



ReactJS

What is React?

- JavaScript library created by Facebook**
- Also used by Netflix & Instagram**
- Used to create JS-driven dynamic web apps**
- In that regard, can be compared to Angular & Vue**

What is Redux?

- A layer on-top of React
- Helps with state management of our app
 - data in the app
 - UI state of the app

Code editing. Redefined.

Free. Open source. Runs everywhere.

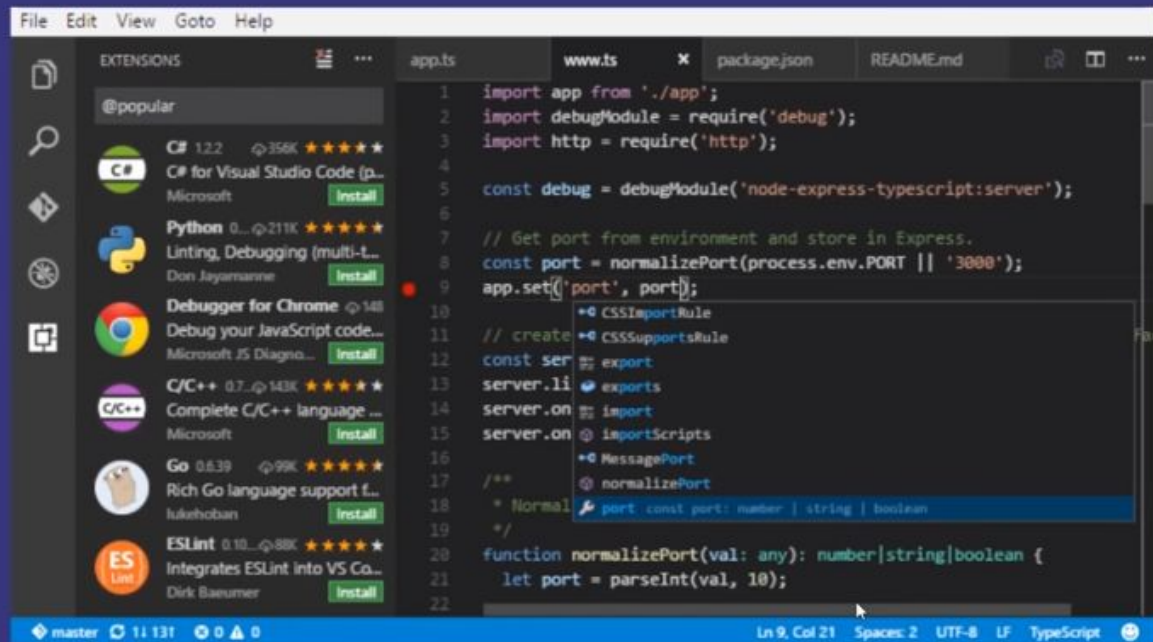
Download for Windows

Stable Build



Other platforms and Insiders Edition

By using VS Code, you agree to its
license and privacy statement.



Babel is a JavaScript compiler.

Use next generation JavaScript, today.

Babel 7 is out! Please read our [announcement](#) and [upgrade guide](#) for more information.

Put in next-gen JavaScript

```
[1, 2, 3].map(n => n ** 2);
```

Get browser-compatible JavaScript out

```
[1, 2, 3].map(function (n) {  
  return Math.pow(n, 2);  
});
```

Special Sponsor



Decentralized certificate authority and naming

JSX - example

<https://babeljs.io/repl/build/7.0/>

```
<div className="App">  
  <header className="App-header">  
    <h1>Hello World!</h1>  
  </header>  
</div>
```

Transpile.....

React without NPM and Babel...?

Yes, it can be done....

But probably less convenient....

For example: we will have to translate JSX into JS ourselves...

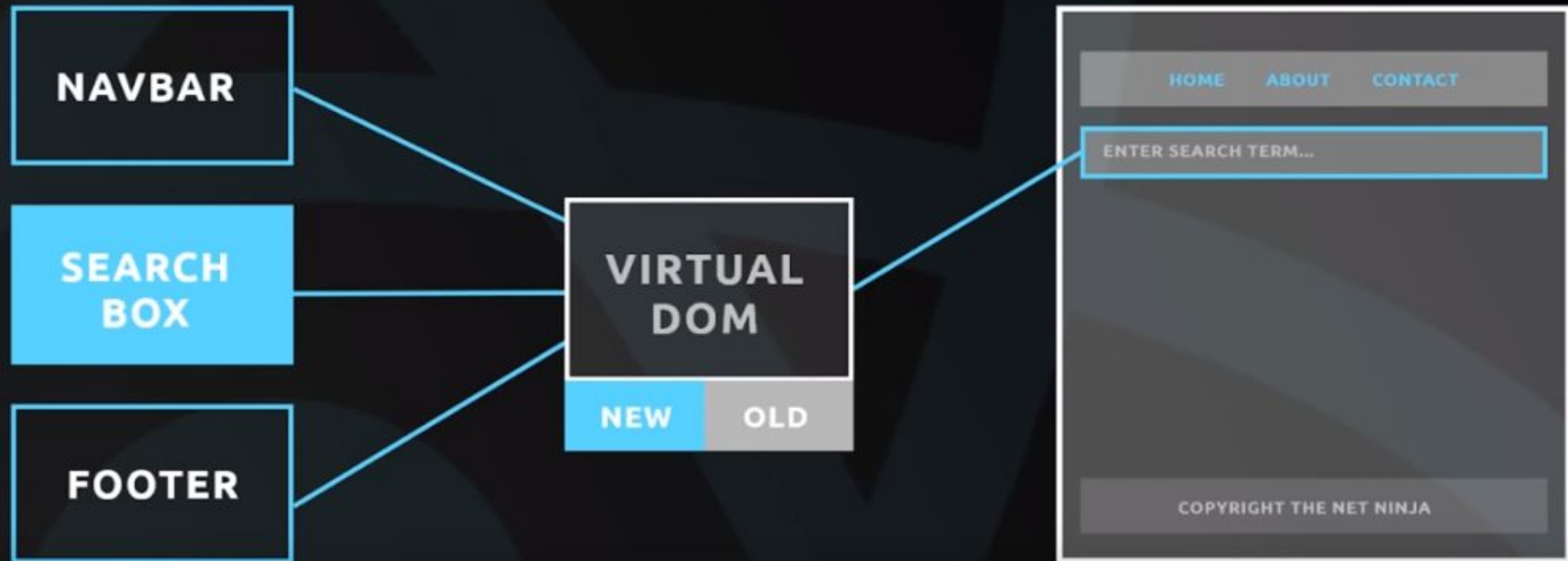
Read more about it:

