

Bootstrap

— START WITH SOURCE FILES —



IN A ROCKET

Learn front-end development at rocket speed

**BOOTSTRAP
CDN**

**COMPILED
CSS & JS**

**SOURCE
FILES**

**PACKAGE
MANAGERS**

B

Home Documentation Examples Themes Jobs Expo Blog

Search...

Getting started

Introduction

Download

Contents

Browsers & devices

JavaScript

Theming

Build tools

Webpack

Accessibility

Layout

Content

Components

Utilities

Extend

Source files

Compile Bootstrap with your own asset pipeline by downloading our source Sass, JavaScript, and documentation files. This option requires some additional tooling:

2

- Sass compiler (or Ruby Sass is supported) for compiling your CSS.
- Autoprefixer for vendor prefixing

Should you require build tools, they are included for developing Bootstrap and its docs, but they're likely unsuitable for your own purposes.

1

bootstrap-4.0.0.zip

Download source

BootstrapCDN

Skip the download with BootstrapCDN to deliver cached version of Bootstrap's compiled CSS and JS to your project.

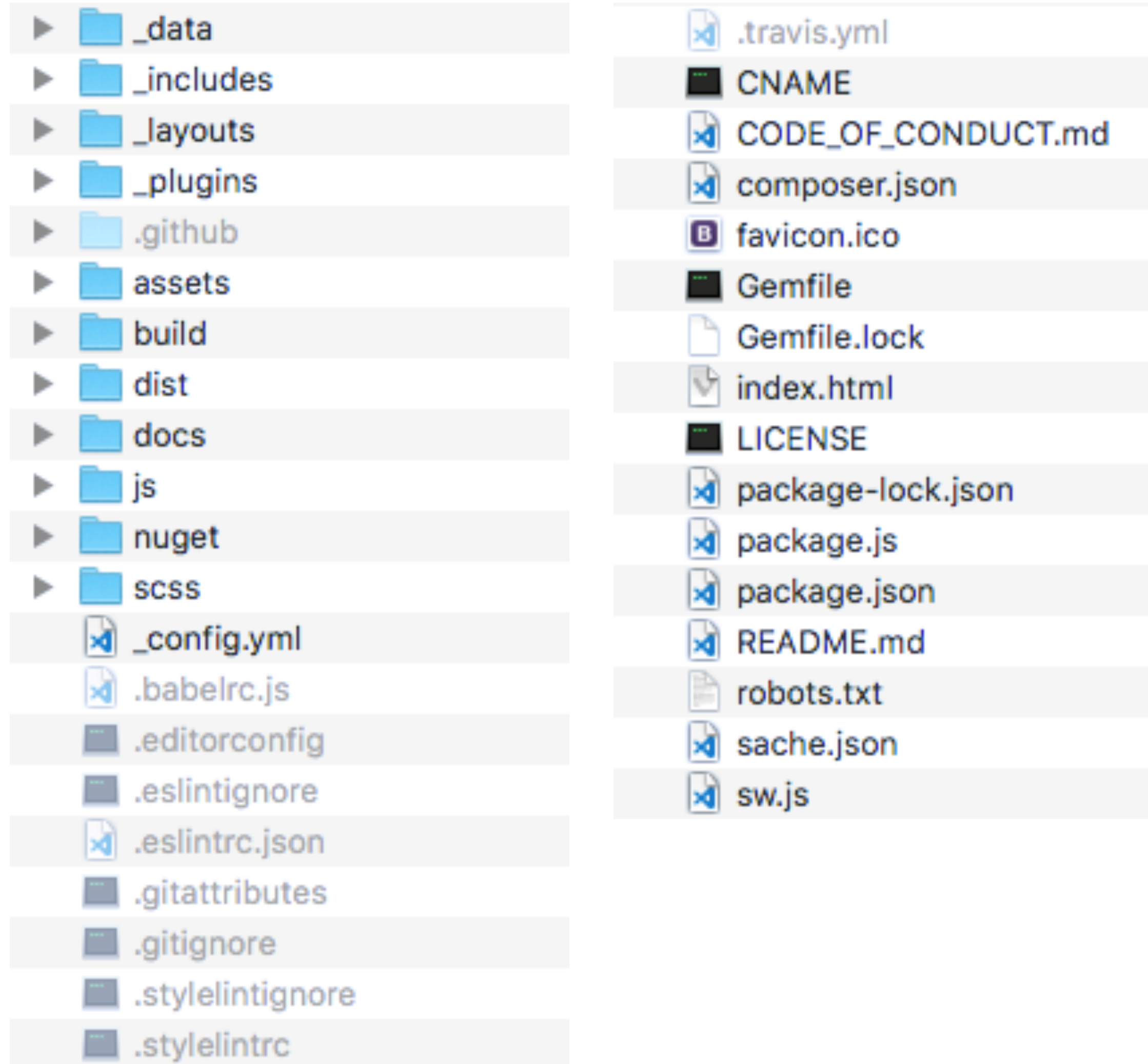
```
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min
```

