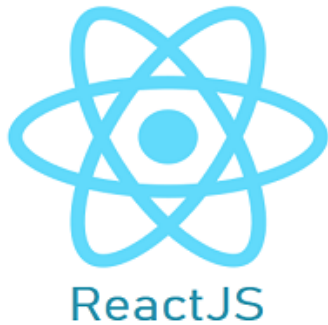


ReactJS Introduction



ReactJS is a **declarative, efficient,** and flexible **JavaScript library** for building reusable UI components. It is an open-source, component-based front end library which is responsible only for the view layer of the application. It was initially developed and maintained by Facebook and later used in its products like WhatsApp & Instagram.

Currently, ReactJS is one of the most popular JavaScript front-end libraries which has a strong foundation and a large community.

We includes all the topics which help to learn ReactJS. These are ReactJS Introduction, ReactJS Features, ReactJS Installation, Pros and Cons of ReactJS, ReactJS JSX, ReactJS Components, ReactJS State, ReactJS Props, ReactJS Forms, ReactJS Events, ReactJS Animation and many more.

Why we use ReactJS?

The main objective of ReactJS is to develop User Interfaces (UI) that improves the speed of the apps. It uses virtual DOM (JavaScript object), which improves the performance of the app. The JavaScript virtual DOM is faster than the regular DOM. We can use ReactJS on the client and server-side as well as with other frameworks. It uses component and data patterns that improve readability and helps to maintain larger apps.

React Introduction

ReactJS is a declarative, efficient, and flexible JavaScript library for building reusable UI components. It is an open-source, component-based front end library responsible only for the view layer of the application.

It was created by **Jordan Walke**, who was a software engineer at **Facebook**. It was initially developed and maintained by Facebook and was later used in its products like **WhatsApp** & **Instagram**. Facebook developed ReactJS in **2011** in its newsfeed section, but it was released to the public in the month of **May 2013**.

Today, most of the websites are built using MVC (model view controller) architecture. In MVC architecture, React is the 'V' which stands for view, whereas the architecture is provided by the Redux or Flux.

A ReactJS application is made up of multiple components, each component responsible for outputting a small, reusable piece of HTML code. The components are the heart of all React applications. These Components can be nested with other components to allow complex applications to be built of simple building blocks. ReactJS uses virtual DOM based mechanism to fill data in HTML DOM. The virtual DOM works fast as it only changes individual DOM elements instead of reloading complete DOM every time.

To create React app, we write React components that correspond to various elements. We organize these components inside higher level components which define the application structure. For example, we take a form that consists of many elements like input fields, labels, or buttons. We can write each element of the form as React components, and then we combine it into a higher-level component, i.e., the form component itself. The form components would specify the structure of the form along with elements inside of it.

Why learn ReactJS?

Today, many JavaScript frameworks are available in the market(like angular, node), but still, React came into the market and gained popularity amongst them. The previous frameworks follow the traditional data flow structure, which uses the DOM (Document Object Model). DOM is an object which is created by the browser each time a web page is loaded. It dynamically adds or removes the data at the back end and when any modifications were done, then each time a new DOM is created for the same page. This repeated creation of DOM makes unnecessary memory wastage and reduces the performance of the application.

Therefore, a new technology ReactJS framework invented which remove this drawback. ReactJS allows you to divide your entire application into various components. ReactJS still used the same traditional data flow, but it is not directly operating on the browser's Document Object Model (DOM) immediately; instead, it operates on a virtual DOM. It means rather than manipulating the document in a browser after changes to our data, it resolves changes on a DOM built and run entirely in memory. After the virtual DOM has been updated, React determines what changes made to the actual browser's DOM. The React Virtual DOM exists entirely in memory and is a representation of the web browser's DOM. Due to this, when we write a React component, we did not write directly to the DOM; instead, we are writing virtual components that react will turn into the DOM.

React Version

A complete release history for React is given below. You can also see the full documentation for recent releases on GitHub.