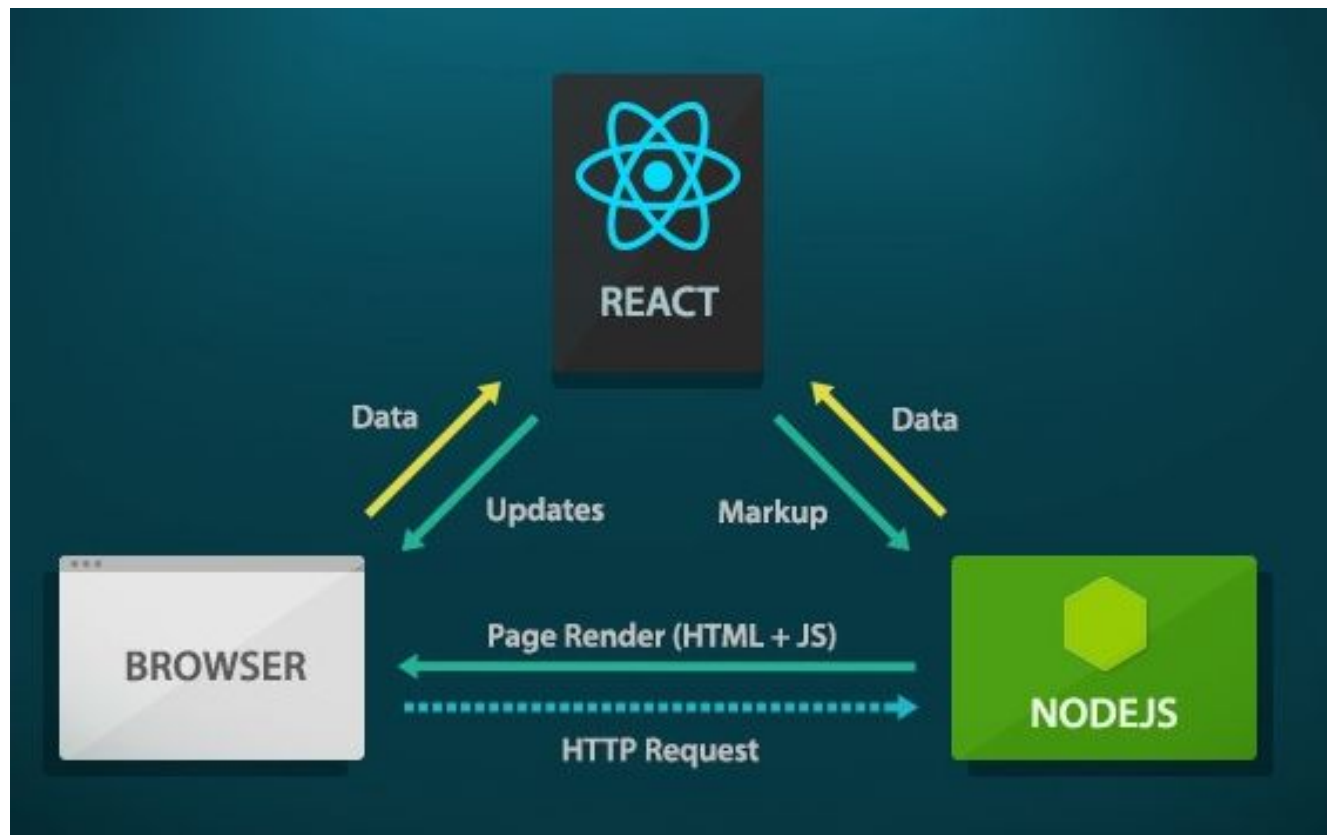
<https://medium.com/@chrislewisdev/react-without-npm-babel-or-webpack-1e9a6049714>



The Virtual DOM

- The Virtual DOM makes React fast...





Components & Templates

- Components look like HTML templates (actually JSX)
- They can contain 'state' (data or UI state)
- They also can contain JavaScript for functionality

```
class App extends React.Component {  
  state = {  
    name: 'Ryu',  
    age: 30  
  }  
  render(){  
    return(  
      <div className="app-content">  
        <h1>Hello, ninjas!</h1>  
        <p>My name is: { this.state.name } and I am { this.state.age }</p>  
      </div>  
    )  
  }  
}
```

START CODING !



Search Extensions in Marketplace

ENABLED 6

ES7 React/Redux/GraphQL...

1.8.5

Simple extensions for React, Red...

dsznajder

Live Server

5.1.1

Launch a development local Ser...

Ritwick Dey

Material Icon Theme

3.5.2

Material Design Icons for Visual...

Philipp Kief

Monokai++

1.6.7

A modern Monokai theme for S...

Davide Casella

Sublime Babel

0.2.10

Sublime Text's babel-sublime gr...


Josh Peng

Vetur

0.12.5

Vue tooling for VS Code

Pine Wu



ES7 React/Redux/GraphQL/React-Native snippets

dsznajder.es7-react-js-snippets

dsznajder | 441,810 | ★★★★★ | Repository

Simple extensions for React, Redux and GraphQL in JS/TS with ES7 syntax

Disable Uninstall

Details

Contributions

Changelog

Dependencies

VS Code ES7 React/Redux/React-Native/JS snippets

Visual Studio Marketplace v1.8.5 installs 441810 rating 5/5 (7)

This extension provide you Javascript and React/Redux snippets in ES7 with babel plugins features for [Vs Code](#)

Here is direct link to marketplace [ES7 React/Redux/React-Native/JS Snippets](#)

Supported languages (file extensions)

- JavaScript (.js)
- JavaScript React (.jsx)
- TypeScript (.ts)
- TypeScript React (.tsx)

Search Extensions in Marketplace

ENABLED 6

ES7 React/Redux/GraphQL...

1.8.5

Simple extensions for React, Red...

dsznajder

Live Server

5.1.1

Launch a development local Ser...

Ritwick Dey

Material Icon Theme

3.5.2

Material Design Icons for Visual...

Philipp Kief

Monokai++

1.6.7

A modern Monokai theme for S...

