**Objectives**

**Define JSX**

JSX (JavaScript XML) is a syntax extension for JavaScript used in React to write HTML-like code within JavaScript files.

🔹 It helps you build UI components more intuitively.

Example:

const greeting = <h1>Hello, JSX!</h1>;

**Explain about ECMA Script**

ECMAScript (ES) is the standard specification on which JavaScript is based.

🔹 It defines how JavaScript should behave.  
🔹 Versions like **ES5, ES6 (ES2015), ES7** added features like:

* let, const
* Arrow functions (()=>)
* Classes, modules, promises
* Destructuring, template literals, etc.

**Explain React.createElement()**

React.createElement() is a method that creates a **React element** (or virtual DOM node).

**Syntax:**

React.createElement(type, props, ...children)

**Example:**

React.createElement('h1', { className: 'title' }, 'Hello World')

This is what JSX compiles to behind the scenes:

<h1 className="title">Hello World</h1>

**Explain how to create React nodes with JSX**

You create nodes (elements) using HTML-like JSX syntax.

**Example:**

const element = <h1>Welcome to React</h1>;

const nested = (

<div>

<h2>Title</h2>

<p>Paragraph text</p>

</div>

);

These elements are just **React nodes** representing UI components.

**Define how to render JSX to DOM**

You use ReactDOM.render() to render JSX into a specific DOM node.

**Example:**

import React from 'react';

import ReactDOM from 'react-dom';

const element = <h1>Hello React</h1>;

ReactDOM.render(element, document.getElementById('root'));

**Explain how to use JavaScript expressions in JSX**

In JSX, you can embed JavaScript expressions using curly braces {}.

**Example:**

const name = "Kratika";

const element = <h1>Hello, {name}!</h1>;

You can also include expressions:

<p>2 + 2 = {2 + 2}</p>

**Explain how to use inline CSS in JSX**

Inline styles in JSX are written as **JavaScript objects**, not strings.

**Example:**

const styleObj = {

color: 'blue',

fontSize: '20px'

};

const element = <h1 style={styleObj}>Styled Text</h1>;

Or inline directly:

<h1 style={{ color: 'red', backgroundColor: 'yellow' }}>Inline Styled</h1>

**Code :**

**--App.css**

body {

  font-family: Arial, sans-serif;

  margin: 20px;

  background-color: #fff;

  color: #000;

}

.container {

  text-align: center;

}

h1 {

  font-size: 28px;

  font-weight: bold;

  margin-bottom: 20px;

}

img {

  margin: 20px 0;

  border-radius: 8px;

  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

}

h2 {

  font-size: 24px;

  font-weight: bold;

}

h3 {

  font-size: 20px;

  margin: 10px 0;

}

.textRed {

  color: red;

}

.textGreen {

  color: green;

}

.office-card {

  margin: 40px auto;

  padding: 20px;

  border: 1px solid #ccc;

  width: 50%;

  border-radius: 10px;

  box-shadow: 0 0 10px rgba(0, 0, 0, 0.15);

}

**--App.js**

import './App.css';

import OfficeList from './OfficeList';

function App() {

  return (

    <div className="App">

      <h1>Office Space , at Affordable Range</h1>

      <OfficeList />

    </div>

  );

}

export default App;

**--index.js**

import React from 'react';

import ReactDOM from 'react-dom/client';

import './index.css';

import App from './App';

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(

  <React.StrictMode>

    <App />j

  </React.StrictMode>

);

**--OfficeList.js**

import React from 'react';

import './App.css';

import officeImg from './office.jpg';

function OfficeList() {

  const offices = [

    { Name: 'DBS', Rent: 50000, Address: 'Chennai' },

    { Name: 'Regus', Rent: 70000, Address: 'Mumbai' },

    { Name: 'Smartworks', Rent: 45000, Address: 'Pune' }

  ];

  return (

    <div>

      {offices.map((item, index) => {

        const rentColor = item.Rent <= 60000 ? 'textRed' : 'textGreen';

        return (

          <div key={index} className="office-box">

            <img src={officeImg} width="25%" height="25%" alt="Office Space" />

            <h2>Name: <b>{item.Name}</b></h2>

            <h3 className={rentColor}>Rent: Rs. {item.Rent}</h3>

            <h3>Address: {item.Address}</h3>

          </div>

        );

      })}

    </div>

  );

}

export default OfficeList;

**Output :**

