**Assignment-8**

**MODULE: 9 ReactJs Intro**

1. **What is React Js?**

* React is a JavaScript library for building user interfaces.
* React is used to build single-page applications.
* React allows us to create reusable UI components.

1. **What is NPM in React Js?**

* As a tool for managing and installing JavaScript packages and dependencies, NPM (Node Package Manager) in React.js makes it easier to create and deploy React applications.

1. **What is the Role of Node Js in react Js?**

* Node.js in React.js serves as a runtime environment for server-side rendering, enabling server-side rendering and supporting the build process, along with managing dependencies through NPM.

1. **What is CLI command In React Js?**

* CLI stands for Command Line Interface.
* CLI commands in React.js are predefined commands like "create-react-app" or "npm start" used to initiate, develop, test, and build React applications via the command line.

1. **What are Components in React Js?**

* In React JS components are reusable, separate pieces of the user interface that combine presentation and logic.
* By combining these modular parts, they make it possible to create complicated user interfaces.

1. **What are Header and Content Components in React Js?**

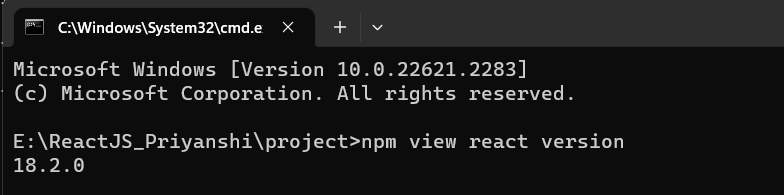
* In React.js, the Header and Content components generally stand in for various web application sections.
* Header: Contains top-of-page elements like navigation, logos, authentication, ensuring a consistent top interface across various pages.
* Content: Displays dynamic core content, such as text, images, forms, on web pages or views, offering reusability.

1. **How to install React Js on Windows, Linux Operating System? How to Install NPM and How to check version of NPM?**

* Installing React.js on Windows and Linux, including NPM installation and version check
* Install Node.js :Download and run the Node.js installer from the official website for both Windows and Linux . : [website.:https://nodejs.org/en/download/current](https://nodejs.org/en/download/current)
* Verify Installation: Open your command prompt or terminal and verify Node.js installation by running `node -v` and `npm -v` commands.
* Install React: Globally install Create React App using `npm install -g create-react-app`.
* Create React Project: Create a new React project with `npx create-react-app my-react-app` (replace "my-react-app" with your project name).
* Navigate to Project: Change to your project directory using `cd my-react-app`.
* Start Development Server: Launch the development server with `npm start`.
* Access Your React App: Open your web browser and visit `http://localhost:3000` to view your React app.
* Check npm Version: In your command prompt or terminal, run `npm -v` to confirm the installed npm version.

1. **How to check the version of React Js?**

* To check the version of React JS using the command mentioned below on our command line.



1. **How to change components of React Js?**

* To change components in React.js, you need to:
* Locate the component you want to change.
* Edit the component's code to implement the desired changes.
* Save the file.
* Test the component to ensure it functions as intended.
* Repeat these steps as needed for different components in your React application. React's hot-reloading feature will update your changes in real-time during development.

1. **How to Create a List View in React Js?**

* index.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8" />

<link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<meta name="theme-color" content="#000000" />

<title>React App</title>

<link

href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"

rel="stylesheet"

/>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>

</head>

<body>

<div id="root"></div>

</body>

</html>

* App.js

import React from "react";

import "./App.css";

import ListView from "./ListView";

function App() {

return (

<>

<div className="App">

<ListView />

</div>

</>

);

}

export default App;

* ListView.jsx

import React, { useState } from "react";

import "./App.css";

function ListView() {

const [items, setItems] = useState([

{ id: 1, text: "Use Array.map", checked: false },

{ id: 2, text: "Not a for loop", checked: false },

{ id: 3, text: "Give each item a unique key", checked: false },

{ id: 4, text: "Avoid using array index as the key", checked: false },

// Add more items as needed

]);

const toggleCheckbox = (itemId) => {

setItems((prevItems) =>

prevItems.map((item) =>

item.id === itemId ? { ...item, checked: !item.checked } : item

)

);