Davide Casella

Sublime Babel

0.2.10


Sublime Text's babel-sublime gr...

Josh Peng

Vetur

0.12.5

Vue tooling for VS Code



Sublime Babel

joshpeng.sublime-babel-vscode

Josh Peng | 153,448 | ★★★★★ | Repository | License

Sublime Text's babel-sublime grammar in VS Code.

Disable Uninstall

Details

Contributions

Changelog

Dependencies

Sublime Babel

installs 153450

When using your favorite theme, Visual Studio Code doesn't always display syntax highlight colors as expected. This extension tries to mimic Sublime Text's [babel-sublime](#) package as much as possible to address poor coloring.

With VS Code 1.9.0, a new token flattener will be implemented that finally allows VS Code to properly follow TextMate grammar rules. This significantly improves syntax highlighting with preexisting tmThemes from other editors. However, the default JavaScript grammar still needs some work to match Sublime Text. This extension solves that for you.

CDN Links

Both React and ReactDOM are available over a CDN.

```
<script crossorigin src="https://unpkg.com/react@16/umd/react.development.js"></script>  
<script crossorigin src="https://unpkg.com/react-dom@16/umd/react-dom.development.js"></script>
```

The versions above are only meant for development, and are not suitable for production.

Minified and optimized production versions of React are available at:

```
<script crossorigin src="https://unpkg.com/react@16/umd/react.production.min.js"></script>  
<script crossorigin src="https://unpkg.com/react-dom@16/umd/react-dom.production.min.js"></script>
```

To load a specific version of `react` and `react-dom`, replace `16` with the version number.

<https://reactjs.org/docs/cdn-links.html>

What's with the Cross-Origin?

*“Normal script elements pass minimal information to the `window.onerror` for scripts which do not pass the standard **CORS** checks...”*

“When a JavaScript runtime error (including syntax errors and exceptions thrown within handlers) occurs, an error event using interface `ErrorEvent` is fired at window and `window.onerror()` is invoked”



UNPKG

unpkg is a fast, global [content delivery network](#) for everything on [npm](#). Use it to quickly and easily load any file from any package using a URL like:

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

From **Mar 23** to **Apr 22** unpkg served **16,373,336,290** requests and a total of **214 TB** of data to **1,468,278,806** unique visitors, **99.14%** of which were served from the cache.



babel-standalone

babel-standalone is a standalone build of Babel for use in non-Node.js environments, including browsers. It's bundled with all the standard Babel plugins and presets, and [a build of babili \(babel-minify\)](#) is optionally available too.

But why?!

It's true that using Babel through Webpack, Browserify or Gulp should be sufficient for most use cases. However, there are some valid use cases for babel-standalone:

- Sites like [JSFiddle](#), [JS Bin](#), the [REPL on the Babel site](#), etc. These sites compile user-provided JavaScript in real-time.
 - Apps that embed a JavaScript engine such as V8 directly, and want to use Babel for compilation
 - Apps that want to use JavaScript as a scripting language for extending the app itself, including all the goodies that ES2015 provides.
 - Integration of Babel into a non-Node.js environment ([ReactJS.NET](#), [ruby-babel-transpiler](#), [php-babel-transpiler](#), etc).
- Use it via UNPKG: <https://unpkg.com/babel-standalone@6/babel.min.js>.

<https://github.com/babel/babel-standalone>


```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">

  <script src="https://unpkg.com/babel-standalone@6/babel.min.js"></script>
  <script crossorigin src="https://unpkg.com/react@16/umd/react.production.min.js"></script>
  <script crossorigin src="https://unpkg.com/react-dom@16/umd/react-dom.production.min.js"></script>
  <title>Document</title>
</head>
<body>
  <div id="app"></div>
  <script type="text/babel">
    class App extends React.Component {
      render(){
        return(
          <div className="app-content">
            <h1>Hello World!</h1>
            <p>Random number: { Math.random() * 10 }</p>
          </div>
        )
      }
    }
    ReactDOM.render(<App />, document.getElementById('app'));
  </script>
</body>
</html>
```



Live Server ritwickdey.liveserver

Ritwick Dey | 1,453,818 | ★★★★★ | Repository | License

Launch a development local Server with live reload feature for static & dynamic pages

Disable ▼ Uninstall

Details Contributions Changelog Dependencies

[I'm sorry but I'm super busy now. If you want to be a maintainer of the project, please feel free to contact me! You've to be passionate about programming]

Live Server

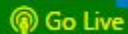
Live Server loves ❤️ your multi-root workspace

Live Server for server side pages like PHP. [Check Here](#)

[For 'command not found error' #78]

vscode marketplace v5.6.1 downloads 4M rating 4.8/5 (177)

travis branch failing appveyor branch passing license MIT



Ln 18, Col 18

Spaces: 2

UTF-8

CRLF

HTML



...Auto reloads on every save!!

Component State

- JavaScript Object
- Describes the current state of the component
 - data, UI-state
- The state of a component can be updated over time
 - a modal could close
 - the data we output could change



React Developer Tools

offered by Facebook

★★★★☆ (1020)

[Developer Tools](#)

1,117,882 users

ADDED TO CHROME

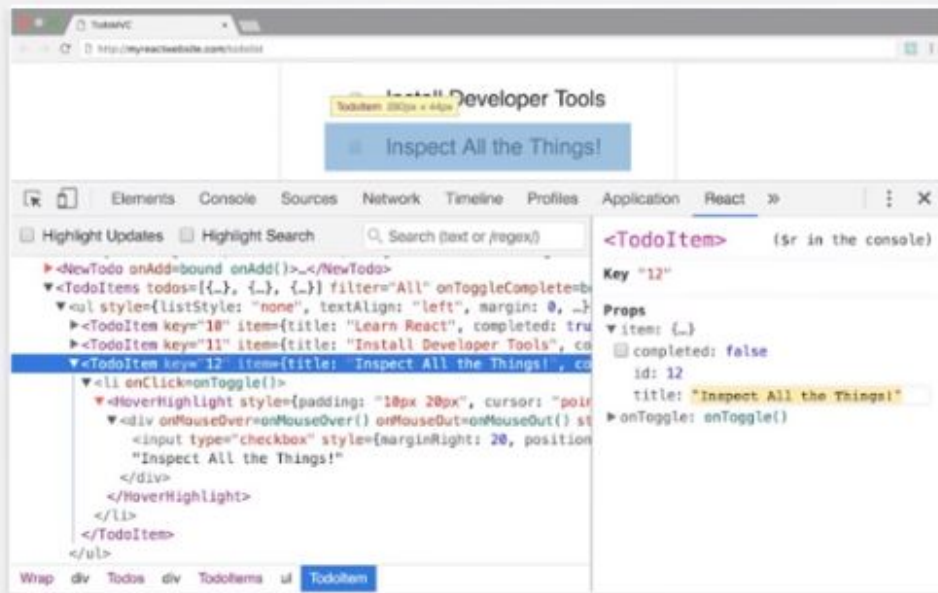


OVERVIEW

REVIEWS

SUPPORT

RELATED



Compatible with your device

Adds React debugging tools to the Chrome Developer Tools.