BOOTSTRAP 4: Best practices and real challenges

inarocket.com

SOURCE FILES

1



B

Home Documentation Examples Themes Jobs Expo Blog

Search...

Getting started

Introduction

Download

Contents

Browsers & devices

JavaScript

Theming

Build tools

Webpack

Accessibility

Layout

Content

Components

Utilities

Extend

Migration

Build tools

Learn how to use Bootstrap's included npm scripts to build our documentation, compile source code, run tests, and more.

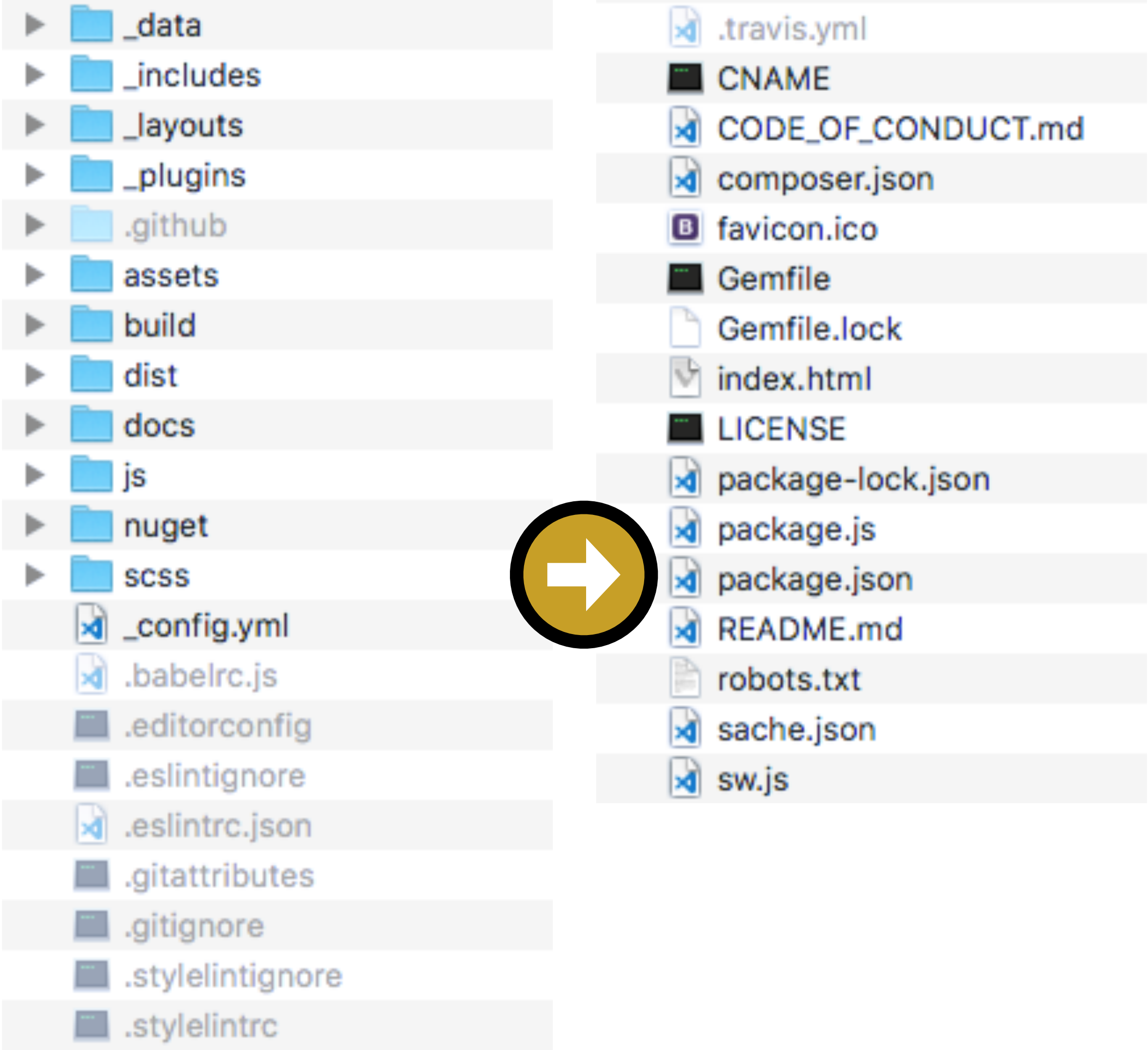
Tooling setup

Bootstrap uses [NPM scripts](#) for its build system. Our [package.json](#) includes convenient methods for working with the framework, including compiling code, running tests, and more.

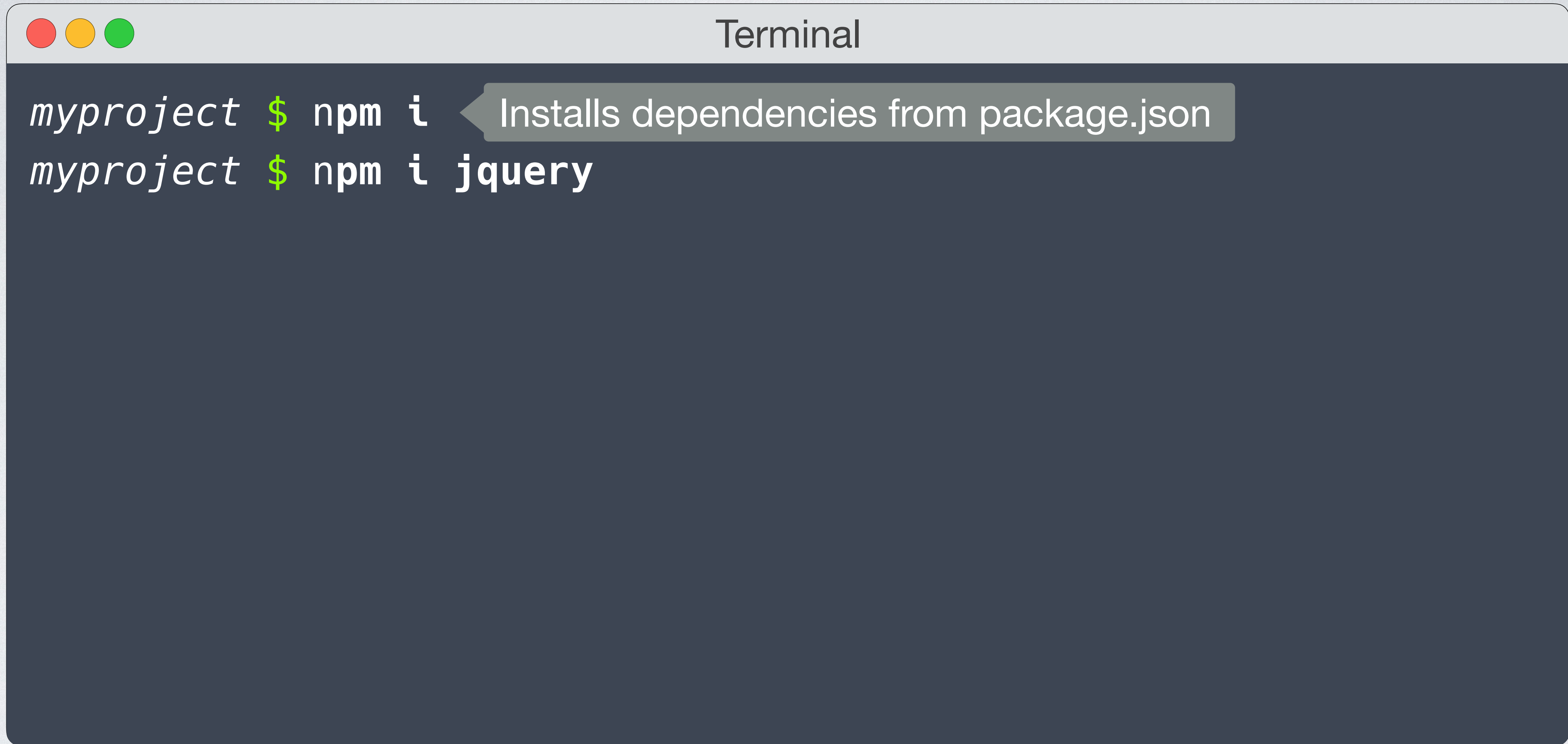
To use our build system and run our documentation locally, you'll need a copy of Bootstrap's source files and Node. Follow these steps and you should be ready to rock:

1. [Download and install Node.js](#), which we use to manage our dependencies.
2. Navigate to the root `/bootstrap` directory and run `npm install` to install our local dependencies listed in [package.json](#).
3. [Install Ruby](#), install [Bundler](#) with `gem install bundler`, and finally run `bundle install`. This will install all Ruby dependencies, such as Jekyll and plugins.
 - **Windows users:** Read [this guide](#) to get Jekyll up and running without problems.

SOURCE FILES



SOURCE FILES



A terminal window titled "Terminal" with a light gray title bar and three colored window control buttons (red, yellow, green) on the left. The terminal has a dark blue background. It shows two lines of text: `myproject $ npm i` followed by a gray tooltip box containing the text "Installs dependencies from package.json", and then `myproject $ npm i jquery` on the next line.

```
myproject $ npm i
myproject $ npm i jquery
```


SOURCE FILES

Terminal

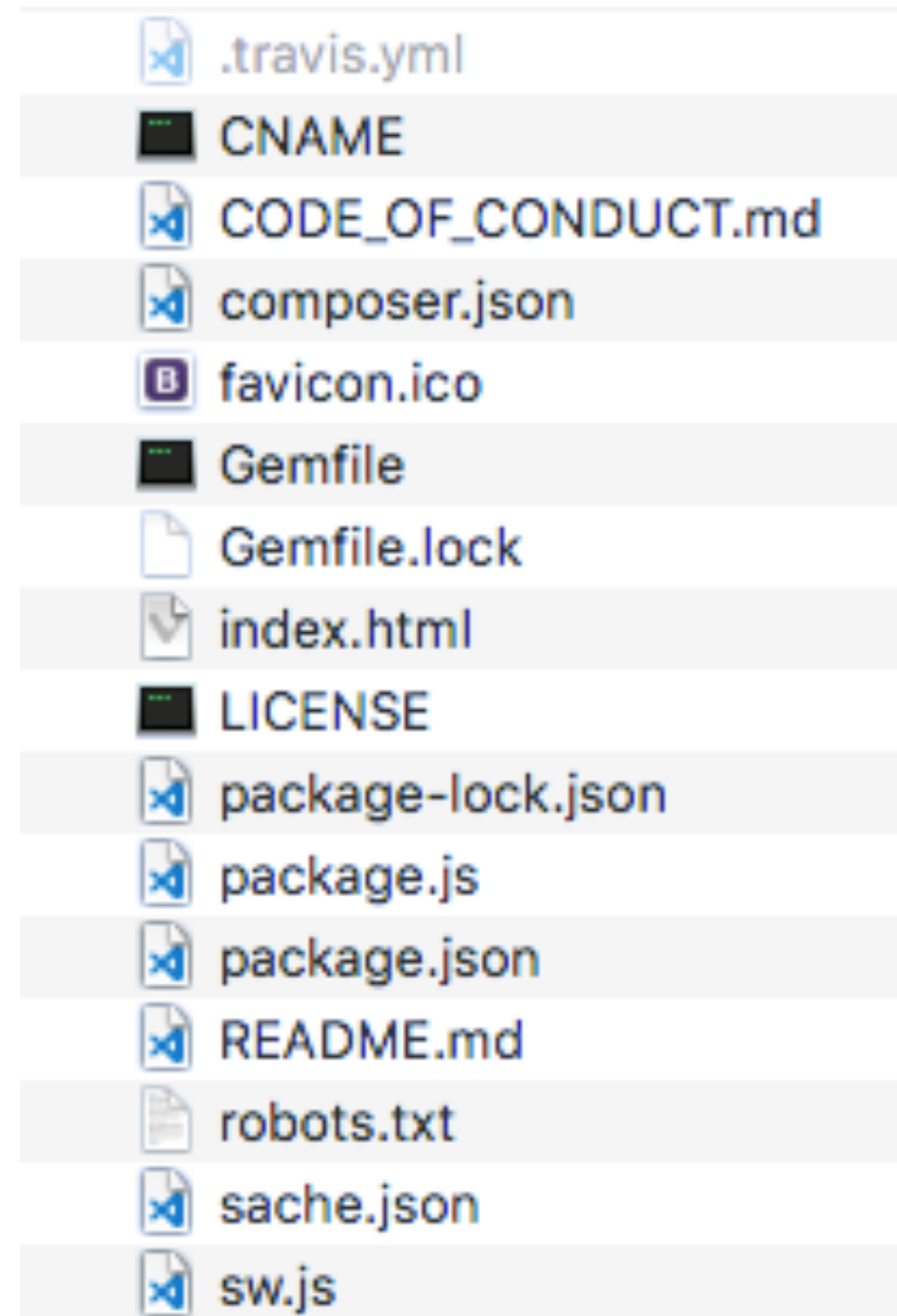
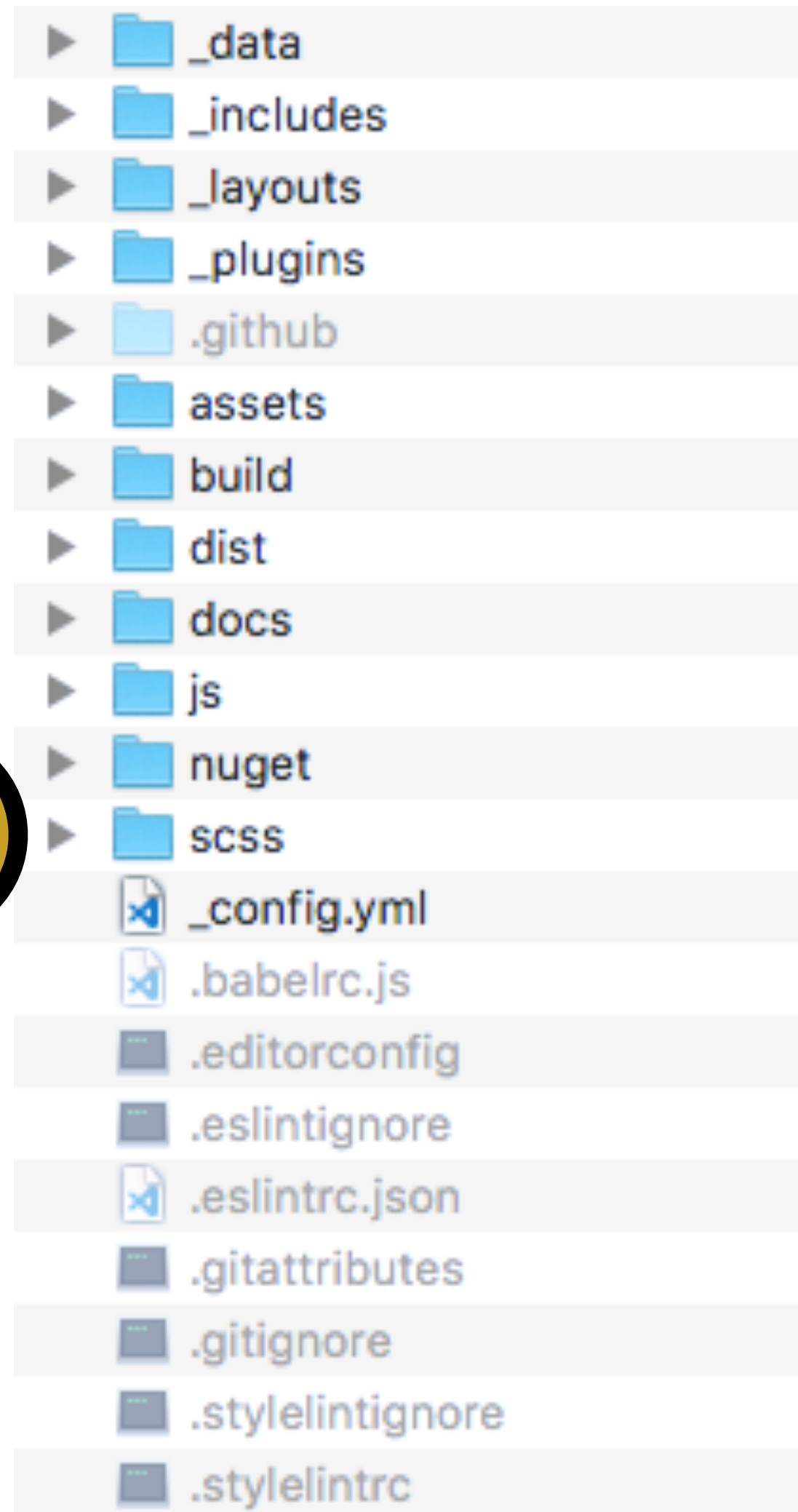
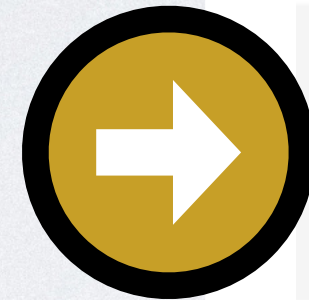
myproject \$ npm ls -dev --depth=0

