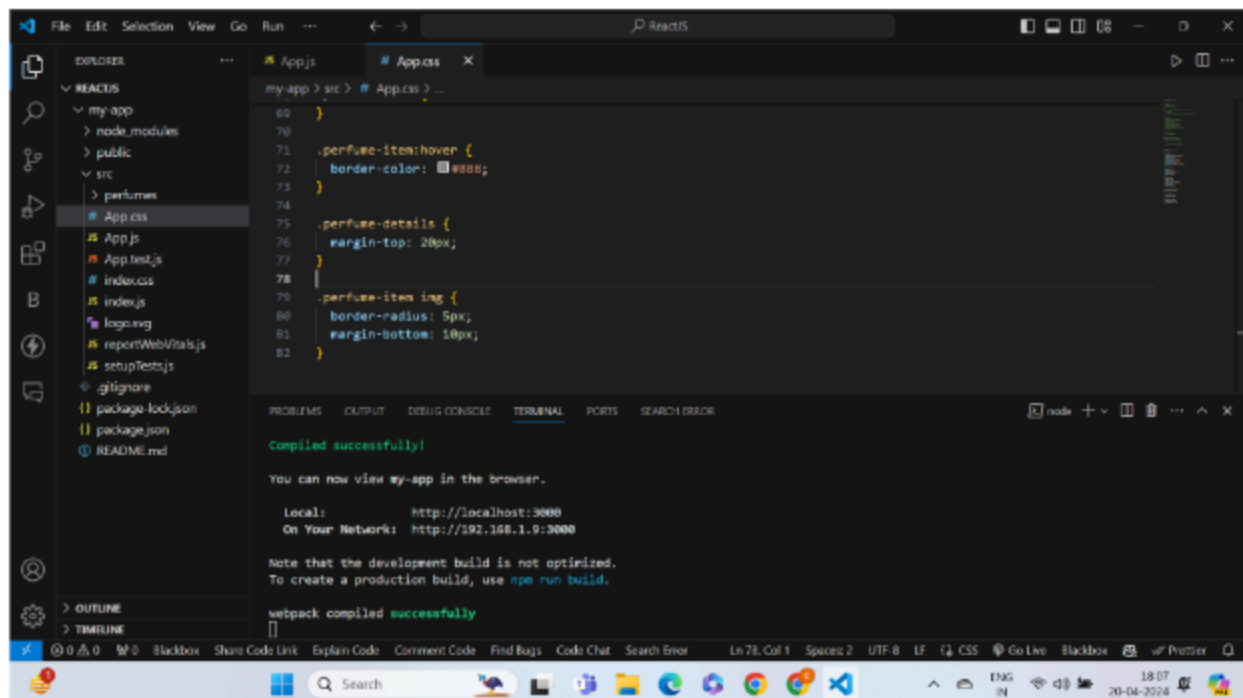
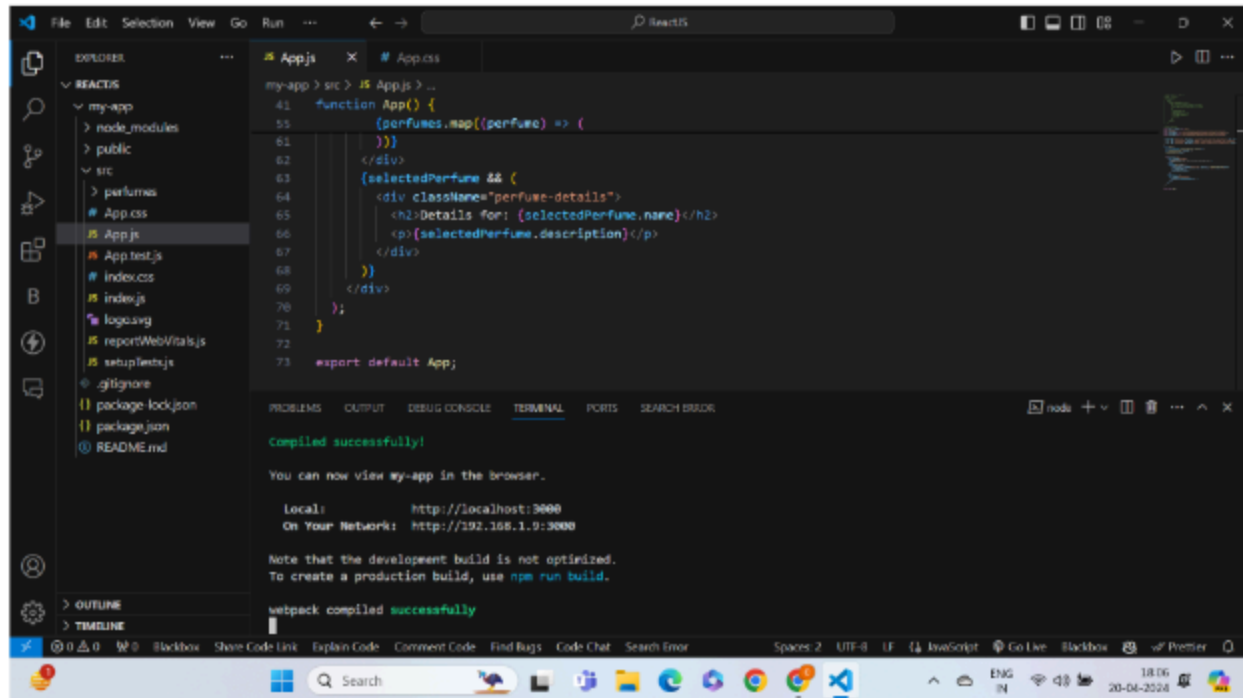


