



**WELCOME TO  
FRONT-END  
BOOTCAMP!**

# INTRODUCTION

## JESEEKIA VAUGHN

[jeseekia@grandcircus.co](mailto:jeseekia@grandcircus.co) | [@metadevgirl](https://twitter.com/metadevgirl) | [jeseekia](#)

GRAND  
CIRCUS  
DETROIT

# INTRODUCTION

## DAVID WOLVERTON

david@grandcircus.co | @dwolverton | dwolverton

# INTRODUCTION

## ADAM DAVID

adam@grandcircus.co | @adamAtGC



# COURSE EXPECTATIONS

What we expect from you...

CIRCUS  
DETROIT

GRAND  
CIRCUS  
DETROIT

# BE PRESENT

Listen, soak up the information (there's a lot of it),  
process it.

GRAND  
CIRCUS  
DETROIT

# ASK QUESTIONS

I'm not big on formality. Feel free to ask any question that comes to mind. I'll also stop frequently for questions.



# SUPPORT EACH OTHER

I'm a big proponent of pair programming. Any in-class exercise or project work may be paired and certain labs *will* be paired or in groups. Larger final projects will be group projects.



# HAVE FUN

See previous comment on my stance on formality and ceremony. Bored people learn nothing. I'll do what I can to make this fun as well as instructive.

GRAND  
CIRCUS  
DETROIT

GRAND  
CIRCUS  
BEER CO.

GRAND  
CIRCUS

# OUR GOALS FOR YOU

# OUR GOALS FOR YOU

Inculcate you with vital tools such as version control, specifically git and github.



# OUR GOALS FOR YOU

Initiate you into the world of JavaScript.



JS

DETROIT

GRAND  
CIRCUS  
DETROIT

DETROIT

GRAND  
CIRCUS  
DETROIT

# OUR GOALS FOR YOU

Learn the basics of web and mobile architecture by  
seeing it in practice.

GRAND  
CIRCUS  
DETROIT



# COURSE OUTLINE

(What are we doing here anyway?)

# 'THE PLAN'

- Week 1 - HTML / CSS / JavaScript
- Week 2 - JavaScript / TDD
- Week 3 - jQuery / AJAX / JSON
- Week 4 - Angular JS / TDD
- Week 5 - Angular JS / Node JS
- Week 6 - Node JS
- Week 7 - Project
- Week 8 - Project / Demo Day

# THE PLAN

- Each week will be split among a number of lectures focused around a weekly topic.
- There will be practical exercises every day.
- Each week will end with a *tools day* where we learn to use the tools that modern front end developers use.

# PRACTICAL WORK

The practical work is divided into two categories

- *Exercises* are short & focused to practice a specific topic or technique
- *Labs* are broader in scope and incorporate more of the material into a single problem.

CIRCUS  
DETROIT

CIRCUS  
DETROIT

CIRCUS  
DETROIT

GRAND  
CIRCUS  
DETROIT

# YOUR NEW JOB

GRAND  
CIRCUS  
DETROIT

GRAND  
CIRCUS  
DETROIT

GRAND  
CIRCUS  
DETROIT

GRAND  
CIRCUS  
DETROIT

GRAND  
CIRCUS

GRAND  
CIRCUS  
DETROIT

GRAND  
CIRCUS  
DETROIT

GRAND  
CIRCUS



# YOUR NEW JOB

As of right now, this class is your new job and I am your boss. The hours are 9am - 5pm Monday through Friday with some work in the evenings and weekends required, and as much optional work as you can stomach.

CIRCUS  
DETROIT

GRAND  
CIRCUS  
DETROIT

CIRCUS  
DETROIT

GRAND  
CIRCUS  
DETROIT

# YOUR NEW JOB

Your primary job responsibility is to

# FIGURE IT OUT

GRAND  
CIRCUS  
DETROIT



# YOUR NEW JOB

Developers are essentially well-paid problem solvers.  
If you come across something you're not sure about,  
you need to make an effort to solve the issue  
yourself.



# YOUR NEW JOB

A checklist for when you get stuck:

1. *First*, ask google.
2. *Second*, ask a fellow bootcamper.
3. *Third*, ask another fellow bootcamper.
4. If you're *still*/stumped, ask Instructor or the TA.