Lists installed dependencies

```
@babel/cli@7.0.0-beta.37
@babel/core@7.0.0-beta.37
@babel/preset-env@7.0.0-beta.37
autoprefixer@7.2.4
babel-eslint@8.2.1
babel-plugin-transform-es2015-modules-strip@0.1.1
bundlesize@0.15.3
clean-css-cli@4.1.10
cross-env@5.1.3
eslint@4.15.0
eslint-plugin-compat@2.1.0
glob@7.1.2
htmlhint-cli@0.0.6
jsunitsaucelabs@1.3.0
karma@2.0.0
karma-chrome-launcher@2.2.0
karma-detect-browsers@2.2.6
karma-firefox-launcher@1.1.0
karma-qunit@1.2.1
node-sass@4.7.2
nodemon@1.14.10
npm-run-all@4.1.2
popper.js@1.12.9
postcss-cli@4.1.1
qunitjs@2.4.1
rollup@0.53.4
rollup-plugin-babel@4.0.0-beta.0
rollup-plugin-node-resolve@3.0.2
shelljs@0.7.8
shx@0.2.2
sri-toolbox@0.2.0
stylelint@8.4.0
stylelint-config-recommended-scss@3.0.0
stylelint-config-standard@18.0.0
stylelint-order@0.8.0
stylelint-scss@2.2.0
uglify-js@3.3.5
vnu-jar@17.11.1
workbox-build@2.1.2
```

node-sass

SOURCE FILES



BOOTSTRAP 3



BOOTSTRAP 4



Bootstrap 4 ships with vanilla CSS,
but its source code utilizes **Sass**.
Now compiles faster thanks to Libsass.



```
$ npm run dist
```