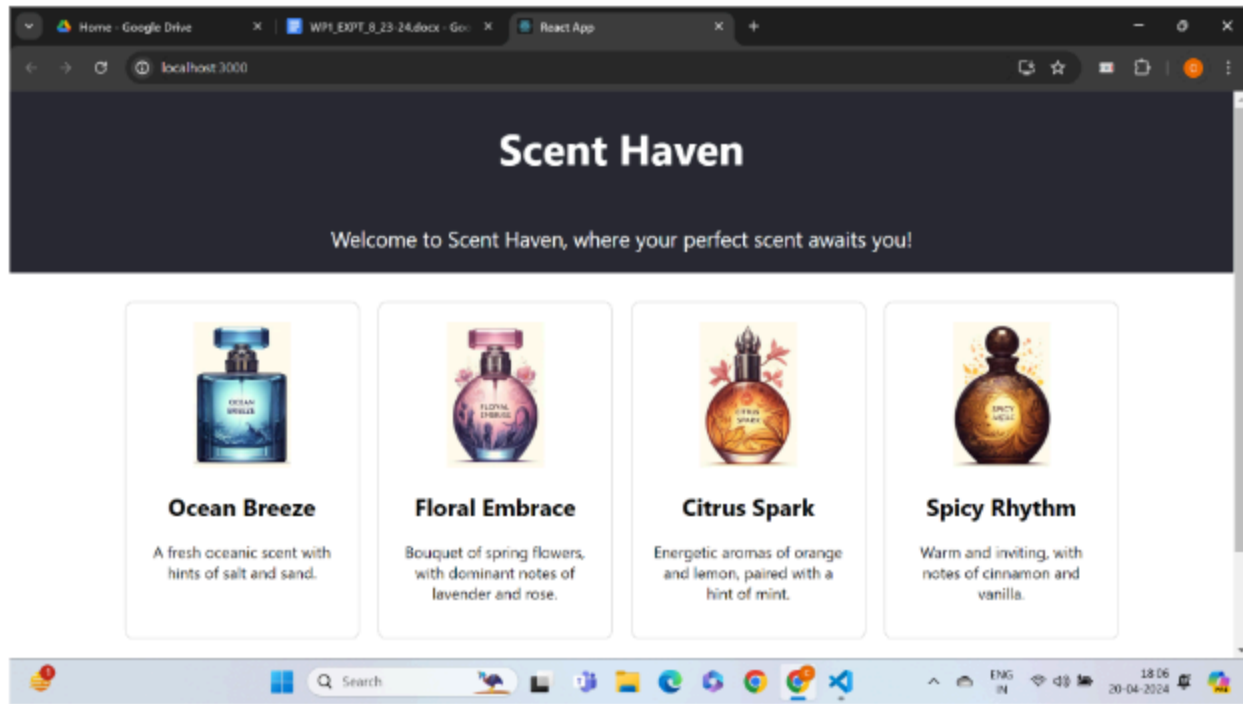
A screenshot of the same web browser window after an event. The address bar still shows 'localhost:3000'. The sidebar and title are the same. The text input field now contains the text 'KJSCE'. Below the input field, the text 'You entered: KJSCE' is displayed in bold black font.