React Developer Tools is a Chrome DevTools extension for the open-source React JavaScript library. It allows you to inspect the React component hierarchies in the Chrome Developer Tools.

You will get a new tab called React in your Chrome DevTools. This shows you the root React components that were rendered on the page, as well as the subcomponents that they ended up rendering.

[Website](#)

[Report Abuse](#)

Additional Information

Version: 3.2.4

Updated: July 16, 2018

Size: 314KIB

Language: English

RELATED

react-detector

★★★★☆ (52)



Refined GitHub

★★★★★ (66)



Pix to pix (Pixel perfect

by Ymatuhin)

★★★★☆ (22)



React Performance

Devtool

★★★★★ (7)





Elements

Console

Sources

Network

Performance

Memory

Application

React



Elements

Profiler



Search (text or /regex/)



Props

Empty object

State

age: 30

name: "Ryu"

▼ <App> == \$r

▼ <div className="app-content">

<h1>Hello, ninjas!</h1>

▼ <p>

"My name is: "

"Ryu"

" and I am "

"30"

</p>

</div>

</App>

App

Supported Events

- [Clipboard Events](#)
- [Composition Events](#)
- [Keyboard Events](#)
- [Focus Events](#)
- [Form Events](#)
- [Mouse Events](#)
- [Pointer Events](#)
- [Selection Events](#)
- [Touch Events](#)
- [UI Events](#)
- [Wheel Events](#)
- [Media Events](#)
- [Image Events](#)
- [Animation Events](#)
- [Transition Events](#)
- [Other Events](#)

<https://reactjs.org/docs/events.html#supported-events>

Event coordinates

`pageX`, `pageY`, `screenX`, `screenY`, `clientX`, and `clientY` returns a number which indicates the number of physical “CSS pixels” a point is from the reference point. The event point is where the user clicked, the reference point is a point in the upper left. These properties return the horizontal and vertical distance from that reference point.

`pageX` and `pageY`:

Relative to the top left of the fully rendered content area in the browser. This reference point is below the URL bar and back button in the upper left. This point could be anywhere in the browser window and can actually change location if there are embedded scrollable pages embedded within pages and the user moves a scrollbar.

`screenX` and `screenY`:

Relative to the top left of the physical screen/monitor, this reference point only moves if you increase or decrease the number of monitors or the monitor resolution.

`clientX` and `clientY`:

Relative to the upper left edge of the content area of the browser window. This point does not move even if the user moves a scrollbar from within the browser.

Create React App

- A command line tool to create React apps
 - Development server
 - Use ES6 features which are not normally supported
 - Keep our code modular
 - Use build tools to create optimized code

Create React App

build

passing

PRs

welcome

Create React apps with no build configuration.

- [Creating an App](#) – How to create a new app.
- [User Guide](#) – How to develop apps bootstrapped with Create React App.

Create React App works on macOS, Windows, and Linux.

If something doesn't work, please [file an issue](#).

🔗 Quick Overview

```
npx create-react-app my-app
cd my-app
npm start
```

([npx](#) comes with [npm 5.2+](#) and higher, see [instructions for older npm versions](#))

Then open <http://localhost:3000/> to see your app.

When you're ready to deploy to production, create a minified bundle with `npm run build`.



Introducing npx: an NPM package runner

`npx` is a tool intended to help round out the experience of using packages from the NPM registry—the same way `npm` makes it super easy to install and manage dependencies hosted on the registry, **`npx` makes it easy to use CLI tools and other executables hosted on the registry.**

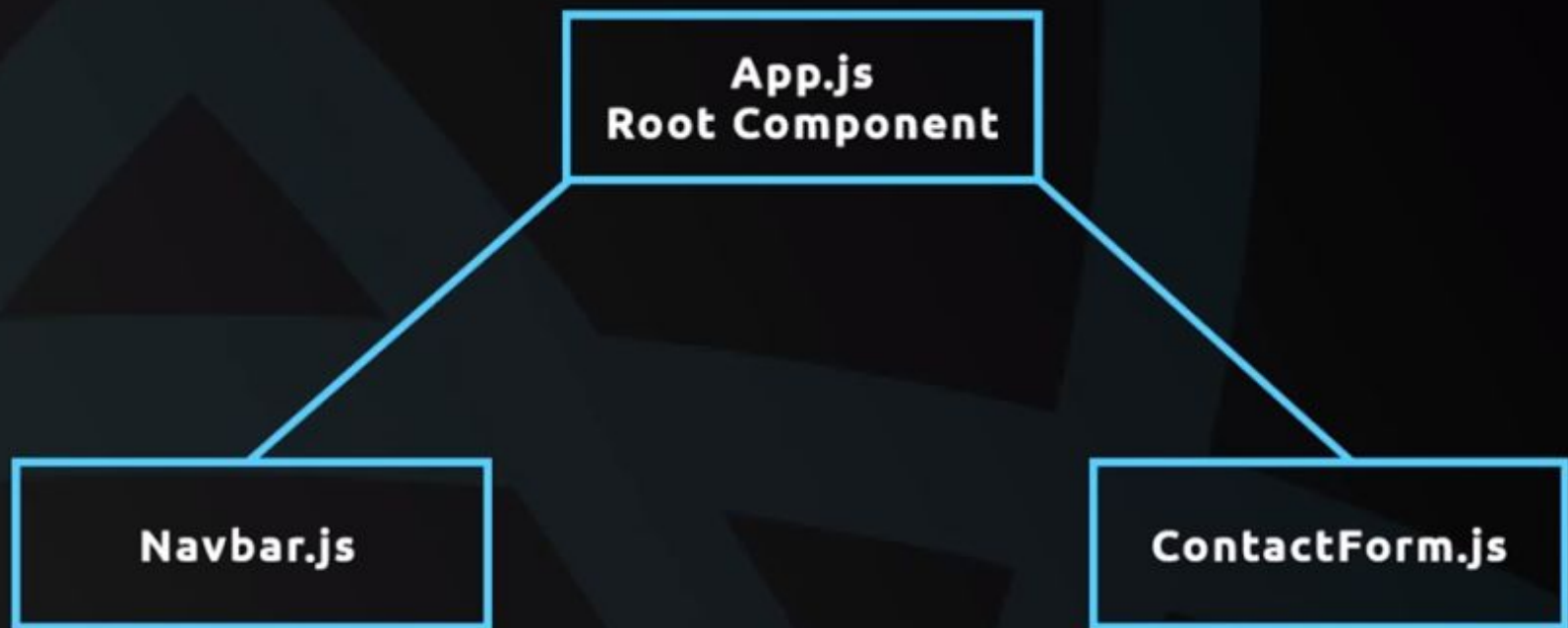
Included with `npm@5.2.0` is an extra binary named `npx`



