## Environment Setup



**Cory House** 

@housecor

reactjsconsulting.com



## React Starter Projects

These are repositories that you copy and modify to create your own React app. Pick a starter project with all the features you need, and none that you don't.

To get started with most of these, you would clone the repository and then start making modifications.

Other starters are installed rather than copied: they become a dependency of your project. These dependency starters make it easier to get updates and they keep your project directory cleaner (because much of the code is wrapped in the dependency).

Q filter projects AGE

198 Projects

Sort ▼



## Option: Build a dev environment.

#### Why?

- 1. Understand how create-react-app works
- 2. Learn how to customize your dev environment



ና[ፖ

((y)

噐

18%

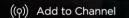
## Building Applications with React and Redux

by Cory House

Learn how to use Redux, React Router, and ES6 to build a real world app with React. Use Webpack, Babel, ESLint, npm scripts, Mocha, Enzyme, and more to build a rich, one step, custom React development environment and build process from the ground up.

Resume Course





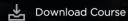


Table of contents

Description

Transcript **Exercise files**  Discussion

Learning Check

Recommended

These exercise files are intended to provide you with the assets you need to create a video-based hands-on experience. With the exercise files, you can follow along with the author and re-create the same solution on your computer. We find this to be even more effective than written lab exercises.

Download exercise files

Want to skip the environment setup? Use the "after" folder in "03-environment setup"

#### Course author



Cory is an independent consultant with over 15 years of experience in software development. He is the principal consultant at reactjsconsulting.com and a Microsoft MVP.

#### Course info

	Intermediate
Rating	**** (1183)

#### Share course









### Our Dev Environment

**Compile JSX** 

**Transpile JS** 

Linting

Generate index.html

Reload on save

One command!



## The Plan



#### **Build development environment**

- Node
- Webpack
- Babel
- ESLint
- npm Scripts





HOME ABOUT

DOWNLOADS

DOCS

GET INVOLVED

SECURITY

NEWS

**FOUNDATION** 

Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine.

### Download for macOS (x64)

10.15.0 LTS

**Recommended For Most Users** 

11.6.0 Current

**Latest Features** 

Run at least Node 8.

Need to run multiple versions of Node? Two options:

- 1. Install Node 8 or newer for this course, then reinstall your old version when you're done.
- 2. Try nvm on Mac or nvm-windows on Windows.

**ጎ**የ

((y)

包

ψ

## Building Applications with React and Redux

by Cory House

Learn how to use Redux, React Router, and ES6 to build a real world app with React. Use Webpack, Babel, ESLint, npm scripts, Mocha, Enzyme, and more to build a rich, one step, custom React development environment and build process from the ground up.

Resume Course



((γ)) Add to Channel



Table of contents Description Transcript **Exercise files** Discussion Learning Check Recommended

These exercise files are intended to provide you with the assets you need to create a video-based hands-on experience. With the exercise files, you can follow along with the author and re-create the same solution on your computer. We find this to be even more effective than written lab exercises.

Download exercise files

#### Course author



Cory House

Cory is an independent consultant with over 15 years of experience in software development. He is the principal consultant at reactjsconsulting.com and a Microsoft MVP.

#### Course info

	Intermediate
Rating	**** (1183)