Activities:

To design a web page using React JS on your theme to manipulate the DOM elements.

Results: (Document printout as per the format discussed by the faculty)





Display of the designed webpage along with the code.

JS FILE

```
import React, { useState } from 'react';

import './App.css';

import ocean_breeze from
'C:/Users/chand/Downloads/ReactJS/my-app/src/perfumes/ocean_breeze.jpg';

import floral_embrace from
'C:/Users/chand/Downloads/ReactJS/my-app/src/perfumes/floral_embrace.jpg'
;

import citrus_spark from
'C:/Users/chand/Downloads/ReactJS/my-app/src/perfumes/citrus_spark.jpg';

import spicy_rhythm from
'C:/Users/chand/Downloads/ReactJS/my-app/src/perfumes/spicy_rhythm.jpg';

const perfumes = [
  { id: 1, name: "Ocean Breeze", description: "A fresh oceanic scent with
hints of salt and sand.", image: ocean_breeze },
  { id: 2, name: "Floral Embrace", description: "Bouquet of spring
flowers, with dominant notes of lavender and rose.", image:
floral_embrace },
  { id: 3, name: "Citrus Spark", description: "Energetic aromas of orange
```

```

and lemon, paired with a hint of mint.", image: citrus_spark },
  { id: 4, name: "Spicy Rhythm", description: "Warm and inviting, with
notes of cinnamon and vanilla.", image: spicy_rhythm }
];

function App() {
  const [selectedPerfume, setSelectedPerfume] = useState(null);

  const handleSelectPerfume = (perfume) => {
    setSelectedPerfume(perfume);
  };

  return (
    <div className="App">
      <header className="App-header">
        <h1>Scent Haven</h1>
        <p>Welcome to Scent Haven, where your perfect scent awaits
you!</p>
      </header>
      <div className="perfume-list">
        {perfumes.map((perfume) => (
          <div key={perfume.id} className="perfume-item" onClick={() =>
handleSelectPerfume(perfume)}>
            <img src={perfume.image} alt={perfume.name} style={{ width:
"100px", height: "150px" }} />
            <h2>{perfume.name}</h2>
            <p>{perfume.description}</p>
          </div>
        ))}
      </div>
      {selectedPerfume && (
        <div className="perfume-details">
          <h2>Details for: {selectedPerfume.name}</h2>
          <p>{selectedPerfume.description}</p>
        </div>
      )}
    </div>
  );
}

export default App;

```

CSS FILE

```
.App {
  text-align: center;
}

.App-header {
  background-color: #282c34;
  min-height: 150px;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  font-size: calc(10px + 2vmin);
  color: white;
}

.perfume-list {
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  padding: 20px;
}

.perfume-item {
  margin: 10px;
  padding: 20px;
  border: 1px solid #ccc;
  border-radius: 8px;
  width: 200px;
  cursor: pointer;
}

.perfume-item:hover {
  border-color: #888;
}

.perfume-details {
  margin-top: 20px;
}

.perfume-item img {
  border-radius: 5px;
  margin-bottom: 10px;
}
```

Scent Haven

Welcome to Scent Haven, where your perfect scent awaits you!



Ocean Breeze

A fresh oceanic scent with hints of salt and sand.



Floral Embrace

Bouquet of spring flowers, with dominant notes of lavender and rose.



Citrus Spark

Energetic aromas of orange and lemon, paired with a hint of mint.



