REACT JS

Introduction

Sreekanth M. E.

Freelance Trainer & Consultant

http://www.SreekanthME.com

ReactJS (aka React.js or React) is an User Interface library for writing SPA.

It is an open-source JavaScript library.

It is not a full-fledged framework that has an opinion on how everything in your app should behave.

React works in the View layer and is concerned mainly about visual elements and keeping them up-to-date.

React library corresponds to just the V in an MVC (Model-View-Controller) Architecture.

Typically, React SPA makes use of Redux and React-Router. Together they behave like a full-fledged framework.

React was created by Jordan Walke, a software engineer at Facebook.

It was open-sourced at JSConf US in May 2013.

It is maintained by Facebook, Instagram and a community of individual developers and corporations

Current version of react is 16.x

Manipulating the HTML DOM is very slow.

Manually querying elements, adding children, removing subtrees, and performing other DOM operations in the browser are time consuming tasks.

In a SPA, usually there is a lot of DOM manipulation involved.

But React uses something called Virtual DOM instead of the Real DOM which overcomes this performance issue.

Manipulating the Virtual DOM is extremely fast.

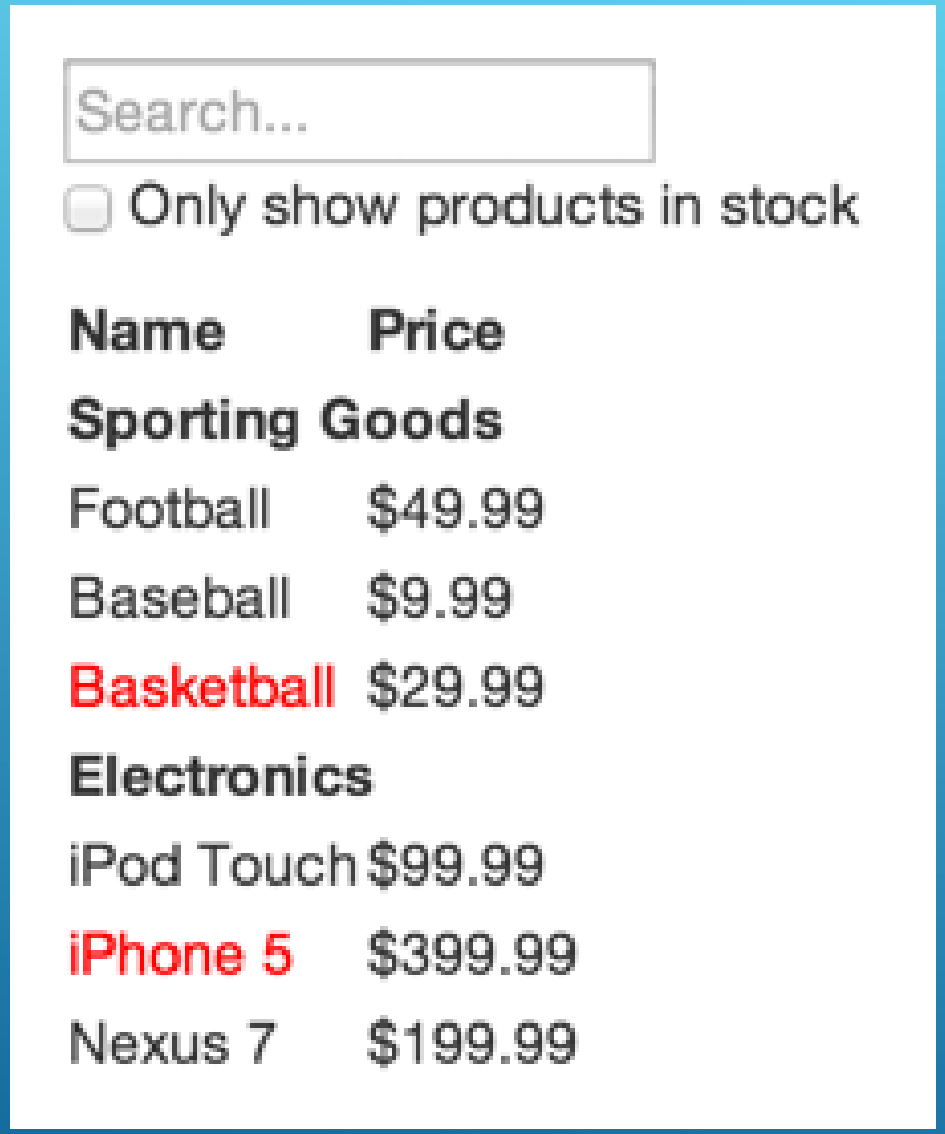
React compares changes between Virtual DOM and Real DOM, figuring out which changes actually matter, and makes the least amount of DOM changes needed to keep everything up-to-date in a process called reconciliation.