Edit `src/App.js` and save to reload.

[Learn React](#)

Nesting Components



Container vs UI Components

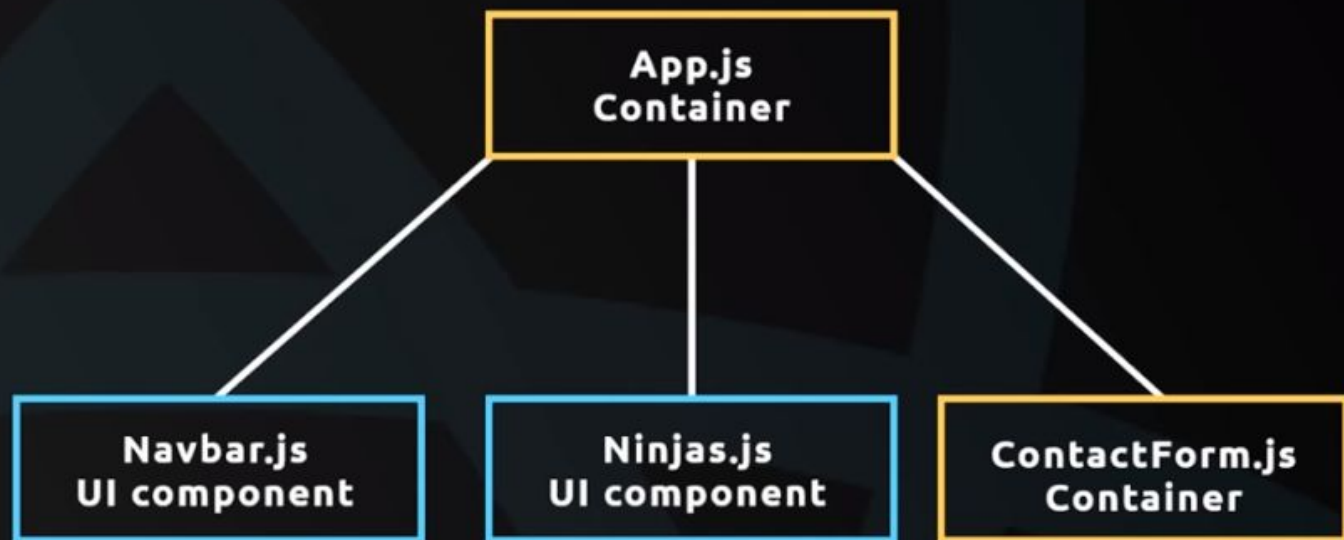
Container Components

- Contain state
- Contain lifecycle hooks
- Not concerned with UI
- Use classes to create