Spicy Rhythm

Warm and inviting, with notes of cinnamon and vanilla.

Scent Haven

Welcome to Scent Haven, where your perfect scent awaits you!



Ocean Breeze

A fresh oceanic scent with hints of salt and sand.



Floral Embrace

Bouquet of spring flowers, with dominant notes of lavender and rose.



Citrus Spark

Energetic aromas of orange and lemon, paired with a hint of mint.



Spicy Rhythm


Warm and inviting, with notes of cinnamon and vanilla.

Details for: Ocean Breeze

A fresh oceanic scent with hints of salt and sand.


Scent Haven

Welcome to Scent Haven, where your perfect scent awaits you!




Ocean Breeze

A fresh oceanic scent with hints of salt and sand.




Floral Embrace

Bouquet of spring flowers, with dominant notes of lavender and rose.



Citrus Spark

Energetic aromas of orange and lemon, paired with a hint of mint.



Spicy Rhythm


Warm and inviting, with notes of cinnamon and vanilla.

Details for: Floral Embrace

Bouquet of spring flowers, with dominant notes of lavender and rose.


Scent Haven

Welcome to Scent Haven, where your perfect scent awaits you!




Ocean Breeze

A fresh oceanic scent with hints of salt and sand.




Floral Embrace

Bouquet of spring flowers, with dominant notes of lavender and rose.



Citrus Spark

Energetic aromas of orange and lemon, paired with a hint of mint.

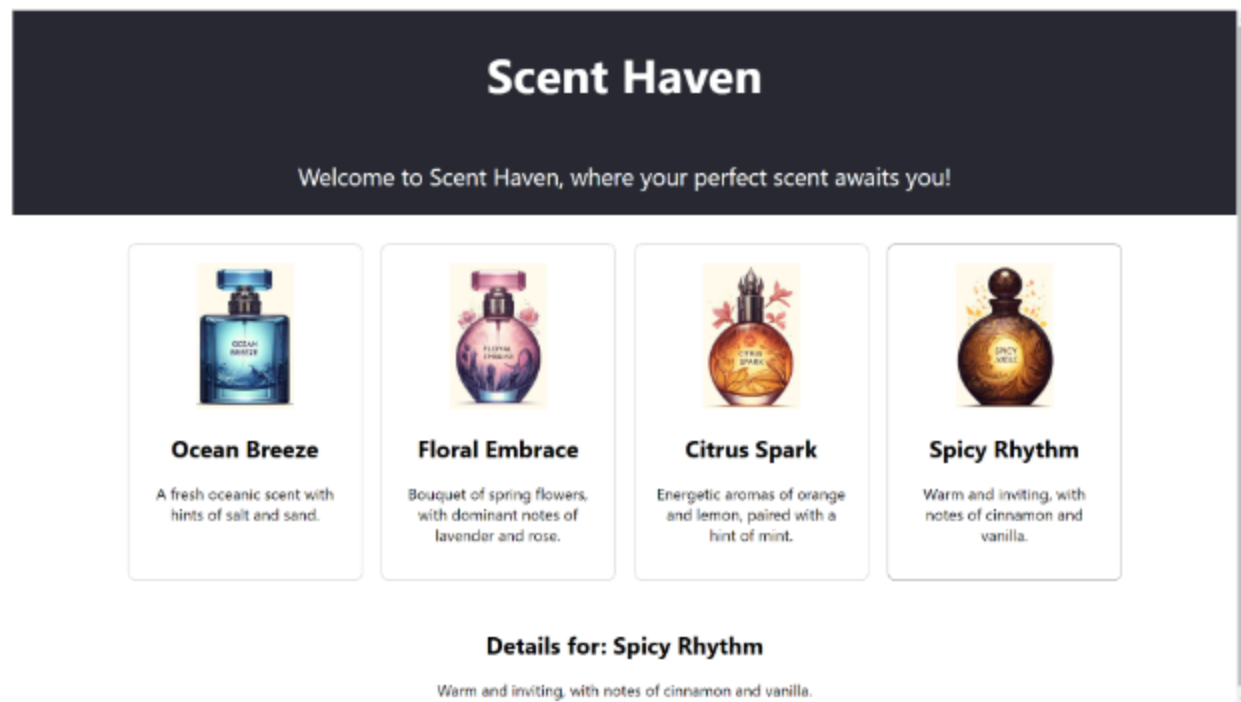


Spicy Rhythm

Warm and inviting, with notes of cinnamon and vanilla.