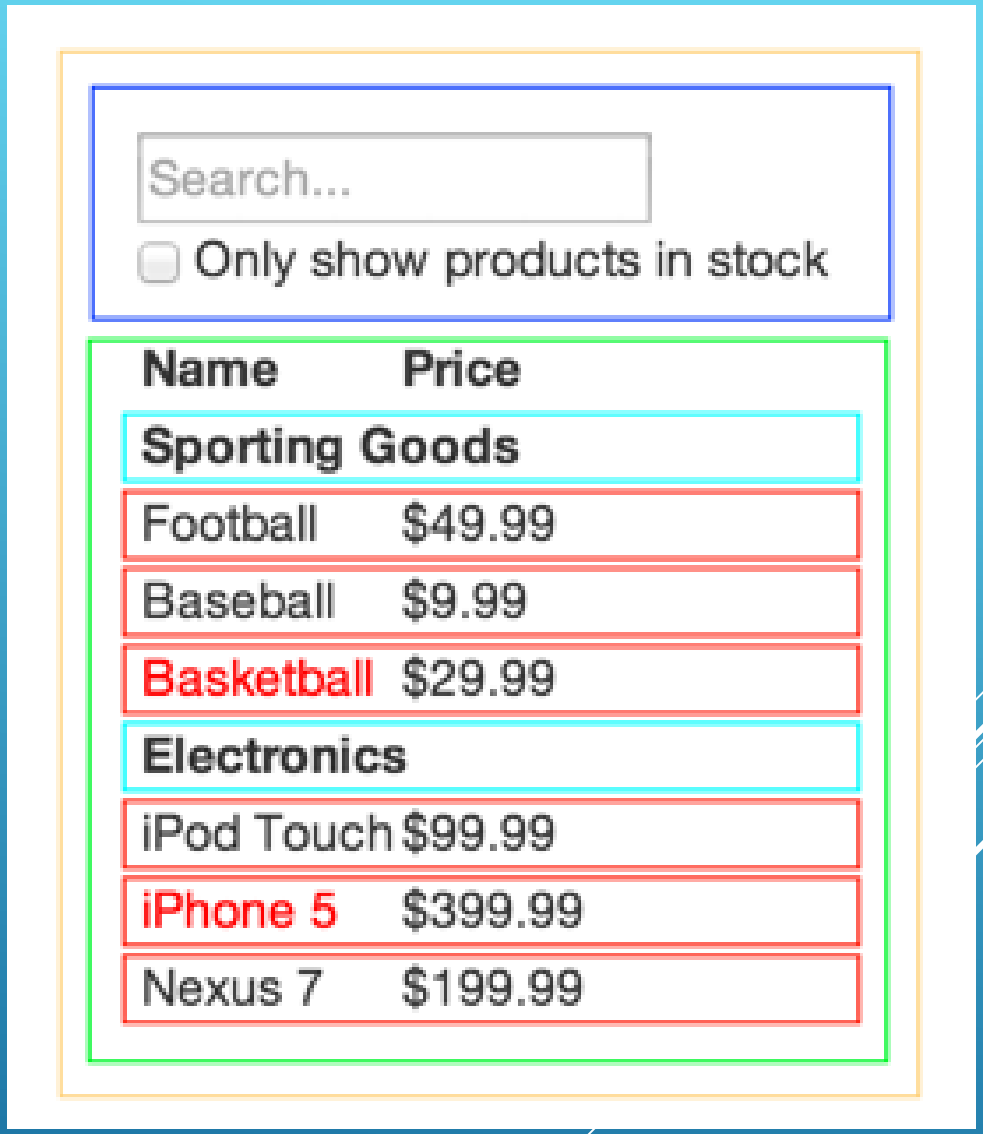
A React SPA is built out of building blocks called Components.

Instead of treating the visual elements in the app as one monolithic chunk, React encourages break up of visual elements into smaller and smaller components

Just like everything else in programming, it is a good idea to have things be modular, compact, and self-contained.

Many of React's core APIs revolve around making it easier to create smaller visual components that can later be composed with other visual components to make larger and more complex visual components.





In React, all visual elements are defined ONLY in JavaScript.

No need for HTML. Even CSS can be written as JavaScript.

It does not mix JavaScript inside HTML templates like some other SPA frameworks do.

In React, visual elements and JavaScript live in the same location.

No need to jump between multiple files to define the look and behavior of one visual component.

React code can be written using pure JavaScript

OR

using a HTML-like syntax called JSX.

Browsers cannot understand JSX or ES6/7.

So React takes the help of Babel to transpile JSX & ES6/7 to plain JavaScript.

React libraries come packaged as Node.js modules.

Babel is used as a dev-dependency to create a production build of ReactJS code.

It is not needed in PROD.

However, for learning purposes, React is also available in standalone mode.

Below are the CDN links for use in browser:

https://unpkg.com/react@16.0.0/umd/react.production.min.js

https://unpkg.com/react-dom@16.0.0/umd/react-dom.production.min.js

When writing ReactJS code inside HTML file, code must be inside SCRIPT tags of type text/babel.

RENDER() METHOD

ReactDOM.render() is used to load React code into HTML DOM.

The render method takes two arguments:

- HTML-like elements (aka JSX) to output
- The DOM node into which React will render the JSX

Applications built with React usually have a single root DOM node. But rendering into multiple DOM nodes is not disallowed.

If integrating React into an existing app, there can be as many isolated root DOM nodes as required.

END OF CHAPTER

APPENDIX

UNPKG CDN

https://unpkg.com/

unpkg is a fast, global content delivery network for everything on npm. Use it to quickly and easily load files using a simple URL like:

https://unpkg.com/package@version/file