

# WiFi Connection

Network: atomic-detroit

Password: cOrrectHOrse



# Beginning HTML and CSS

## Session 1

# Welcome!

Girl Develop It is here to provide affordable and accessible programs to learn software through mentorship and hands-on instruction.

## Some "rules"

- We are here for you!
- Every question is important
- Help each other
- Have fun

# Meet Your Instructor

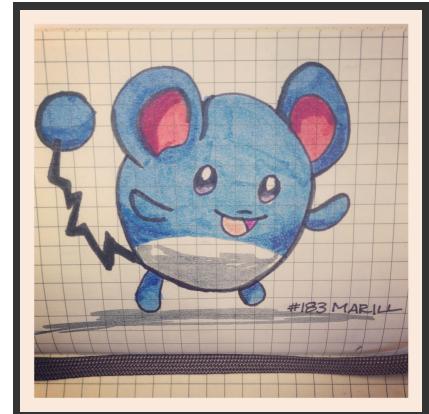
**Cara Jo Miller**

- Pixel Wrangler at [Are You a Human](#)
- Founder of [Run This Race](#)
- Leader at [Refresh Detroit](#)
- Co-organizer of  
[WordCamp Detroit](#)
- Twitter: [@corrinajo](#)
- E-mail: [corrinajo@gmail.com](mailto:corrinajo@gmail.com)

# Meet Your Instructor

## How I got started with the web!

- 1996: Built first computer from spare parts in my Dad's shop.
- 1997: Got bored on a snow day and decided to learn HTML from [Lissa Explains it All](#)
- 1997: Big into Pokémon and drawing. Coded first site to showcase Pokémon drawings to family members and friends.
- 1998-2012: Decided that the internet was the coolest thing in the world and decided to make a career out of it.



# Now it's all about you!

Tell us about yourself.

- Who are you?
- What do you hope to get out of the class?
- What's your new year's resolution?!



# What is HTML?

HTML is the code that allows us to build websites

The screenshot shows the homepage of the Girl Develop It website. At the top left is the logo "Girl Develop It" with the tagline "don't be shy. develop it." Below the logo are social media icons for Twitter and Facebook. At the top right are navigation links: CLASSES, CHAPTERS, EVENTS, RESOURCES, and PARTNERS. The main banner features a photograph of people working at laptops, with the text "Empowering women of diverse backgrounds from around the world to learn how to develop software". Below the banner are three circular icons: a teal one labeled "CLASSES" containing a laptop icon, a green one labeled "COMMUNITY" containing two stylized white leaf or paintbrush-like icons, and a yellow one labeled "RESOURCES" containing a wrench icon. At the bottom of the page are two buttons: "CITIES WHERE WE ARE MAKING AN IMPACT" and "UPCOMING EVENTS".

# What is HTML?

If you 'view the source', you see this

```
33 </li>
34 <li>
35 <a href="http://www.facebook.com/girldvelopit" alt="Girl Develop It on Facebook" target="_blank"></a>
36 </li>
37 </ul>
38 <div class='header'>
39 <a href="/"></a>
40 <span id='menu'>
41 <nav class='menu' id='menu'>
42 <ul>
43 <li id='about'>
44 <a href="/partners">PARTNERS</a>
45 </li>
46 <li id='materials'>
47 <a href="/materials">RESOURCES</a>
48 </li>
49 <li id='events'>
50 <a href="/events">EVENTS</a>
51 </li>
52 <li id='chapters'>
53 <a href="/chapters">CHAPTERS</a>
54 </li>
55 <li id='classes'>
56 <a href="/classes">CLASSES</a>
57 </li>
58 </ul>
59 </nav>
60
61 <div class='clear'></div>
62 </span>
63 </div>
64 </header>
65
66 <section id='page'>
67 <div id='page-container'>
68 <div id='body-content'>
69 <div class='topics'>
70 <div class='overlay'></div>
71 <div class='overlay-div'>
72 <p class='tagline'>
73 Empowering women of diverse backgrounds from around the world to learn how to develop software
74 </p>
75 <ul class='topic-buttons'>
76 <li class='topic-button'>
```

# History of HTML

- Invented by Tim Berners-Lee
- Created "hypertext" to share scientific papers
- First web page August 6, 1991
- Standardized by w3 Consortium (pack of super nerds)

# History of HTML

- HyperText Markup Language
- Early 90s
- HTML 4 in 1997
- XHTML in 2000
- HTML 5 in 2008

# Terms

- **Web design**  
The process of planning, structuring and creating a website
- **Web development**  
The process of programming dynamic web applications
- **Front end**  
The outwardly visible elements of a website or application
- **Back end**  
The inner workings and functionality of a website or application.