# YOUR NEW JOB

The rule about the checklist:

Don't be a hero and spin your wheels for hours without any progress. If you're genuinely stuck, ask for help and keep asking until you get it.





# YOUR NEW JOB

One final thing...

Because we want you to learn how to diagnose problems, we are going to eventually pull back from helping in certain ways as the course progresses.

# YOUR NEW JOB

Starting at the end of week 2 if your console is displaying syntax errors or files not found(such as scripts, CSS sheets, images not loading), we will ask you to run through your code where errors are found and fix it.

# YOUR NEW JOB

*Most importantly*

## PRACTICE!!

We won't always have homework for this class, but any extra time you spend working on this stuff will only deepen your understanding of the material. If you finish an assignment early, look for ways to go a step further. We have *loads* of resources for extra work.

***YOU WILL GET OUT OF THIS WHAT YOU PUT INTO IT***

# GETTING STARTED

**Note** that some of this should already be done. This is mostly a checklist of things we will need for smooth(ish) sailing in this course.

# GET CONNECTED

wifi: Madison

pw: Bro@dw@y#



# CHECKLIST

- Google Chrome
- Sublime Text / Atom.io
- Slack
- Github / git
- NodeJS

# QUESTIONS?

GRAND  
CIRCUS  
DETROIT



GRAND  
CIRCUS

DETROIT

GRAND  
CIRCUS

DETROIT

GRAND  
CIRCUS

DETROIT



GR  
CIR

DET

GRAND  
CIRCUS

DETROIT

GRAND  
CIRCUS

DETROIT



GRAND  
CIRCUS

DETROIT

GRAND  
CIRCUS

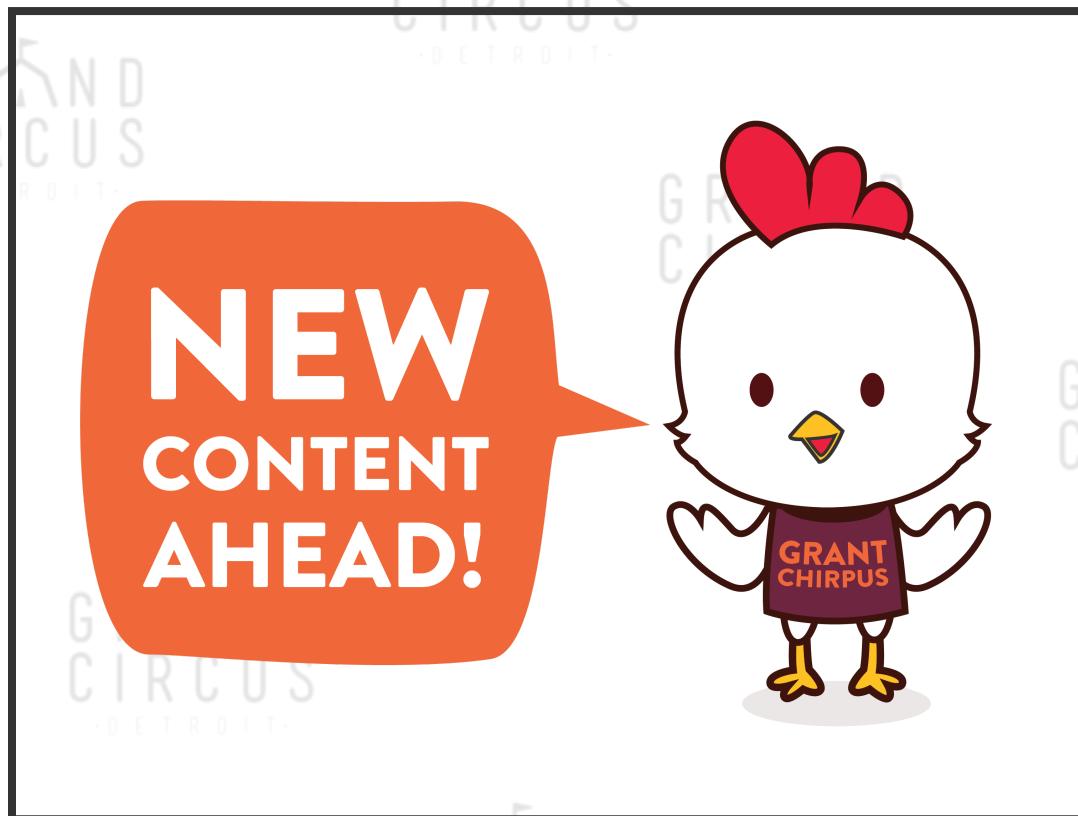
DETROIT

GRAND  
CIRCUS

DETROIT

GRAND  
CIRCUS

DETROIT



# GOALS FOR THIS UNIT

1. The Website as Communication
2. Introduction to HTML & CSS
3. HTML 101
4. **Exercise:** Build a static HTML site from scratch

# ASSOCIATED READING

From HTML & CSS:

- Introduction: 2-11
- Chapter 1: 12-39
- Chapter 2: 40-61
- Chapter 3: 62-73
- Chapter 4: 74-93
- Chapter 5: 94-119
- Chapter 6: 126-143
- Chapter 8: 176-199

Every website on the internet uses HTML & CSS.





...most of them use JavaScript as well, in one form or another.



# MEET YOUR NEW BEST FRIEND!

It's called the Dev Tools Inspector

(right click in the browser

and select **Inspect Element**)

The screenshot shows the browser's developer tools with the "Elements" tab selected. The DOM tree on the left displays the HTML structure of a presentation slide. The right panel shows the "Styles" tab, which lists the CSS rules applied to the selected element, along with the "Computed" and "Event Listeners" tabs.

```
