Get started in Web Development

About General Assembly

General Assembly is an educational institution creating a global community of individuals empowered to pursue work they love.

We offer **full and part-time courses, workshops and events** on sought-after 21st-century skills like **web development** and user experience design, data science, product management and digital marketing.

Eric Boyer

Designer & Developer

sproutsocial

YOU!

Agenda

Why?

•

How?

•

What?

•

Where?

Why?

Why?

- Make ideas come to life

- Communication

- Customize Websites

- Help Others

- Leads to other tech related interests

Why?

- **Jobs** that offer

  - Professional Development

  - Fascinating Problems

  - Flexibility

  - Stability

  - Opportunities to utilize your other skill sets

"I think everybody in this country should learn how to program a computer because it teaches you how to think."

Steve Jobs

APPLE

How?

How?

- A web browser (**Google Chrome, Firefox**)

- A text editor ***Not Microsoft Word***

  - Notepad (Windows)

  - Notepad++

  - Sublime Text

  - VS Code

How?

- Git - A versioning system that lets you and other collaborators check-in/check-out/keep track of code changes.

- A GitHub account - GitHub is a service and nice interface to use Git (anything in the open source community is public, but also has pricing for hosting private repositories).

- A domain name (gandi, name.com).

- Web Hosting (simple: DreamHost).

- App Hosting (advanced: Heroku).

What?

What?

- **What is the web?**

  - A network of inter-linked specially formatted documents located on individual computers that are visible to other computers around the world.

- **What is a web page?**

  - A document (or many) written in a language that a "browser" is able to read and present.

What?

- **"Front End"**

  - **HTML**: Content and Structure

  - **CSS**: Style and Layout

  - **JavaScript**: Action and Behavior *(*not* Java)*

Everything you see and interact with when using the web is HTML, CSS, and JavaScript.

What?

- "Front End"

  - **HTML**: Noun

    - **<p>Hi</p>** means "paragraph"

  - **CSS**: Adjective

    - **p { color: white; }** means "paragraph is white"

  - **JavaScript**: Verb

    - **$("p").on("click", function(){ $(this).hide(); });**
      means "When you click the paragraph, it will hide"

What?

- **"Front End"**

  - **HTML**: Noun, Structure, Building Blocks

```
1  <div>
2    <h1>Lunch</h1>
3    <p>We've got some great lunch options for you.</p>
4    <ul>
5      <li>Taco</li>
6      <li>Burrito</li>
7      <li>Tostada</li>
8    </ul>
9  </div>
```

What?

- "Front End"

  - **CSS**: Adjective, Style

```
1   <style>
2 v .menu .menu-header {
3     margin: 0 0 10px 0;
4     font-size: 20px;
5     color: salmon;
6   }
7 v .menu .menu-content {
8     font-size: 14px;
9     color: black;
10  }
11  </style>
12 v <div class="menu">
13    <h1 class="menu-header">Lunch</h1>
14    <p class="menu-content">We've got some great lunch options for you.</p>
15 v   <ul class="menu-items">
16      <li>Taco</li>
17      <li>Burrito</li>
18      <li>Tostada</li>
19    </ul>
20  </div>
```

What?

- "Front End"

  - **JavaScript**: Verb, Behavior

  - Can also import and use "libraries" - files of functions written by other people - so that you don't need to reinvent the wheel.

  - One very commonly used JS library is **jQuery**; jQuery functions are designed to make writing JS easier by having functions act like built-in JavaScript functionality, but have simpler syntax.

```
1  <style>
2  .menu .menu-header {
3    margin: 0 0 10px 0;
4    font-size: 20px;
5    color: salmon;
6  }
7  .menu .menu-content {
8    font-size: 14px;
9    color: black;
10 }
11 </style>
12 <script>
13   $(".js-menu-item").on("click", function(){
14     $(this).remove();
15   });
16 </script>
17 <div class="menu">
18   <h1 class="menu-header">Lunch</h1>
19   <p class="menu-content">We've got some great lunch options for you.</p>
20   <ul class="menu-items">
21     <li class="menu-item js-menu-item">Taco</li>
22     <li class="menu-item js-menu-item">Burrito</li>
23     <li class="menu-item js-menu-item">Tostada</li>
24   </ul>
25 </div>
```

What?

- "Front End" Challenges

  - **Device Resolutions** (Retina, 4K, Low Resolutions)

  - **Device Sizes** (Phone, Tablet, Desktop)

  - **Browsers** (Firefox, Chrome, Safari, Internet Explorer 9, 10, 11, Microsoft Edge)

  - **Performance**

http://www.csszengarden.com/

http://codepen.io/

What?

- **"Back End"**

  - Infrastructure

  - Data Management

  - Data Processing / Receiving / Sending

  - Servers

  - PHP, Ruby/Rails, Python/Django, C

What?

- "Back End"

  - Ruby

    - A programming language, similar to JS.

    - It has variables, logical tests and methods/functions.

    - Open source community with libraries (pre-written code to solve common problems and an active fast moving community) called Gems.

What?

- "Back End"

  - Rails

    - A '**gem**' or library originally created by the Basecamp / 37 Signals team.

    - Ruby is a language, but Rails is a "web application framework" built IN Ruby, that is used to efficiently create web applications.

    - Think of Rails as being a template or a good starting point to do common things like: Create, Read, Update, Delete, Navigate to a URL and have a web page returned, etc.

What?

- **"Back End"**

  - **Rails - MVC**

    - **Model** - Abstraction of the data that your application stores

    - **View** - Visual result, in HTML/CSS/JS that the browser can understand and present

    - **Control** - Logic that tells the application how to behave

What?

- "Back End" but, on the front?

  - **Angular**, **Node**, **Backbone**, **React** are all frameworks in some ways to Rails, but built in JS.

  - They also perform back-end operations, but because they're written in JavaScript, some of those operations can be run in the browser, which can make the application faster & more efficient (depending on the project).

What?

- **"Back End" API**

  - **A**pplication **P**rogramming **I**nterface

  - *Can be* publicly visible interface to the application that other applications can utilize.

  - Twitter API

What?

- **Terminology**

  - **Clients** vs. **Servers**

What?

- **Terminology**

  - **Clients** vs. **Servers**

  - **IP Addresses** (Internet Protocol Address)

What?

- **Terminology**

  - **Clients** vs. **Servers**

  - **IP Addresses** (Internet Protocol Address)

  - **DNS** (Domain Name Service)

What?

- **Client**: "Hey, I need directions to www.facebook.com"

- **DNS Server**: "Oh, hey there. www.facebook.com maps to 31.13.71.14"

- **Server (31.13.71.14)**: "You rang? Here's the facebook.com app i've got."

What?

- **Terminology**

  - **CDN** (Content Delivery Network)

https://www.facebook.com/

Where?

Online Resources

- w3schools.com

- learn.shayhowe.com/html-css

- stackoverflow.com

- teamtreehouse.com

- Dash (General Assembly)

General Assembly

- **FEWD** (<u>Front End Web Development</u>)

  - **March 6th - May 10th**

  - Mon & Wed

  - 6pm - 9pm

  - $3,950

  - with me!

General Assembly

- **WDI** (Web Development Immersive)

  - **May 22nd - August 18th**

  - Mon through Friday

  - 9am - 5pm

  - $13,500

General Assembly

- **Chicago Campus**

Q&A

Thanks!

Eric Boyer

Twitter: @eboyer

Email: eboyer@gmail.com