**10. ReactJs – HOL**

**1. Define JSX**

* JSX stands for **JavaScript XML**, a syntax extension for JavaScript.
* It allows writing HTML-like code inside JavaScript.
* JSX gets transpiled to React.createElement() calls by tools like Babel.

**2. Explain about ECMA Script**

* ECMAScript is the **standardized scripting language** behind JavaScript.
* ES versions (like ES6) introduce new features to improve the language.
* React commonly uses ES6+ features such as arrow functions, classes, and modules.

**3. Explain React.createElement()**

* React.createElement() is used to create React elements without JSX.
* Syntax: React.createElement(type, props, children).
* JSX is syntactic sugar for React.createElement().

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

* Write HTML-like syntax inside JavaScript: <h1>Hello</h1>.
* Wrap multiple elements in a single parent (like <div> or <>).
* JSX elements are stored in variables and used in rendering.
* Define how to render JSX to DOM
* Use ReactDOM.render() to render JSX into the real DOM.
* Syntax: ReactDOM.render(<App />, document.getElementById('root')).
* The target DOM element must exist in the HTML file.

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

* JavaScript expressions can be embedded inside { } in JSX.
* Examples include {2 + 2}, {name}, or {items.map()}.
* Only expressions (not statements like if) are allowed.

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

* Use the style attribute with a JavaScript object: { color: 'red' }.
* Style property names use camelCase: backgroundColor, not background-color.
* Example: <h1 style={{ color: 'blue' }}>Title</h1>.

**React Application – officespacerentalapp**

Create a React Application named “officespacerentalapp” which uses React JSX to create elements, attributes and renders DOM to display the page.

Create an element to display the heading of the page.

Attribute to display the image of the office space

Create an object of office to display the details like Name, Rent and Address.

Create a list of Object and loop through the office space item to display more data.

To apply Css, Display the color of the Rent in Red if it’s below 60000 and in Green if it’s above 60000.





**Output:**

