

Ember.js

A quick introduction



For this course

- For more about Ember:
<http://emberjs.com/>
- For course materials (feel free to clone!):
<https://github.com/thatdevgirl/ember-intro>

Ember:

A Javascript
framework for
creating web
applications.

Framework:

A thing that describes
the structure of how
you should write your
code.

Model:

Responsible for how your data is structured.

View:

Responsible for what is displayed on your page / application.

Controller:

Responsible for passing information between the model and the view.

Setting up Ember

Prerequisites

Ember requires **Node** and **Bower**.

- **Node:** <https://nodejs.org/en/download/>
> `node -v`
- Install **bower**:
> `sudo npm install -g bower`

We'll also be using **npm** quite a bit to install dependencies: <https://www.npmjs.com/>

Install Ember

- **ember-cli** is Ember's *command line interface*.
> `sudo npm install -g ember-cli`

Let's build something!

(and by something, I mean a blog)

Let's get started

- > ember new {{projectName}}
- > ember server
- Go to <http://localhost:4200>
- Check the console: ember 1.13.x = (ノ^_^ノ)
- Update bower.json
- > bower install
- > ember server
- Check the console: ember 2.x = (✿^_^)

Template:

Responsible for rendering the page.

Route:

Responsible for sending data to the template.

Model:

Responsible for how your data is structured.

Component:

Reusable part of the web app.

Making an app

- Create index route:
> `ember generate route {{name}}`
- Create model:
> `ember generate model {{name}}`
- Create components:
> `ember generate component {{name}} —pod`

Connecting to a backend

We're going to use Firebase for the application backend (<https://www.firebase.com/>).

- > `ember install emberfire`
- Update config; restart ember server