SN	Version	Release Date	Significant Changes
1.	0.3.0	29/05/2013	Initial Public Release
2.	0.4.0	20/07/2013	Support for comment nodes <code><div>{/* */</div></code> , Improved server-side rendering APIs, Removed <code>React.autoBind</code> , Support for the <code>key</code> prop, Improvements to forms, Fixed bugs.
3.	0.5.0	20/10/2013	Improve Memory usage, Support for Selection and Composition events, Support for <code>getInitialState</code> and <code>getDefaultProps</code> in mixins, Added <code>React.version</code> and <code>React.isValidClass</code> , Improved compatibility for Windows.
4.	0.8.0	20/12/2013	Added support for <code>rows</code> & <code>cols</code> , <code>defer</code> & <code>async</code> , loop for <code><audio></code> & <code><video></code> , <code>autoCorrect</code> attributes. Added <code>onContextMenu</code> events, Upgraded <code>jstransform</code> and <code>esprima-fb</code> tools, Upgraded <code>browserify</code> .
5.	0.9.0	20/02/2014	Added support for <code>crossOrigin</code> , <code>download</code> and <code>hrefLang</code> , <code>mediaGroup</code> and <code>muted</code> , <code>sandbox</code> , <code>seamless</code> , and <code>srcDoc</code> , <code>scope</code> attributes, Added <code>any</code> , <code>arrayOf</code> , <code>component</code> , <code>oneOfType</code> , <code>renderable</code> , <code>shape</code> to <code>React.PropTypes</code> , Added support for <code>onMouseOver</code> and <code>onMouseOut</code> event, Added support for <code>onLoad</code> and <code>onError</code> on <code></code> elements.
6.	0.10.0	21-03-2014	Added support for <code>srcSet</code> and <code>textAnchor</code> attributes, add update function for immutable data, Ensure all void elements don't insert a closing tag.
7.	0.11.0	17/07/2014	Improved SVG support, Normalized <code>e.view</code> event, Update <code>\$apply</code> command, Added support for namespaces, Added new <code>transformWithDetails</code> API, includes pre-built packages under <code>dist/</code> , <code>MyComponent()</code> now returns a descriptor, not an instance.
8.	0.12.0	21/11/2014	Added new features Spread operator (<code>{...}</code>) introduced to deprecate <code>this.transferPropsTo</code> , Added support for <code>acceptCharset</code> , <code>classID</code> , <code>manifest</code> HTML attributes, <code>React.addons.batchedUpdates</code> added to API, <code>@jsx</code>

			React.DOM no longer required, Fixed issues with CSS Transitions.
9.	0.13.0	10/03/2015	Deprecated patterns that warned in 0.12 no longer work, ref resolution order has changed, Removed properties this._pendingState and this._rootNodeID, Support ES6 classes, Added API React.findDOMNode(component), Support for iterators and immutable-js sequences, Added new features React.addons.createFragment, deprecated React.addons.classSet.
10.	0.14.1	29/10/2015	Added support for srcLang, default, kind attributes, and color attribute, Ensured legacy .props access on DOM nodes, Fixed scryRenderedDOMComponentsWithClass, Added react-dom.js.
11.	15.0.0	07/04/2016	Initial render now uses document.createElement instead of generating HTML, No more extra s, Improved SVG support, ReactPerf.getLastMeasurements() is opaque, New deprecations introduced with a warning, Fixed multiple small memory leaks, React DOM now supports the cite and profile HTML attributes and cssFloat, gridRow and gridColumn CSS properties.
12.	15.1.0	20/05/2016	Fix a batching bug, Ensure use of the latest object-assign, Fix regression, Remove use of merge utility, Renamed some modules.
13.	15.2.0	01/07/2016	Include component stack information, Stop validating props at mount time, Add React.PropTypes.symbol, Add onLoad handling to <link> and onError handling to <source> element, Add isRunning() API, Fix performance regression.
14.	15.3.0	30/07/2016	Add React.PureComponent, Fix issue with nested server rendering, Add xmlns, xmlnsXlink to support SVG attributes and referrerPolicy to HTML attributes, updates React Perf Add-on, Fixed issue with ref.
15.	15.3.1	19/08/2016	Improve performance of development builds, Cleanup internal hooks, Upgrade fbjs, Improve startup time of