<!DOCTYPE html>
<html lang="en">
  <head></head>
  <body style="transition: -webkit-transform 0.8s ease; -webkit-transition: -webkit-transform 0.8s ease;">
    <!-- Any section element inside of this container is displayed as a slide -->
    <div class="reveal convex center slide" role="application" data-transition-speed="default" data-background-transition="fade">
      <!-- EN SLIDES DIV -->
      <div class="slides" style="width: 960px; height: 700px; zoom: 1.00028571428571;"></div>
      <div class="backgrounds"></div>
      <div class="progress" style="display: block;"></div>
      <aside class="controls" style="display: block;"></aside>
      <div class="slide-number"></div>
      <div class="pause-overlay"></div>
      <div id="aria-status-div" aria-live="polite" aria-atomic="true" style="position: absolute; height: 1px; width: 1px; overflow: hidden; clip: rect(1px 1px 1px 1px);">DEMO!
        (right click in the browser and select Inspect Element)
      </div>
      <div id="js/head.min.js"></script>
      <script src="js/reveal.js"></script>
      <script>
        // Full list of configuration options available at:
        // https://github.com/hakimel/reveal.js#configuration
        Reveal.initialize({
          controls: true,
          progress: true,
          history: true,
          center: true,
          transition: 'slide', // none/fade/slide/convex/concave/zoom
          // Optional reveal.js plugins
          dependencies: [
            { src: 'lib/js/classList.js', condition: function() { return !document.body.classList; } },
            { src: 'plugin/markdown/markdown.js', condition: function() { return !!document.querySelector(' [data-markdown]'); } },
            { src: 'plugin/markdown/markdown.js', condition: function() { return !!document.querySelector(' [data-markdown]'); } },
            { src: 'plugin/highlight/highlight.js', async: true, condition: function() { return !!document.querySelector(' pre code' ); }, callback: function() { hljs.initHighlightingOnLoad(); } },
            { src: 'plugin/zoom-js/zoom.js', async: true },
            { src: 'plugin/notes/notes.js', async: true }
          ]
        });

        </script>
        <script type="text/javascript" src="plugin/markdown/markdown.js"></script>
        <script type="text/javascript" src="plugin/markdown/markdown.js"></script>
        <script type="text/javascript" src="plugin/zoom-js/zoom.js"></script>
        <script type="text/javascript" src="plugin/notes/notes.js"></script>
      </body>
    </html>
```

