React JS

React JS is an open source JavaScript Front End library used to develop User Interface.  
React JS is used to handle the view layer for web and Mobile applications. React JS allows us to create reusable UI components. React JS is concerned with the components that utilizes the expressiveness of JavaScript along with a HTML –like template syntax

React uses JSX for templating instead of regular JavaScript  
1.It is faster because it performance optimization while compiling code to JavaScript  
2.It is also a type-safe and most of the errors can be caught during compilation  
3.If we are familiar with HTML, it makes easier and faster to write templates

In React JS, the render function can render only one HTML element at a time  
**Nested Elements**  
In case we want to render multiple HTML elements then we need wrap it with one container element like <div>  
**Attributes**  
We can create our own custom attributes in addition to regular HTML properties and attributes using data-prefixes   
**Expressions**  
JavaScript expressions can be used inside the JSX. We just need to wrap it with curly brackets{}. We cannot use if-else condition inside JSX, instead we can use conditional (ternary) expressions  
**Ex:** Here variable **i** is equal to 1 then the browser will render true, if the **i** value is changed some other value the browser will render false

import React from 'react';

class App extends React.Component {

render() {

var i = 1;

return (

<div>

<h1>{i == 1 ? 'True!' : 'False'}</h1>

</div>  
 );  
 }  
 }

export default App;

**Styling:**React uses inline styling.  
When we want to set inline styles, we need to use camelCase syntax. React will automatically append px after the number value on specific elements

**React JS has 3 main aspects  
1.Virtual DOM,   
2.Data binding (It has one way data binding) and   
3. It provides the Server Side Rendering**

**DOM: (Document Object Model)**When a web page is loaded, the browser creates a Document Object Model of the page**.**JavaScript can access and change all the elements of an HTML documentit dynamically add or remove elements from a web page at the back end this lead to more memory consumption which ultimately made our application slow

**Virtual DOM**Like the actual DOM, the virtual DOM is a node tree that lists like elements and their attributes and content as objects and properties. React render() function creates a node tree from React components and updates this tree in response to mutations in the data model, caused by actions  
Each time the underlying data changes in a React applications, a new Virtual DOM representation of the user interface is created  
Updating the browser’s DOM is a three step process in React  
1. Whenever anything may have changed, the entire UI will be re-rendered in a virtual DOM representation.  
2. The difference between the previous Virtual DOM representation and the new one will be calculated.  
3. The real DOM will be updated with what has actually changed. This is like applying a patch

**Advantages of React JS**1. Using virtual DOM the performance of application is increased  
2. React JS used on client as well as server side **3.** Readability is improved, using readability it is easy to debug and maintain  
4. React JS can be easily integrated with other frameworks like AngularJS

**Components:**Using react our entire application is divide into various components. We take like one component as a parent component and the other remaining components are child components. This approach allows us to update and change our components without reloading the entire page. React uses a virtual DOM instead of regular DOM which makes our applications lighter and faster  
**Components in React JS can be in two forms**  
Stateful: Remembers everything it does  
Stateless: Doesn’t remember anything it does

**Props**Props are immutable**.** This is why the container component should define the state that can be updated and changed, while the child component should only pass data from the state using props