			React, Fix memory leak in server rendering, fix React Test Renderer, Change trackedTouchCount invariant into a console.error.
16.	15.4.0	16/11/2016	React package and browser build no longer includes React DOM, Improved development performance, Fixed occasional test failures, update batchedUpdates API, React Perf, and ReactTestRenderer.create().
17.	15.4.1	23/11/2016	Restructure variable assignment, Fixed event handling, Fixed compatibility of browser build with AMD environments.
18.	15.4.2	06/01/2017	Fixed build issues, Added missing package dependencies, Improved error messages.
19.	15.5.0	07/04/2017	Added react-dom/test-utils, Removed peerDependencies, Fixed issue with Closure Compiler, Added a deprecation warning for React.createClass and React.PropTypes, Fixed Chrome bug.
20.	15.5.4	11/04/2017	Fix compatibility with Enzyme by exposing batchedUpdates on shallow renderer, Update version of prop-types, Fix react-addons-create-fragment package to include loose-envify transform.
21.	15.6.0	13/06/2017	Add support for CSS variables in style attribute and Grid style properties, Fix AMD support for addons depending on react, Remove unnecessary dependency, Add a deprecation warning for React.createClass and React.DOM factory helpers.
22.	16.0.0	26/09/2017	Improvcd error handling with introduction of "error boundaries", React DOM allows passing non-standard attributes, Minor changes to setState behavior, remove react-with-addons.js build, Add React.createClass as create-react-class, React.PropTypes as prop-types, React.DOM as react-dom-factories, changes to the behavior of scheduling and lifecycle methods.
23.	16.1.0	9/11/2017	Discontinuing Bower Releases, Fix an accidental extra global variable in the UMD builds, Fix onMouseEnter and onMouseLeave firing, Fix <textarea> placeholder,

			Remove unused code, Add a missing package.json dependency, Add support for React DevTools.
24.	16.3.0	29/03/2018	Add a new officially supported context API, Add new packagePrevent an infinite loop when attempting to render portals with SSR, Fix an issue with this.state, Fix an IE/Edge issue.
25.	16.3.1	03/04/2018	Prefix private API, Fix performance regression and error handling bugs in development mode, Add peer dependency, Fix a false positive warning in IE11 when using Fragment.
26.	16.3.2	16/04/2018	Fix an IE crash, Fix labels in User Timing measurements, Add a UMD build, Improve performance of unstable_observedBits API with nesting.
27.	16.4.0	24/05/2018	Add support for Pointer Events specification, Add the ability to specify propTypes, Fix reading context, Fix the getDerivedStateFromProps() support, Fix a testInstance.parent crash, Add React.unstable_Profiler component for measuring performance, Change internal event names.
28.	16.5.0	05/09/2018	Add support for React DevTools Profiler, Handle errors in more edge cases gracefully, Add react-dom/profiling, Add onAuxClick event for browsers, Add movementX and movementY fields to mouse events, Add tangentialPressure and twist fields to pointer event.
29.	16.6.0	23/10/2018	Add support for contextType, Support priority levels, continuations, and wrapped callbacks, Improve the fallback mechanism, Fix gray overlay on iOS Safari, Add React.lazy() for code splitting components.
30.	16.7.0	20/12/2018	Fix performance of React.lazy for lazily-loaded components, Clear fields on unmount to avoid memory leaks, Fix bug with SSR, Fix a performance regression.
31.	16.8.0	06/02/2019	Add Hooks, Add ReactTestRenderer.act() and ReactTestUtils.act() for batching updates, Support

			synchronous thenables passed to React.lazy(), Improve useReducer Hook lazy initialization API.
32.	16.8.6	27/03/2019	Fix an incorrect bailout in useReducer(), Fix iframe warnings in Safari DevTools, Warn if contextType is set to Context.Consumer instead of Context, Warn if contextType is set to invalid values.