Styles Computed Event Listeners »

element.style { } user agent stylesheet script { display: none; } Inherited from body body { reveal.css:30 position: relative; line-height: 1; background-color: #ffff; color: #0000; }

html, body, .reveal div, .reveal span, .reveal applet, .reveal object, .reveal iframe, .reveal h1, .reveal h2, .reveal h3, .reveal h4, .reveal h5, .reveal h6, .reveal p, .reveal blockquote, .reveal pre, .reveal a, .reveal abbr, .reveal acronym, .reveal address, .reveal big, .reveal cite, .reveal code, .reveal dfn, .reveal em, .reveal img, .reveal ins, .reveal kbd, .reveal q, .reveal s, .reveal samp, .reveal small, .reveal strike, .reveal strong, .reveal sub, .reveal sup, .reveal tt, .reveal var, .reveal b, .reveal u, .reveal center, .reveal dl, .reveal dt, .reveal dd, .reveal ol, .reveal ul, .reveal li, .reveal fieldset, .reveal form, .reveal label, .reveal legend, .reveal table, .reveal caption, .reveal tbody, .revealtfoot, .reveal th, .reveal td, .reveal article, .reveal aside, .reveal canvas, .reveal details, .reveal embed, .reveal figure, .reveal figcaption, .reveal footer, .reveal header, .reveal output, .reveal ruby, .reveal section, .reveal summary, .reveal time, .reveal mark, .reveal audio, video { margin: 0; padding: 0; border: 0; font-size: 100%; font-weight: inherit; vertical-align: baseline; }

Inherited from html html, body, .reveal div, .reveal span, .reveal applet, .reveal object, .reveal iframe, .reveal h1, .reveal h2, .reveal h3, .reveal h4, .reveal h5, Find in Styles



# TERMINOLOGY

# TERMINOLOGY

Some key terms or phrases that are used as a matter of course in the software industry. You may know some or all of these, or you may have heard the terms but be unclear about their actual meaning. They are common jargon among developers

# WEB DEVELOPMENT

Web development is a broad term for the work involved in developing a web site for the Internet. In industry parlance, 'web development' usually refers to the more code-related tasks such as programming JavaScript, coding HTML and CSS. It can even extend to tasks related to the back end server infrastructure such as creating web services and handling business logic for a company or product.



CIRCUS  
DETROIT



CTR  
DET

GRAND  
CIRCUS  
DETROIT

# WEB DESIGN

The process of planning & structuring a website; specifically, the visual aspects and assets for the site. Recently, this job description has also begun to include interaction design. That is, designing the user experience (UX), information architecture, and the flow of the application or site.

# WEB SITE

A largely informational web page. While they may include dynamic elements and react to user inputs. The general purpose of a web site is to provide information about a person, business, product, or service.

# WEB APPLICATION

A more recent term to indicate a web site whose sole purpose is not just informational, but rather functional. Web applications have become robust enough to do everything from our taxes, manage our personal calendars, or even do standard desktop publishing tasks.

# FRONT END

The visible and interactive parts of a website or application.



# BACK END

The 'invisible' or inner functionality of a website or application. Examples include costly calculations, interacting with a database or making use of web service end points. While we're at it...

# WEB SERVICE

A software function provided at a network address over the Web. The W3C defines a Web service generally as:

---

*"A software system designed to support interoperable machine-to-machine interaction over a network."*

---

# DATABASE

A software system for storing data long term. This is also sometimes referred to as a 'persistance layer'.

# VERSION CONTROL

A software tool for managing changes to a set of files, website, application or any collection of files and for reconciling the differences between those files when conflicts emerge.

# APPLICATION PROGRAMMING INTERFACE (API)

An API is a set of routines, protocols, and tools for building software applications. An API expresses a software component in terms of its operations, inputs, outputs, and underlying types. An API defines functionalities that are independent of their respective implementations, *which allows definitions and implementations to vary without compromising each other.*



# QUESTIONS?



# NESTING

HTML elements 'nest' inside of one another. The element that opens first closes last.