# Tools

- Browser  
Chrome  
Firefox
- Development Toolkit  
Chrome - Inspector  
Firefox - Firebug
- Text Editor  
TextWrangler - Mac  
Notepad ++ - Windows  
**Sublime Text - Mac or Windows**

# Get Started: Sublime Text 2

**We'll be using Sublime Text 2 in class today.**

1. Download ST2 from  
[www.sublimetext.com/2](http://www.sublimetext.com/2)
2. Turn off Auto Correct & Tab Completion

Modify your the preferences / settings by adding these lines to the code:

- "tab\_completion": false
- "auto\_complete": false,

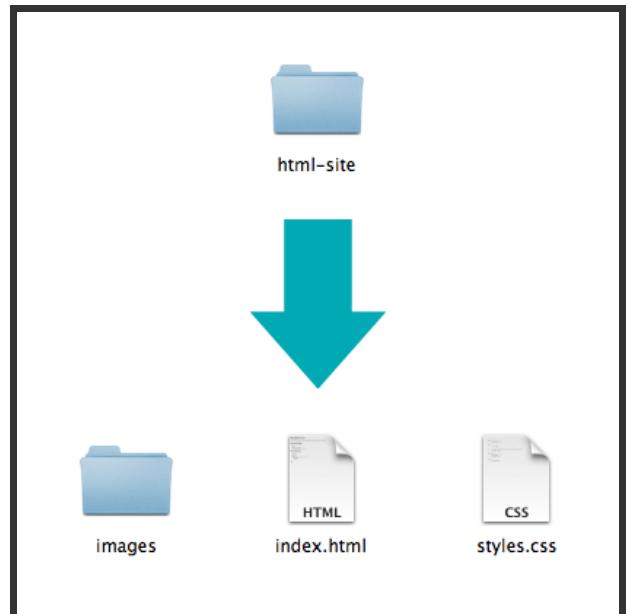
# Get Started: Folder Structure

All the files for your site should be stored within the same folder.

## This includes:

- HTML Files
- CSS Files
- Images
- Script files
- Anything else that will appear on your site

Note: File names should not include spaces or special characters. File names ARE case sensitive.



# What we'll be building today

Today we will be learning how to code a site from scratch using paragraphs, headings, links, images, and lists.

## Welcome to my first site!

Welcome to Girl Develop It! Today we will be making a simple site that has headings, paragraphs, images, and links.

### The Basics

At first, it won't seem like much, but these are the building blocks of web development. A site cannot exist without these very basic elements.

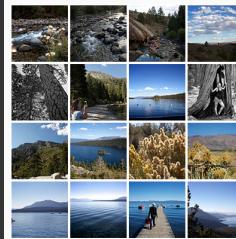
*Images add interest to any website (even this one)*

My best friends and I having lunch on Lake Tahoe!



**Links take users to interesting (or not so interesting) new places!**

I took over [400 photos](#) in Lake Tahoe, but I only uploaded [33 of my favorite photos to Flickr](#).



Lists help organize content better

My **favorite** things about Lake Tahoe:

1. The Emerald Bay 1/2 Marathon
2. Switchback roads
3. Seeing the mountains
4. Making new friends
5. Dancing with professional GoGo Dancers

Send me a message and I'll tell you all about my trip.

Line breaks help break up long lines of text.

Send me an e-mail at:

[corinajo@gmail.com](mailto:corinajo@gmail.com)

or follow me on [Twitter](#)

# Anatomy of a website

Your Content

+ HTML: Structure  
+ CSS: Presentation  
= Your Website

A website is a way to present your content to the world, using HTML and CSS to present that content & make it look good.

# Anatomy of a website

Concrete example

- A paragraph is your content
- Putting your content into an HTML tag to make it look like a paragraph is Structure

`<p>A paragraph is your content</p>`

- Make the font of your paragraph blue and 18pt is presentation

**A paragraph is your content**

# Anatomy of an HTML element

- **Element**

- An individual component of HTML
- Paragraph, heading, table, list, div, link, image, etc.