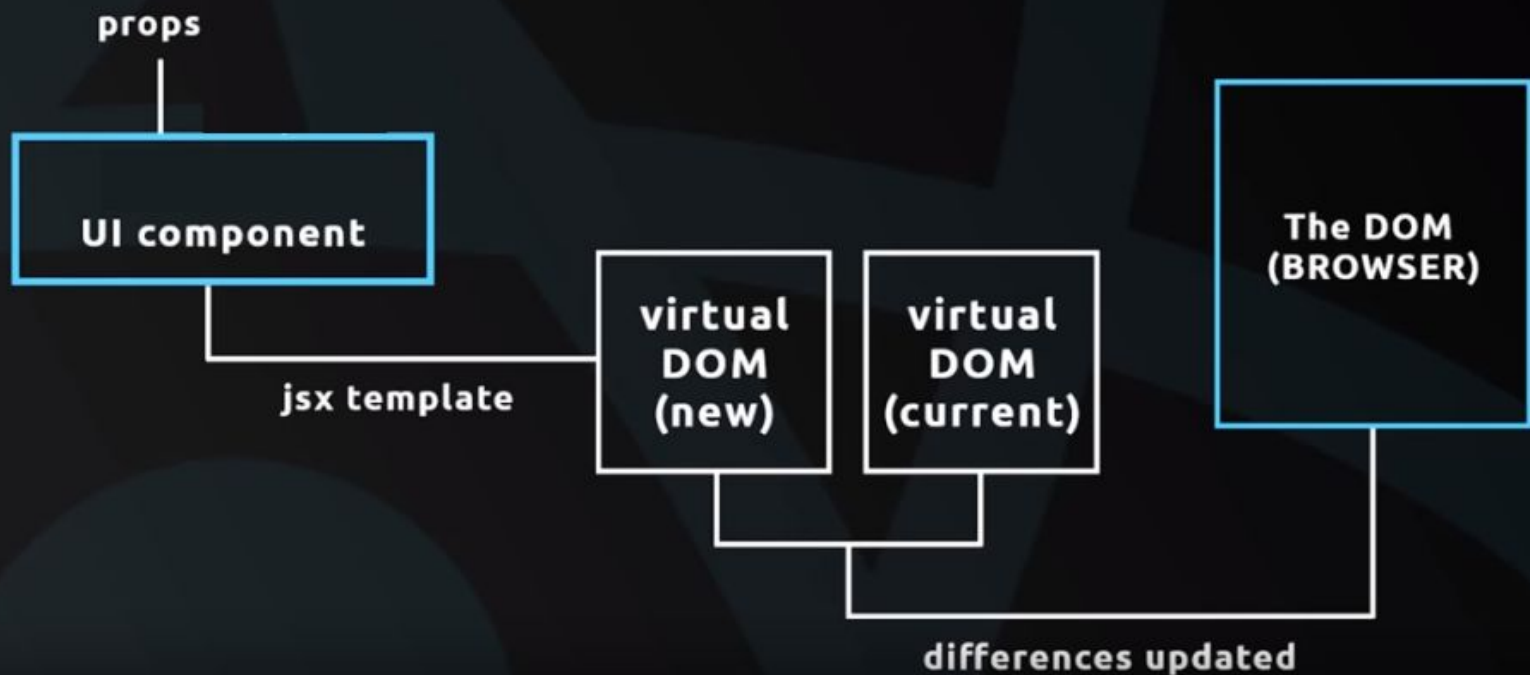
UI Components

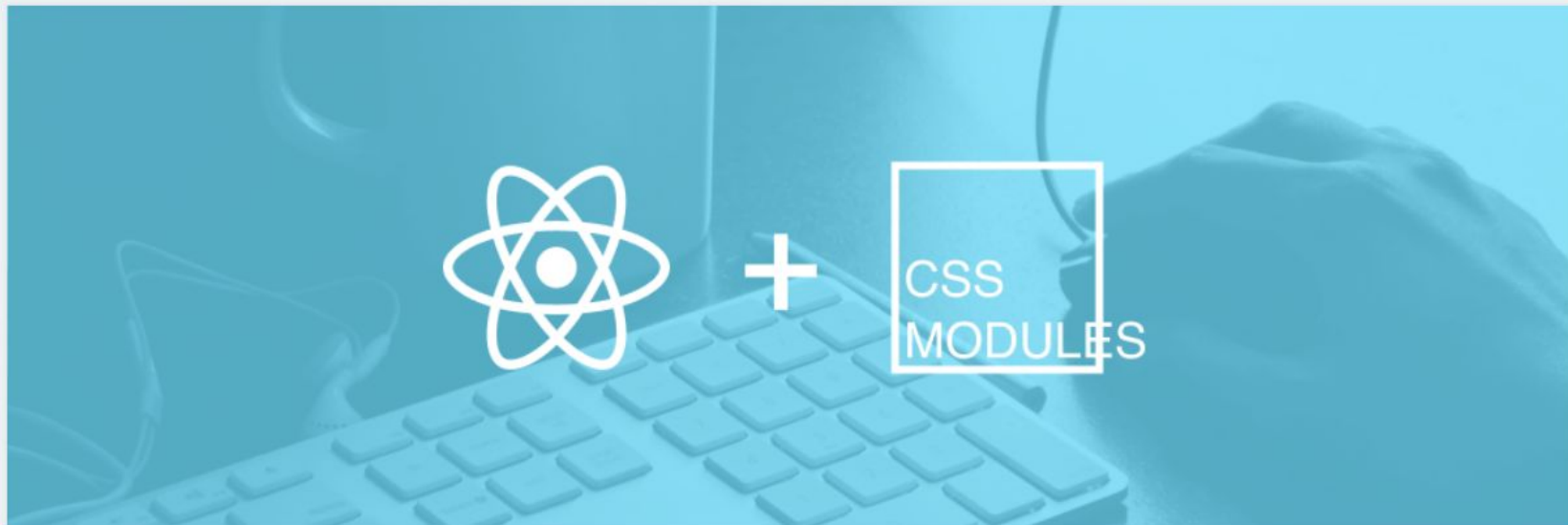
- Don't contain state
- Receive data from props
- Only concerned with UI
- Use functions to create

Container vs UI Components



Recap & Virtual DOM





NEO IGHODARO

Getting started with CSS modules in React

<https://blog.pusher.com/css-modules-react/>



1.0.0 ▼

Search



About

Getting Started

CSS

Components

JavaScript

Forms

Mobile

Showcase

Themes

updated

Getting Started

Learn how to easily start using Materialize in your website.

Download

Materialize comes in two different forms. You can select which version you want depending on your preference and expertise. To start using Materialize, all you have to do is download one of the options below.

Materialize

This is the standard version that comes with both the minified and unminified CSS and JavaScript files. This option requires little to no setup. Use this if you are unfamiliar with Sass.

MATERIALIZE



Sass

This version contains the source SCSS files. By choosing this version you have more control over which components to include. You will need a Sass compiler if you choose this option.

SOURCE



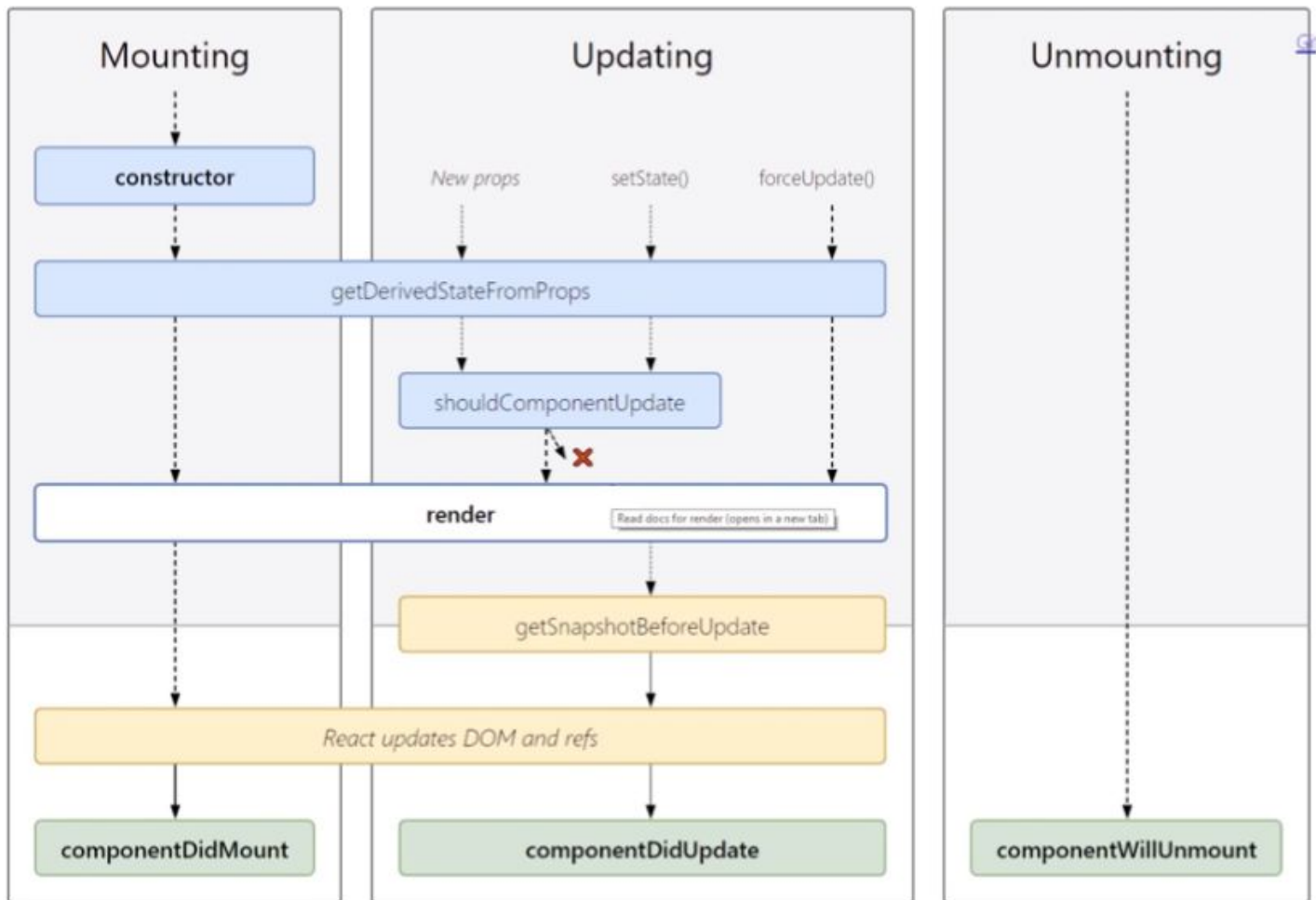
<https://materializecss.com/getting-started.html>

Component Lifecycle Phases:

Mounting phase- when component first created and mounted into the DOM

Updating- An update can be caused by changes to props or state

Unmounting- when a component is being removed from the DOM



Mounting Phase:

constructor- allow another way to set the state

getDerivedStateFromProps- enables component to update its internal state as a result of changes in our props. Triggers in first render and in changes in parent components. We receive props and state and we can compare the prop to the current state and return new state object or null for no changes (rarely use).

render- React takes JSX and prepare to render to the DOM (required)

componentDidMount- fires once the component first mounts. Good place to fire AJAX

Updating Phase:

getDerivedStateFromProps- like before

shouldComponentUpdate- receive the next prop and the next state. We can compare older versus newer, so we can return false if we want to prevent the component from updating and re-rendering. We decide if we want an update

getSnapshotBeforeUpdate- we get read access to the DOM before the change is committed to it. We can get current values from the DOM such as window position, and return that inside the method - this value will return to the final update hook

componentDidUpdate- after the template is rendered to the actual DOM. we can have write access to the DOM. we can fire AJAX here. Caution! Do not update state because we can get infinite loop

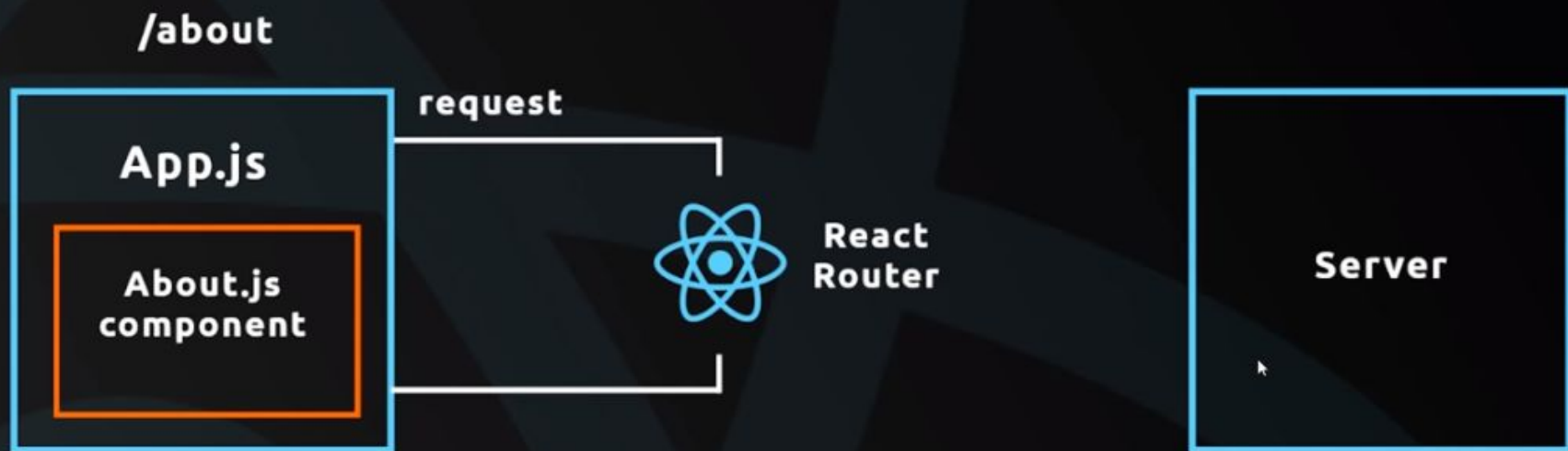
componentWillUnmount()

```
componentWillUnmount()
```

`componentWillUnmount()` is invoked immediately before a component is unmounted and destroyed. Perform any necessary cleanup in this method, such as invalidating timers, canceling network requests, or cleaning up any subscriptions that were created in `componentDidMount()`.

You **should not call `setState()`** in `componentWillUnmount()` because the component will never be re-rendered. Once a component instance is unmounted, it will never be mounted again.