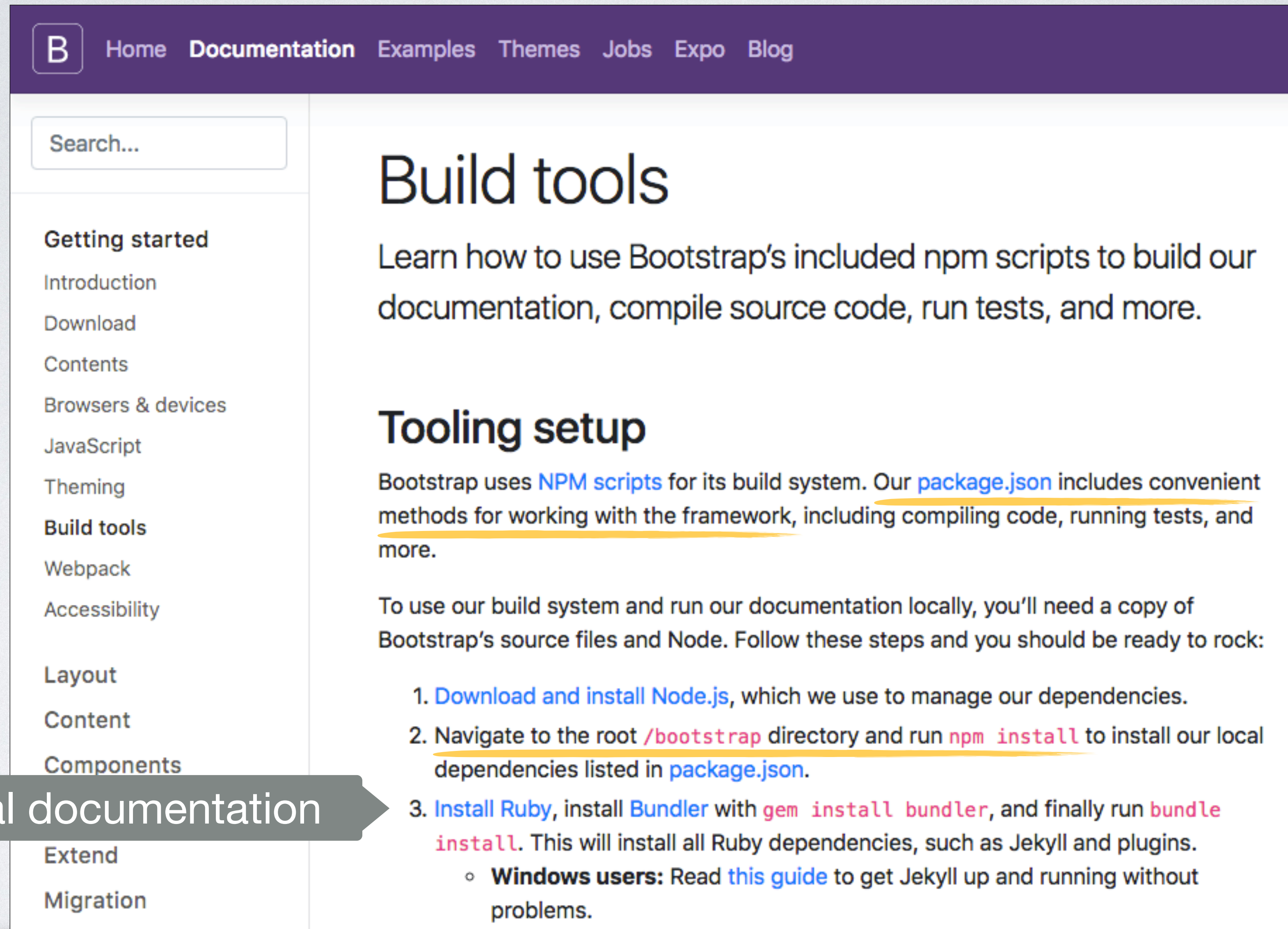
Creates the /dist directory with compiled files.
Uses Sass, Autoprefixer, and UglifyJS.


```
index.html — bootstrap-4.0.0

1 <!doctype html>
2 <html lang="en">
3
4 <head>
5   <!-- Required meta tags -->
6   <meta charset="utf-8">
7   <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
8
9   <!-- Bootstrap CSS -->
10  <link rel="stylesheet" href="../dist/css/bootstrap.min.css">
11
12  <title>Hello, world!</title>
13</head>
14
15<body>
16  <h1>Hello, world!</h1>
17
18  <!-- Optional JavaScript -->
19  <!-- jQuery first, then Popper.js, then Bootstrap JS -->
20  <script src="../node_modules/jquery/dist/jquery-3.2.1.slim.min.js"></script>
21  <script src="../dist/js/bootstrap.bundle.min.js"></script>
22</body>
23
24</html>
25
```