};

return (

<div className="list-view">

<h1 className="title">The "React Way" to Render a List</h1>

{items.map((item) => (

<div key={item.id} className="list-item">

<div className="checkbox-container">

<label className="checkbox-label">

<input

type="checkbox"

checked={item.checked}

onChange={() => toggleCheckbox(item.id)}

/>

<span className="checkbox-custom"></span>

</label>

</div>

<div className={`item-text ${item.checked ? "checked" : ""}`}>

{item.text}

</div>

</div>

))}

</div>

);

}

export default ListView;

* App.css

.list-view {

background-color: rgb(108, 175, 37);

padding: 20px;

}

.list-item {

background-color: white;

color: grey;

padding: 10px;

margin-bottom: 10px;

border-radius: 10px;

display: flex;

align-items: center;

}

.checkbox {

width: 20px;

height: 20px;

border-radius: 50%;

background-color: white;

border: 2px solid grey;

margin-right: 10px;

display: flex;

align-items: center;

justify-content: center;

cursor: pointer;

}

.checkbox input[type="checkbox"] {

display: none;

}

.checkmark {

/\* src/components/ListView.css \*/

body {

background-color: lightgreen;

margin: 0;

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

}

.list-view {

background-color: white;

padding: 20px;

border-radius: 10px;

box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.2);

text-align: center;

}

.title {

text-align: center;

}

.list-item {

display: flex;

align-items: center;

background-color: white;

color: grey;

margin-bottom: 10px;

border-radius: 10px;

overflow: hidden;

}

.checkbox-container {

display: flex;

align-items: center;

justify-content: center;

width: 30px;

height: 30px;

background-color: lightgreen;

border-radius: 50%;

}

.checkbox-label {

cursor: pointer;

display: inline-block;

position: relative;

}

.checkbox-label input[type="checkbox"] {

display: none;

}

.checkbox-custom {

position: absolute;

top: 50%;

left: 50%;

transform: translate(-50%, -50%);

width: 15px;

height: 15px;

background-color: white;

border-radius: 50%;

display: none;

}

.checkbox-label input[type="checkbox"]:checked + .checkbox-custom {

display: block;

}

.item-text {

padding: 10px;

flex: 1;

}

width: 12px;

height: 12px;

border-radius: 50%;

background-color: white;

border: 2px solid grey;

display: none;

}

.checkbox input[type="checkbox"]:checked + .checkmark {

display: block;

}

.item-text {

flex: 1;

}

* Output:



1. **Create Increment decrement state change by button click?**

* index.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8" />

<link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<meta name="theme-color" content="#000000" />

<title>React App</title>

<link

href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"

rel="stylesheet"

/>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>

</head>

<body>

<div id="root"></div>

</body>

</html>

* App.css

.App {

text-align: center;

padding: 20px;

display: flex;

flex-direction: column;

align-items: center;

justify-content: center;

min-height: 100vh;

}

h1 {

font-size: 24px;

margin: 0;

}

.count-container {

background-color: white;

border-radius: 5px;

padding: 20px;

margin: 20px 0;

box-shadow: 0px 0px 5px rgba(0, 0, 0, 0.5);

}

.count {

font-size: 48px;

color: black;

margin: 0;

}

.button-container {

display: flex;

justify-content: space-between;

width: 100%;

}

.button {

background-color: gray;

color: white;

border: none;

border-radius: 5px;

padding: 10px 20px;

font-size: 16px;

cursor: pointer;

}

.button:hover {

background-color: darkgray;

}

* App.js

import logo from "./logo.svg";

import "./App.css";

import React, { useState } from "react";

function App() {

const [count, setCount] = useState(0);

const increment = () => {

setCount(count + 1);

};

const decrement = () => {

if (count > 0) {

setCount(count - 1);

}

};

const reset = () => {

setCount(0);

};

return (

<div className="App" style={{ backgroundColor: "blue" }}>

<h1 style={{ color: "white" }}>React Web</h1>

<div className="count-container">

<p className="count">{count}</p>

</div>

<div className="button-container">

<button

className="button"

onClick={decrement}

style={{ backgroundColor: "gray" }}

>

Decrement

</button>

<button

className="button"

onClick={reset}

style={{ backgroundColor: "gray" }}

>

Reset

</button>

<button

className="button"

onClick={increment}

style={{ backgroundColor: "gray" }}

>

Increment

</button>

</div>

</div>

);

}

export default App;

* Output:

