Your First Node App: Make A Twitter Bot

Emily Aviva Kapor-Mater

Published with GitBook

Table of Contents

Introduction

Your First Node App:
Build A Twitter Bot

Introduction

Have you ever wondered what the big deal is with Node?

This book grew out of a tutorial I wrote after developing some of my own Twitter bots.

About This Book

This book was written by Emily Aviva Kapor-Mater (website, github, twitter). It was last substantively updated 11 December 2015.

Getting Set Up

Chapter 1: Getting Set Up

Tools You Will Need

In this tutorial, I assume you know how to use a text editor like Atom, and are at least somewhat comfortable with the terminal command line.

If you haven't got it already, you should install Node.

You should have an account (free or otherwise) at Heroku, and you should download and install the Heroku toolbelt.

A GitHub account is a very good idea so you can store and show off your code, but is not required. You do, however, need to have Git installed on your computer. (It comes by default with the Mac and most Linux versions.)

Preliminary setup

We are writing our bot in Node, which is a runtime for JavaScript. We will be doing version tracking with Git. We need to initialize our new application by making an empty directory, making an empty Git repository, and then using `npm` (the Node package manager) to start a new blank app.

```
mkdir my-first-node-app
cd my-first-node-app
git init
npm init
```

For the moment, it's okay to just hit "enter" if you don't know what to write when `npm init` prompts you for input. We'll come back to it later. For now, we'll end up with a file in the root level of our project directory called `package.json`, which is a basic configuration file for our new Node application, and a new directory called `node_modules`, which will contain **dependencies**: third-party packages that we will use, basically, to avoid having to write everything ourselves.

We'll also want to create a file called `.gitignore`, which tells Git not to track everything listed in it. The only thing here we don't want to track is the `node_modules` directory that will store our app's dependencies. We can create the `.gitignore` file from our text editor, or we can create and write it from the command line:

```
echo "node_modules" >> .gitignore
```

Making a Tweet Generator

Chapter 2: Making a Tweet Generator

Introducing libraries

Before we can actually tweet anything, we need to have a tweet generator: otherwise, we'd just be tweeting lots of blank tweets. The generator itself is actually not going to be part of the bot app itself, but instead it will constitute a separate **library**: a script with reusable code.

Let's set up a directory called `lib` in our project directory and make a file for our generator inside it. We'll also need another library of words and phrases for our generator, so let's make that too.

```
mkdir lib
touch lib/generate.js lib/dictionary.js
```

The most important thing our generator program will do is return a string, which we can then use as the body of a tweet. While we're setting it up, let's also have it pull in our dictionary, even though there's nothing in it yet.

```
// generate.js
'use strict';

var dictionary = require('./dictionary');

function generate() {
  return 'Hello, world!';
}

module.exports = generate;
```

The last line tells Node that when this file is imported from another program with the `require()` command, we want it to make the `generate()` function available. (We don't put `()` after the function name here, because we want to export the function itself; not the *execution* of that function: a subtle but crucial distinction.) So, as you might be guessing, we will need to have a similar `module.exports` statement in our dictionary library as well:

```
// dictionary.js
'use strict';

var dictionary = {};

module.exports = dictionary;
```

Generating a random string

Right now, our dictionary is simply an empty object. Let's fill it with some words. But which words? I know! Let's fill it with some words that will help us build up a bot to generate pretentious food truck names.