Create a new snippet
for this starter template



The screenshot shows the Bootstrap 4 documentation website. The top navigation bar is purple with a 'B' logo and links to Home, Documentation, Examples, Themes, Jobs, Expo, and Blog. The 'Documentation' link is highlighted. On the left, a sidebar contains a search bar and a list of navigation links: Getting started, Introduction, Download, Contents, Browsers & devices, JavaScript, Theming, Build tools, Webpack, Accessibility, Layout, Content, Components, Extend, and Migration. The 'Build tools' link is highlighted. The main content area has a heading 'Build tools' and a paragraph: 'Learn how to use Bootstrap's included npm scripts to build our documentation, compile source code, run tests, and more.' Below this is a section titled 'Tooling setup' with a paragraph: 'Bootstrap uses NPM scripts for its build system. Our package.json includes convenient methods for working with the framework, including compiling code, running tests, and more.' This paragraph is underlined. Below it is another paragraph: 'To use our build system and run our documentation locally, you'll need a copy of Bootstrap's source files and Node. Follow these steps and you should be ready to rock:'. This is followed by a numbered list of three steps: 1. Download and install Node.js, which we use to manage our dependencies. 2. Navigate to the root /bootstrap directory and run npm install to install our local dependencies listed in package.json. 3. Install Ruby, install Bundler with gem install bundler, and finally run bundle install. This will install all Ruby dependencies, such as Jekyll and plugins. A sub-bullet point for 'Windows users' reads: 'Read this guide to get Jekyll up and running without problems.'

B Home Documentation Examples Themes Jobs Expo Blog

Search...

Getting started
Introduction
Download
Contents
Browsers & devices
JavaScript
Theming
Build tools
Webpack
Accessibility

Layout
Content
Components
Extend
Migration

Build tools

Learn how to use Bootstrap's included npm scripts to build our documentation, compile source code, run tests, and more.

Tooling setup

Bootstrap uses NPM scripts for its build system. Our package.json includes convenient methods for working with the framework, including compiling code, running tests, and more.

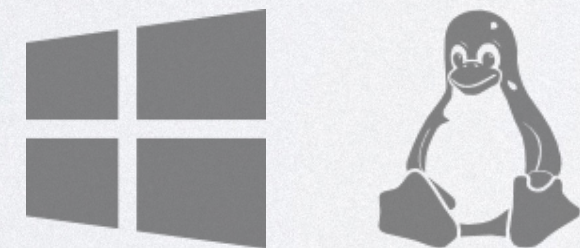
To use our build system and run our documentation locally, you'll need a copy of Bootstrap's source files and Node. Follow these steps and you should be ready to rock:

1. Download and install Node.js, which we use to manage our dependencies.
2. Navigate to the root `/bootstrap` directory and run `npm install` to install our local dependencies listed in `package.json`.
3. Install Ruby, install Bundler with `gem install bundler`, and finally run `bundle install`. This will install all Ruby dependencies, such as Jekyll and plugins.
 - **Windows users:** Read [this guide](#) to get Jekyll up and running without problems.

Build local documentation



Zeal



Dash



Bootstrap

— START WITH SOURCE FILES —



IN A ROCKET

Learn front-end development at rocket speed