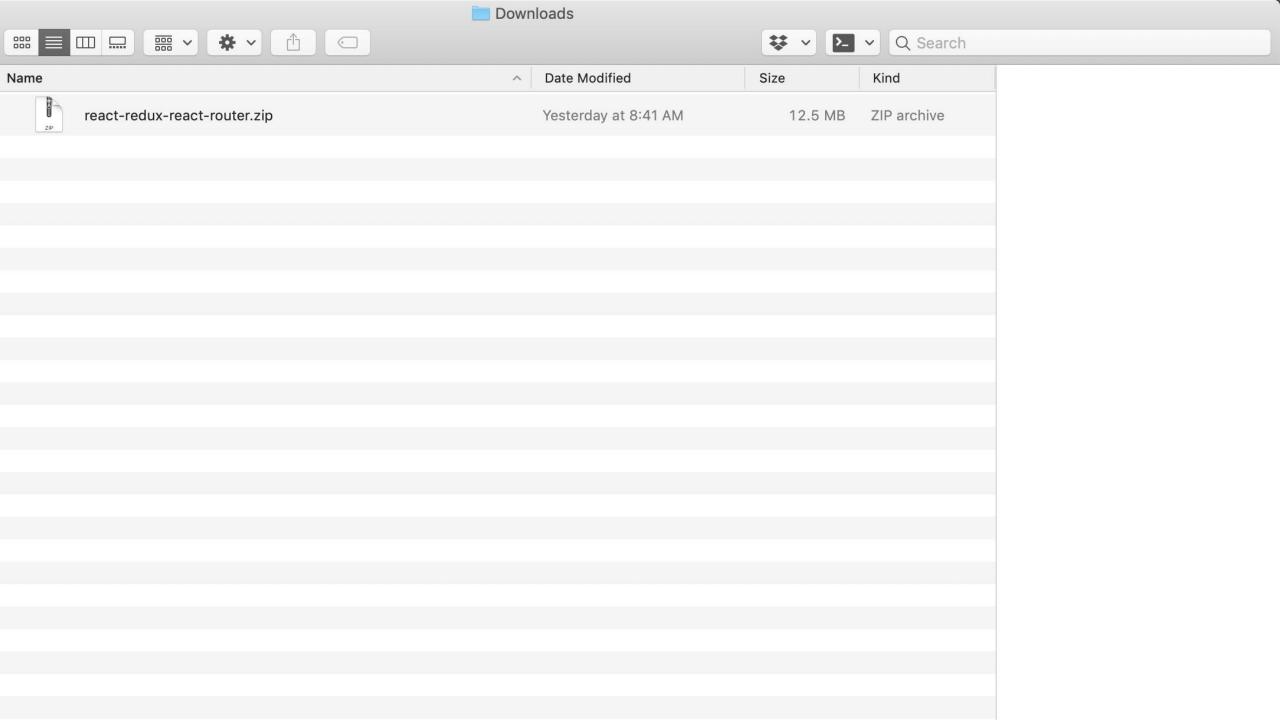
Share course

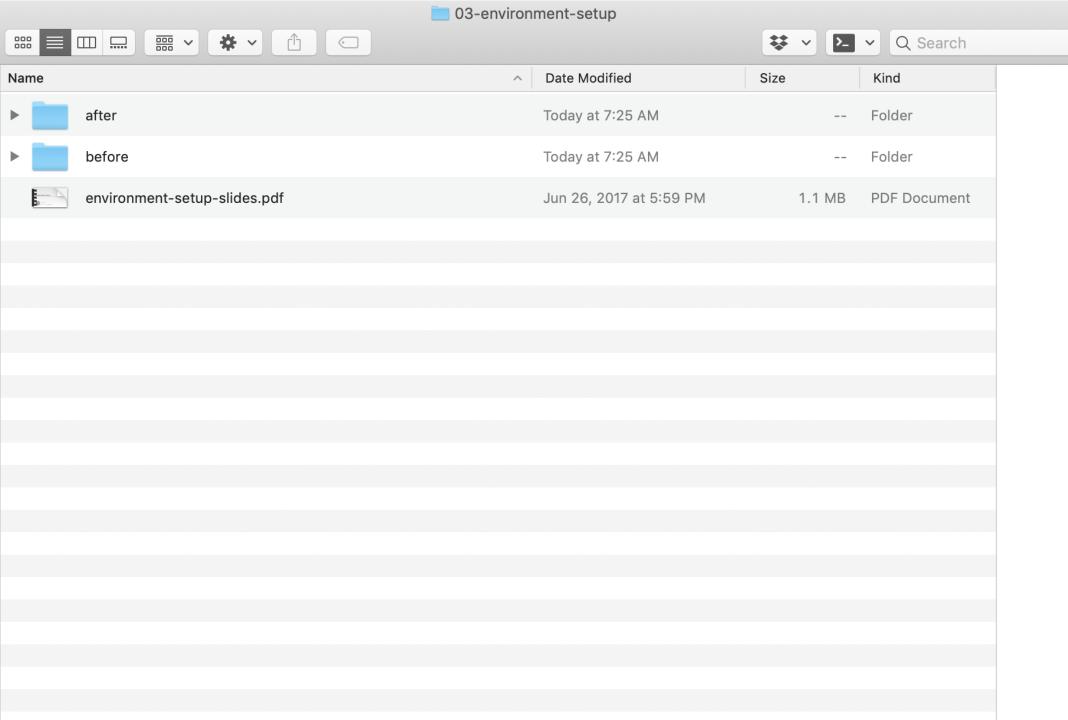












Version 1.30 is now available! Read about the new features and fixes from November.

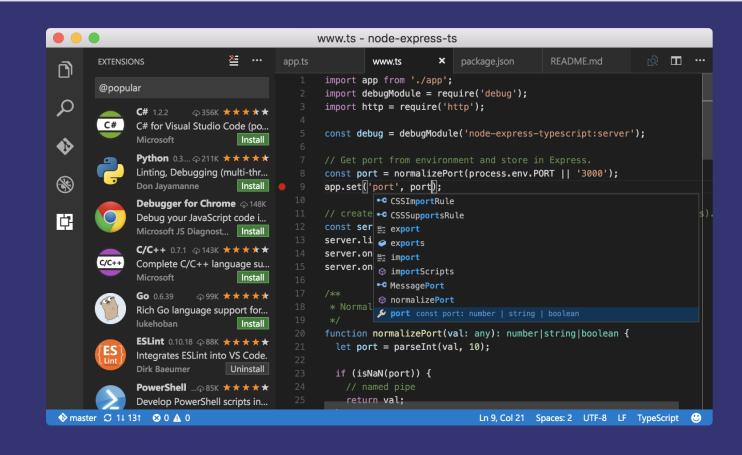


Free. Open source. Runs everywhere.

**Download for Mac** Stable Build

Other platforms and Insiders Edition

license and privacy statement.





IntelliSense







Extensions

**TRY IT OUT** 

**GET STARTED** 

**OPTIONS** 

#### # What is Prettier?

- \* An opinionated code formatter
- \* Supports many languages
- \* Integrates with most editors
- \* Has few options

#### # Why?

- \* You press save and code is formatted
- \* No need to discuss style in code review
- \* Saves you time and energy
- And more

#### Works with the Tools You Use



ES2017 JSX Flow TypeScript **JSON** 



Vue Angular



CSS3+ Less SCSS styled-components 💋 styled-jsx



GraphQL GraphQL Schemas



CommonMark GitHub Flavored Markdown MDX



YAML



Work in Progress Elm (via elm-format) Java PHP **PostgreSQL** 

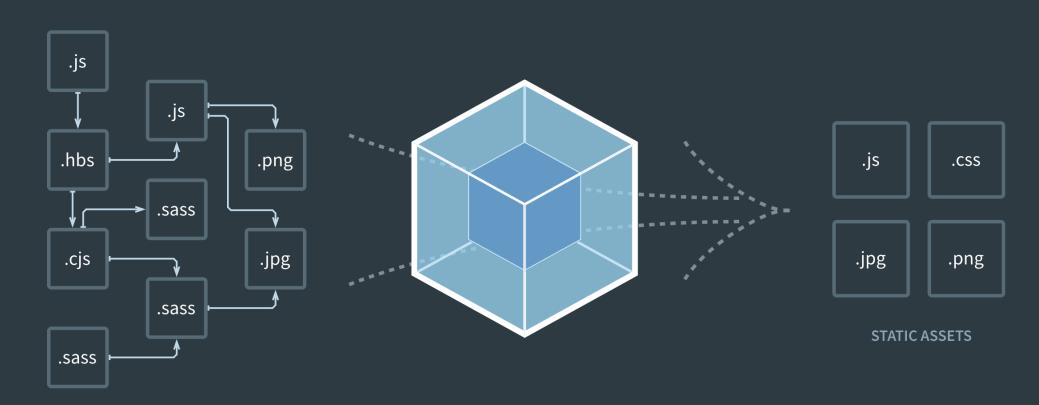
Python







## bundle your assets



# 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

var [a,,b] = [1,2,3];

Get browser-compatible JavaScript out

# 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 Get browser-compatible JavaScript out var [a,,b] = [1,2,3]; Check out our REPL to experiment more with Babel!

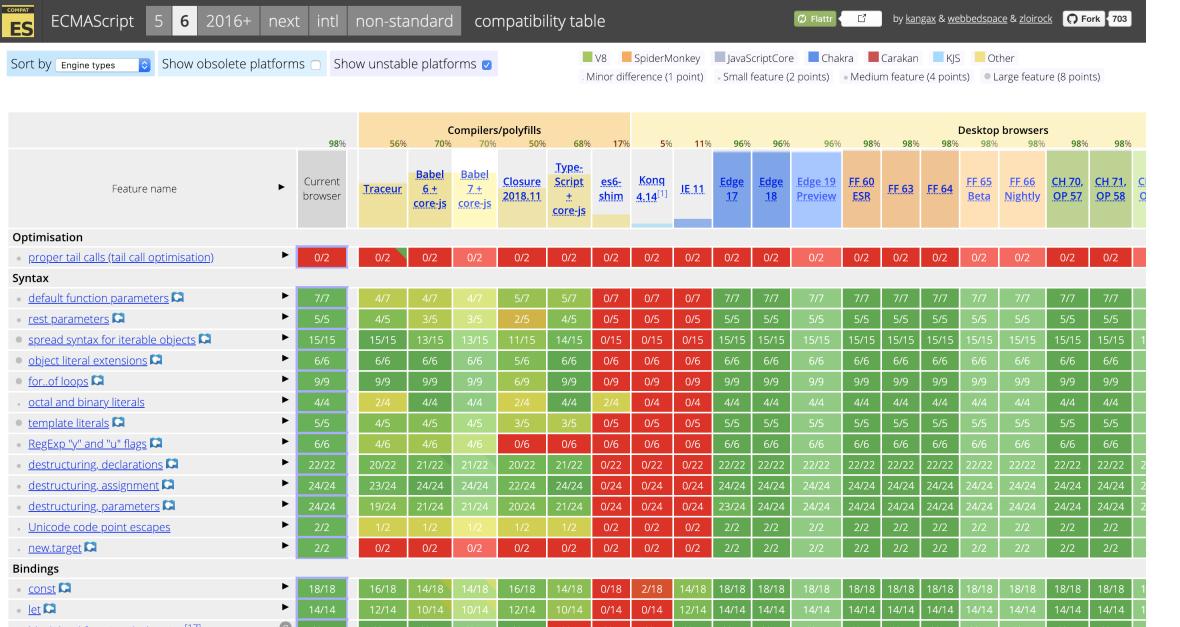
## Why Babel?

- 1. Transpile modern JS
- 2. Compile JSX to JS



	Edition	Date published	Changes from prior edition
ES1	1	June 1997	First edition
ES2	2	June 1998	Editorial changes to keep the specification fully aligned with ISO/IEC 16262 international standard
ES3	3	December 1999	Added regular expressions, better string handling, new control statements, try/catch exception handling, tighter definition of errors, formatting for numeric output and other enhancements
ES4	4	Abandoned	Fourth Edition was abandoned, due to political differences concerning language complexity. Many features proposed for the Fourth Edition have been completely dropped; some were incorporated into the sixth edition.
ES5	5	December 2009	Adds "strict mode," a subset intended to provide more thorough error checking and avoid error-prone constructs. Clarifies many ambiguities in the 3rd edition specification, and accommodates behaviour of real-world implementations that differed consistently from that specification. Adds some new features, such as getters and setters, library support for JSON, and more complete reflection on object properties. <sup>[9]</sup>
	5.1	June 2011	This edition 5.1 of the ECMAScript standard is fully aligned with third edition of the international standard ISO/IEC 16262:2011.
ES6 / ES2015	6	June 2015 <sup>[10]</sup>	The sixth edition, initially known as ECMAScript 6 (ES6) and later renamed to ECMAScript 2015 (ES2015) <sup>[10]</sup> adds significant new syntax for writing complex applications, including classes and modules, but defines them semantically in the same terms as ECMAScript 5 strict mode. Other new features include iterators and for/of loops, Python-style generators and generator expressions, arrow functions, binary data, typed arrays, collections (maps, sets and weak maps), promises, number and math enhancements, reflection, and proxies (metaprogramming for virtual objects and wrappers). As the first "ECMAScript Harmony" specification, it is also known as "ES6 Harmony."
ES7 / ES2016	7	June 2016 <sup>[11]</sup>	ECMAScript 2016 (ES2016) <sup>[11]</sup> , the seventh edition, intended to continue the themes of language reform, code isolation, control of effects and library/tool enabling from ES2015, includes two new features: the exponentiation operator (**) and Array.prototype.includes.
ES8 / ES2017	8	June 2017 <sup>[12]</sup>	ECMAScript 2017 (ES2017), the eighth edition, includes features for concurrency and atomics, syntactic integration with promises (async/await). <sup>[13][12]</sup>
ES9 / ES2018	9	June 2018 <sup>[8]</sup>	ECMAScript 2018 (ES2018), the ninth edition, includes features for asynchronous iteration and generators, new regular expression features and rest/spread parameters. <sup>[8]</sup>









# 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 Get browser-compatible JavaScript out var [a,,b] = [1,2,3]; Check out our REPL to experiment more with Babel!

## Why Babel?

- 1. Transpile modern JS
- 2. Compile JSX to JS







The pluggable linting utility for JavaScript and JSX

Get Started »

#### Welcome

ESLint is an open source project originally created by Nicholas C. Zakas in June 2013. Its goal is to provide a pluggable linting utility for JavaScript.

#### **Latest News**

• ESLint v5.11.1 released 26 December 2018

## Summary



#### Dev environment complete!

- Transpile: Babel
- Bundle: Webpack
- Lint: ESLint
- Web server: Webpack
- Generating index.html: Webpack
- Loading changes on save: Webpack
- One command via npm scripts!

Next up: React component approaches