# CONTENT TAGS

# COMMON CONTENT TAGS

Tag	Description
<code>div</code>	defacto container element
<code>p</code>	used for body copy
<code>h1 thru h6</code>	designating titles/subtitles
<code>ol</code>	create a numbered list
<code>ul</code>	create an unordered list
<code>li</code>	list elements
<code>a</code>	link to other pages or sites
<code>link</code>	import external documents

# SEMANTIC HTML5 TAGS

class	Description
section	Container tag used for page organization
header	Container for introductory and navigational stuff
footer	Container for footer content (site map, internal links, etc.)
nav	Container for a major block of navigation links
audio	Multimedia tag for playing audio files
video	Multimedia tag for playing video files
canvas	element can be used to draw graphics via JavaScript

# HTML COMMENTS

Like any other good coding language, HTML offers comments. They operate like comments in any other language. They are ignored by the browser engine.

```
<!-- Hello, I am a comment. -->
```

# TABLES

## A NOTE ON TABLES

You may be thinking it now or you may think later that tables would be a great way to position content for site...



-DETROIT-



Don't do that. Just... don't. Trust me on this. It's a bad idea. It will give us an ulcer.

GRAND  
CIRCUS  
DETROIT

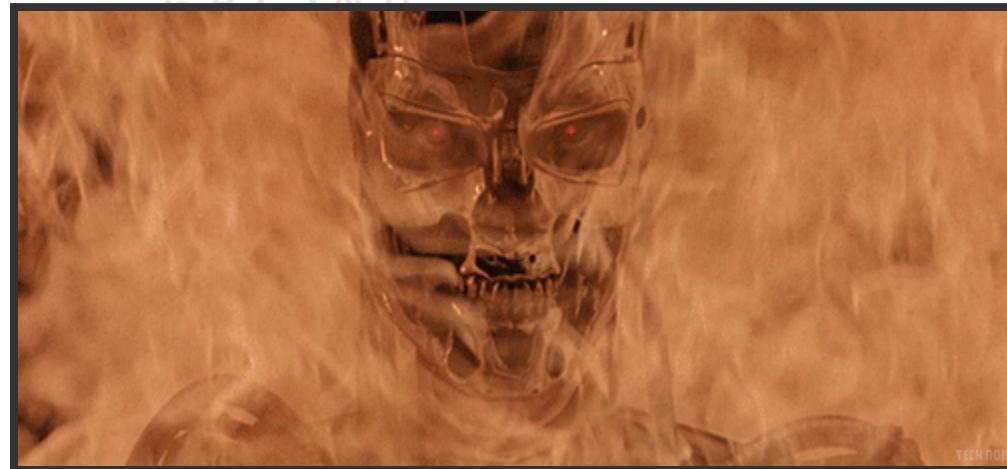
CIRCUS  
DETROIT

GRAND  
CIRCUS  
DETROIT

GRAND

GRAND  
CIRCUS

GRAND



GRAND  
CIRCUS  
DETROITGRAND  
CIRCUS  
DETROITGRAND  
CIRCUS  
DETROIT

# WHAT'S WRONG WITH THIS CODE?

Look at the following examples and tell me what is wrong with the code.

GRAND  
CIRCUS  
DETROIT

GRAND  
CIRCUS

GRAND  
CIRCUS  
DETROIT

GRAND  
CIRCUS

```
<html>
  <head>
    <body>
      </head>
      </body>
    </html>
```

```
<html>
  <head>
    <title>The Best Site Evar!!
  </head>
  <body>

    <p>Check out this riveting content!</p>

  </body>
</html>
```

GRAND  
CIRCUS  
DETROIT

GRAND  
CIRCUS

< p style=hotStuff>Check out this riveting content!</p>

CIRCUS  
DETROIT

CIRCUS  
DETROIT

CIRCUS  
DETROIT

GRAND  
CIRCUS

GRAND  
CIRCUS  
DETROIT

GR  
CIR  
DET

GR  
CIR  
DET



# FOLDER STRUCTURE

This stuff is not exciting but it's *important*.



CIRCUS  
DETROIT



GRAND  
CIRCUS  
DETROIT



# FOLDER STRUCTURE

The Rules of *Threes*

# RECAP

You should understand and be able to use:

- HTML elements
- Proper nesting
- HTML Comments
- Correct folder structure