Details for: Citrus Spark

Energetic aromas of orange and lemon, paired with a hint of mint.



Questions:

1. What are the different components of ReactJS?

Ans: ReactJS is a popular JavaScript library for building user interfaces, particularly for single-page applications. Here are the key components of ReactJS:

- **Components:** The building blocks of any React application are its components. Components are independent and reusable bits of code. They can be class components or functional components, each with their own lifecycle methods for managing state and effects.
- **JSX:** JSX (JavaScript XML) is a syntax extension for JavaScript, used in React to describe what the UI should look like. It allows HTML to be written inside JavaScript and makes the syntax easier to understand.
- **Props:** Short for properties, props are a way of passing data from parent to child components. They are read-only and help components interact as function arguments.
- **State:** State is a set of data that determines the behavior of a component and how it will render. It is mutable and local to the component where it is declared.
- **Context:** Provides a way to pass data through the component tree without having to pass props down manually at every level.
- **Hooks:** Introduced in React 16.8, hooks are functions that let you "hook into" React state and lifecycle features from function components. Common hooks are `useState`, `useEffect`, and `useContext`.
- **Lifecycle Methods:** In class components, these methods are used to perform actions at specific

points in a component's lifecycle, such as creation, updating, and destruction.

- **Redux/Flux:** Libraries for managing and centralizing application state, often used in React applications to manage state that is needed across many components.

2. What is Virtual DOM? How does virtual DOM work? What is the purpose of render of react DOM?

Ans: The Virtual DOM is a programming concept used in React and other libraries as a performance optimization. It is a lightweight copy of the real DOM (Document Object Model), and it exists entirely in memory. Instead of updating the real DOM directly, React updates the Virtual DOM, which acts like a staging area for changes.

1. When a component's state changes, React creates a new Virtual DOM tree, representing the updated state.
2. React then compares this new Virtual DOM tree with the previous version using a diffing algorithm.
3. React identifies which objects have changed in the Virtual DOM.
4. Only those changes are then efficiently updated on the real DOM.

The render method is one of the most crucial lifecycle methods in React. Its main purpose is to examine `this.props` and `this.state` and return HTML via JSX. In simple terms, whenever React needs to update the UI, it will call the render method to get a new description of the component's UI based on the current state and props. This process involves generating the appropriate Virtual DOM, which React will use to update the actual DOM effectively.

Outcomes: Implement web application using React JS, Angular JS, JSON and CBOR

Conclusion: (Conclusion to be based on objectives and outcomes achieved)

The experiment effectively demonstrated the capabilities of ReactJS in building dynamic user interfaces efficiently through the use of components, JSX, and state management. The objectives of designing an interactive web page were successfully met, showcasing the practical application of theoretical concepts.

Grade: AA/AB/BB/BC/CC/CD/DD/FF

Signature of faculty in-charge with date

References:

Books/ Journals/ Websites

1. React – A JavaScript library for building user interfaces (reactjs.org)
2. "React - A JavaScript library for building user interfaces". React. Retrieved 7 April 2018.
3. Krill, Paul (May 15, 2014). "React: Making faster, smoother UIs for data-driven Web apps". InfoWorld.
4. Hemel, Zef (June 3, 2013). "Facebook's React JavaScript User Interfaces Library Receives Mixed Reviews". InfoQ.
5. Dawson, Chris (July 25, 2014). "JavaScript's History and How it Led To ReactJS". The New

Stack.

6. Dere, Mohan (2018-02-19). "How to integrate create-react-app with all the libraries you need to make a great app". freeCodeCamp. Retrieved 2018-06-14.
 7. React Tutorial (w3schools.com)
 8. ReactJS - Overview - Tutorialspoint
 9. ReactJS Tutorials - GeeksforGeeks
-