React Router



Higher Order Components



Route Parameters

- Part of a URL that can change
- Usually see them on URLs for individual records

User Profile Route:

`mysite.com/users/yoshi2k1`
`mysite.com/users/mariotheman`

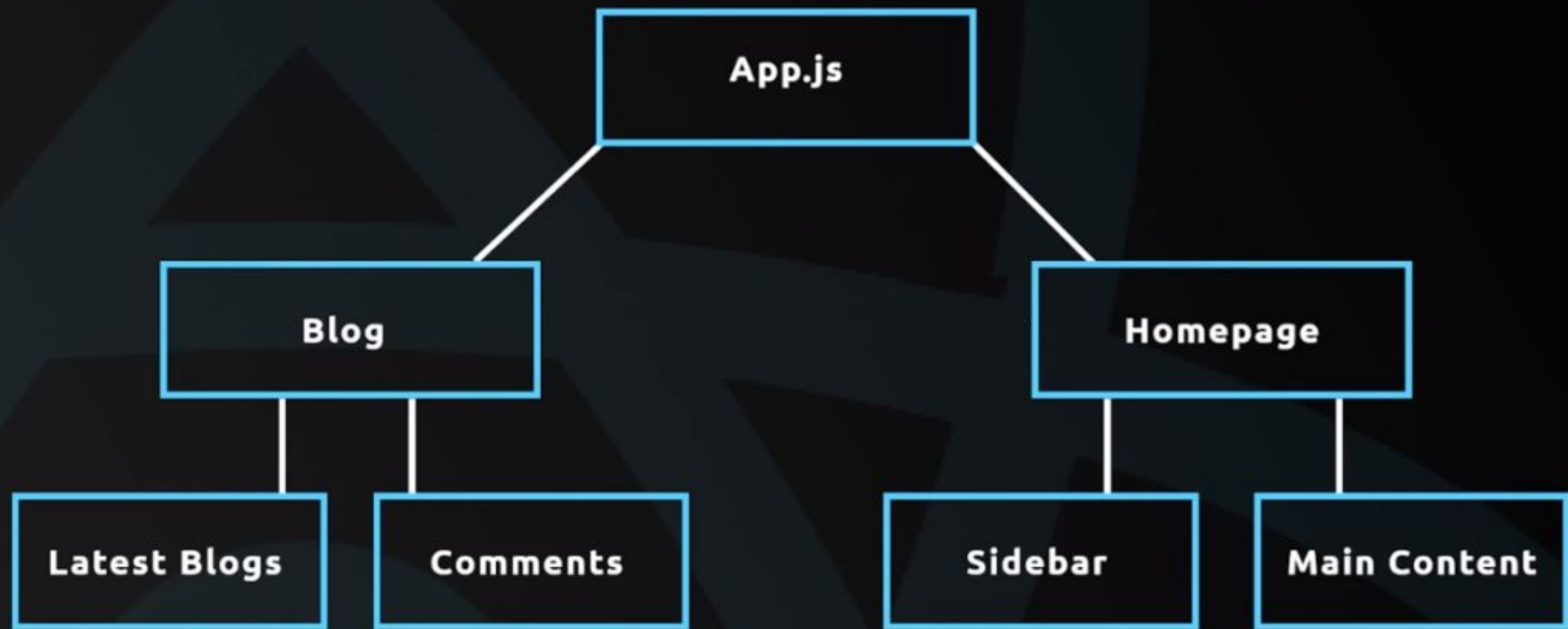
Cooking recipe page:

`mycookingsite.com/recipes/12345`
`mycookingsite.com/recipes/98765`

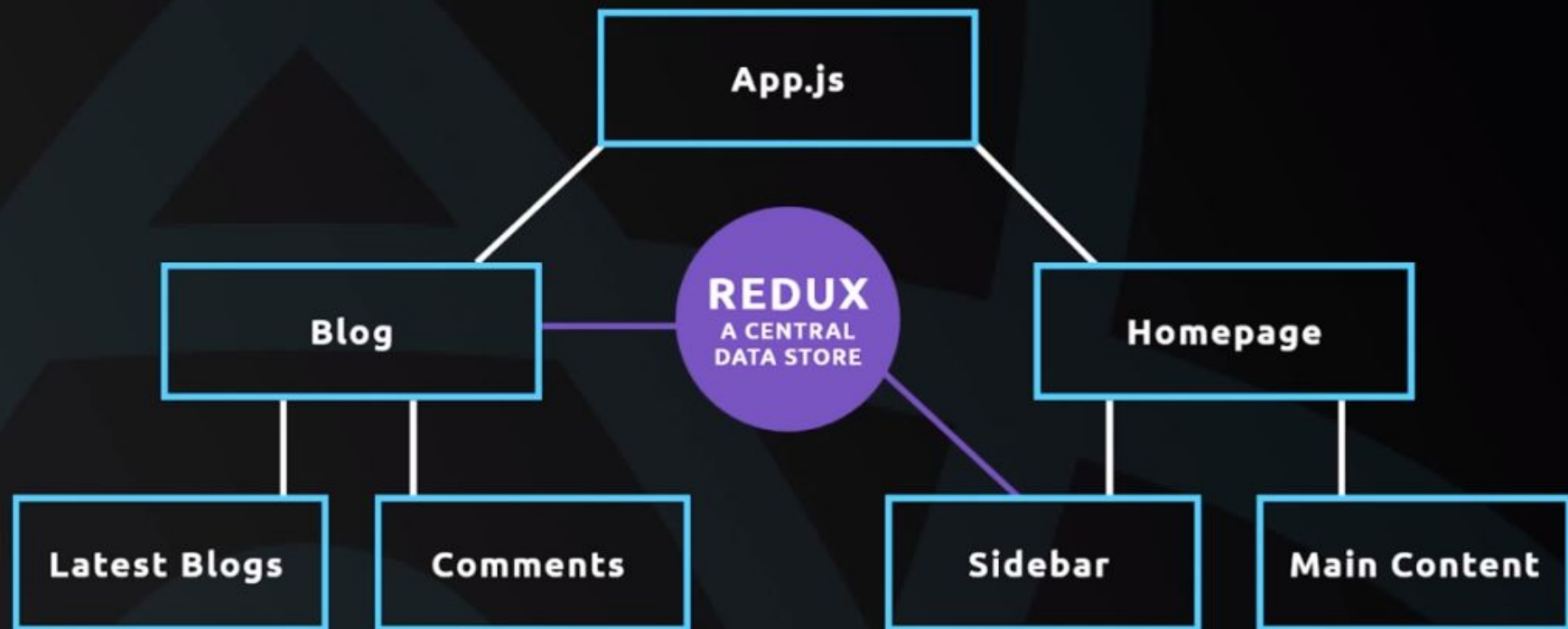
Redux

- Central data store for all app data
- Any component can access data from it
- Makes state management easy

Redux



Redux



Redux

