### What Is React?



React is a JavaScript-based UI development library. Facebook and an open-source developer community run it. Although React is a library rather than a language, it is widely used in web development. The library first appeared in May 2013 and is now one of the most commonly used frontend libraries for web development.

React offers various extensions for entire application architectural support, such as Flux and React Native, beyond mere UI.

Why React?

React’s popularity today has eclipsed that of all other front-end development frameworks. Here is why:

* Easy creation of dynamic applications: React makes it easier to create dynamic web applications because it requires less coding and offers more functionality, as opposed to JavaScript, where coding often gets complex very quickly.
* Improved performance: React uses Virtual DOM, thereby creating web applications faster. Virtual DOM compares the components’ previous states and updates only the items in the Real DOM that were changed, instead of updating all of the components again, as conventional web applications do.
* Reusable components: Components are the building blocks of any React application, and a single app usually consists of multiple components. These components have their logic and controls, and they can be reused throughout the application, which in turn dramatically reduces the application’s development time.
* Unidirectional data flow: React follows a unidirectional data flow. This means that when designing a React app, developers often nest child components within parent components. Since the data flows in a single direction, it becomes easier to debug errors and know where a problem occurs in an application at the moment in question.
* Small learning curve: React is easy to learn, as it mostly combines basic [HTML](https://www.simplilearn.com/tutorials/html-tutorial/what-is-html) and JavaScript concepts with some beneficial additions. Still, as is the case with other tools and frameworks, you have to spend some time to get a proper understanding of React’s library.
* It can be used for the development of both web and mobile apps: We already know that React is used for the development of web applications, but that’s not all it can do. There is a framework called React Native, derived from React itself, that is hugely popular and is used for creating beautiful mobile applications. So, in reality, React can be used for making both web and mobile applications.
* Dedicated tools for easy debugging: Facebook has released a Chrome extension that can be used to debug React applications. This makes the process of debugging React web applications faster and easier.

The above reasons more than justify the popularity of the React library and why it is being adopted by a large number of organizations and businesses. Now let’s familiarize ourselves with React’s features.

## Features of React

React offers some outstanding features that make it the most widely adopted library for frontend app development. Here is the list of those salient features.

### **JSX**



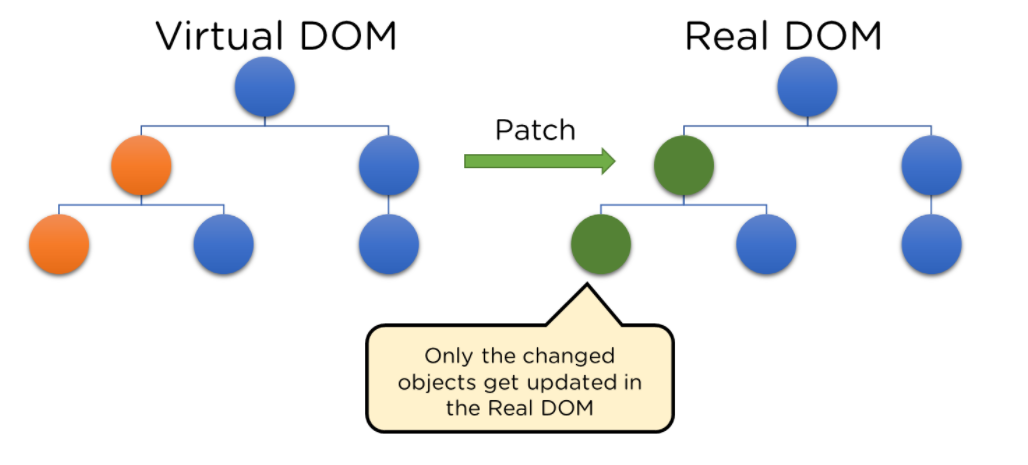
JSX is a JavaScript syntactic extension. It's a term used in React to describe how the user interface should seem. You can write HTML structures in the same file as JavaScript code by utilizing JSX.

const name = 'Welcome to React Training';

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

The above code shows how JSX is implemented in React. It is neither a string nor HTML. Instead, it embeds HTML into JavaScript code.

### **Virtual Document Object Model (DOM)**



The Virtual DOM is React's lightweight version of the Real DOM. Real DOM manipulation is substantially slower than virtual DOM manipulation. When an object's state changes, Virtual DOM updates only that object in the real DOM rather than all of them.

* What is the Document Object Model (DOM)?

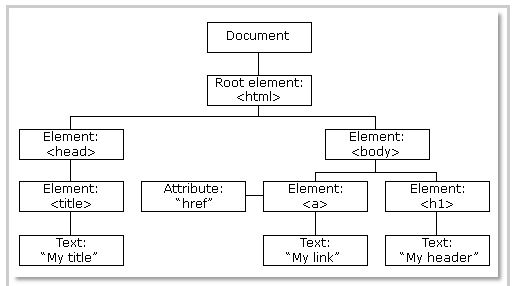


Fig: DOM of a Webpage

DOM (Document Object Model) treats an XML or HTML document as a tree structure in which each node is an object representing a part of the document.

* How do Virtual DOM and React DOM interact with each other?

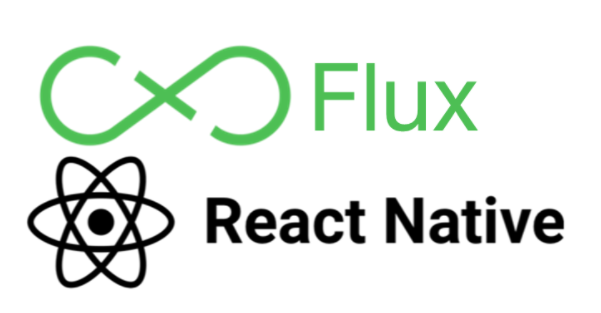
When the state of an object changes in a React application, VDOM gets updated. It then compares its previous state and then updates only those objects in the real DOM instead of updating all of the objects. This makes things move fast, especially when compared to other front-end technologies that have to update each object even if only a single object changes in the web application.

### **Architecture**

In a **Model View Controller(MVC) architecture**, React is the 'View' responsible for how the app looks and feels.

MVC is an architectural pattern that splits the application layer into Model, View, and Controller. The model relates to all data-related logic; the view is used for the UI logic of the application, and the controller is an interface between the Model and View.

### **Extensions**



React goes beyond just being a UI framework; it contains many extensions that cover the entire application architecture. It helps the building of mobile apps and provides server-side rendering. Flux and Redux, among other things, can extend React.

### **Data Binding**

Since React employs one-way data binding, all activities stay modular and quick. Moreover, the unidirectional data flow means that it's common to nest child components within parent components when developing a React project.



Fig: One-way data binding

### **Debugging**

Since a broad developer community exists, React applications are straightforward and easy to test. Facebook provides a browser extension that simplifies and expedites React debugging.

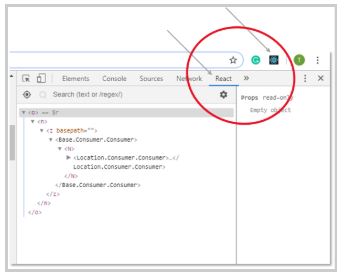


Fig: React Extension

This extension, for example, adds a React tab in the developer tools option within the Chrome web browser. The tab makes it easy to inspect React components directly.