- **Tag**

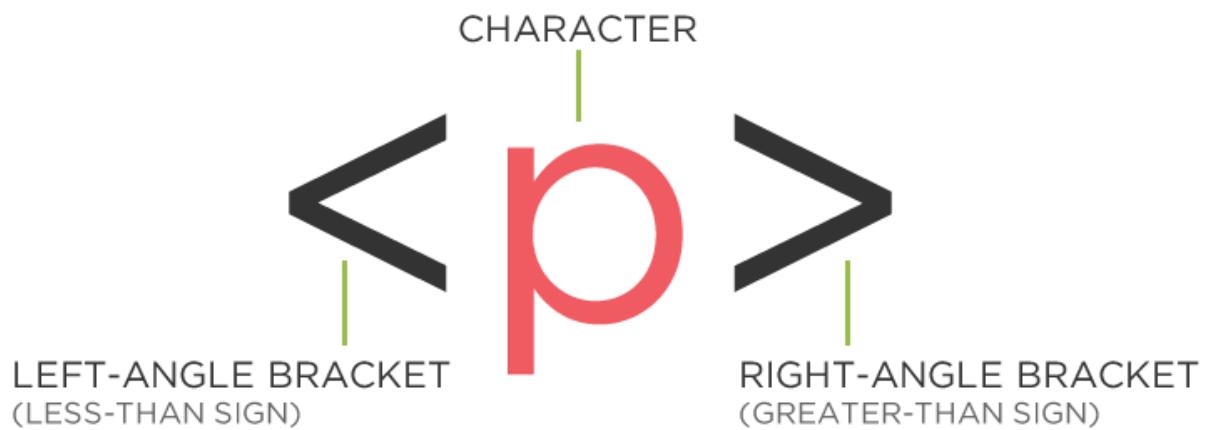
- Marks the beginning & end of an element
- Opening tag and Closing Tag
- Tags contain characters that indicate the tags purpose

```
<tagname>Stuff in the middle</tagname>
```

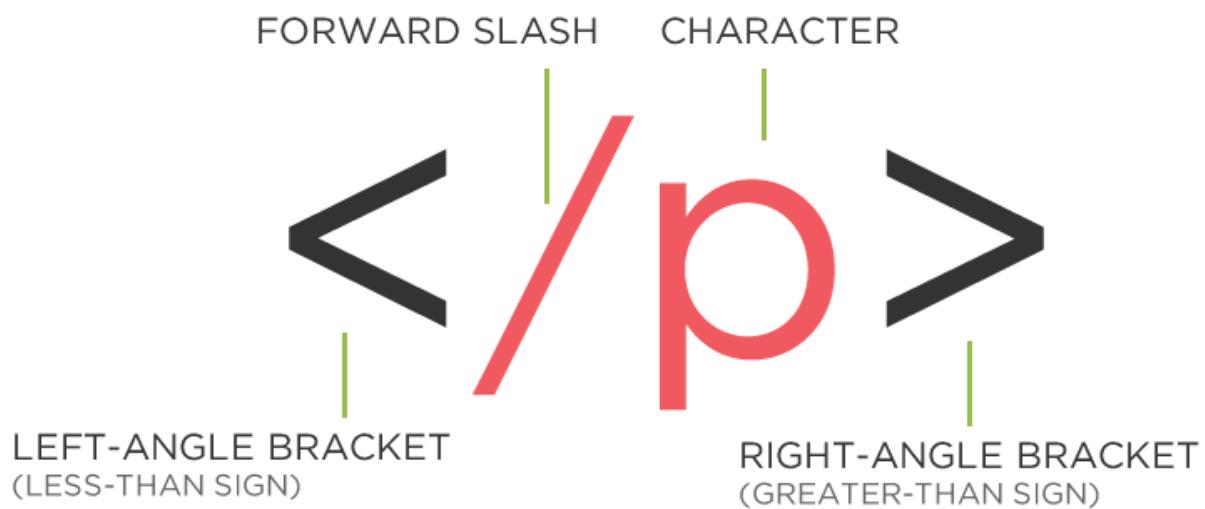
```
<p> This is a sample paragraph.</p>
```

# Tag Breakdown

## OPENING TAG



## CLOSING TAG



# Anatomy of an HTML element

- **Container Element**

- An element that can contain other elements or content
- A paragraph (<p>) contains text

- **Stand Alone Element**

- An element that cannot contain anything else

<br/>

<img/>

# Anatomy of an HTML element

- **Attribute**

- Provides additional information about the HTML element
- Class, ID, language, style, identity, source
- Placed inside an opening tag, before the right angle bracket.

- **Value**

- Value is the value assigned to a given attribute.
- Values must be contained inside quotation marks.

```
<div id="copyright">© GDI 2013</div>

<a href="http://girldevelopit.com">GDI</a>
```

# Doctype

The first thing on an HTML page is the doctype, which tells the browser which version of the markup language the page is using.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML  
4.01 Transitional//EN" "http://  
www.w3.org/TR/html4/loose.dtd">
```

```
<!DOCTYPE html>
```

\* The doctype is case-insensitive.  
DOCtype, doctype, DocType and DoCtYpe are all valid.

# HTML Tag

After <doctype>, the page content must be contained between <html> tags.

```
<!DOCTYPE html>
<html>

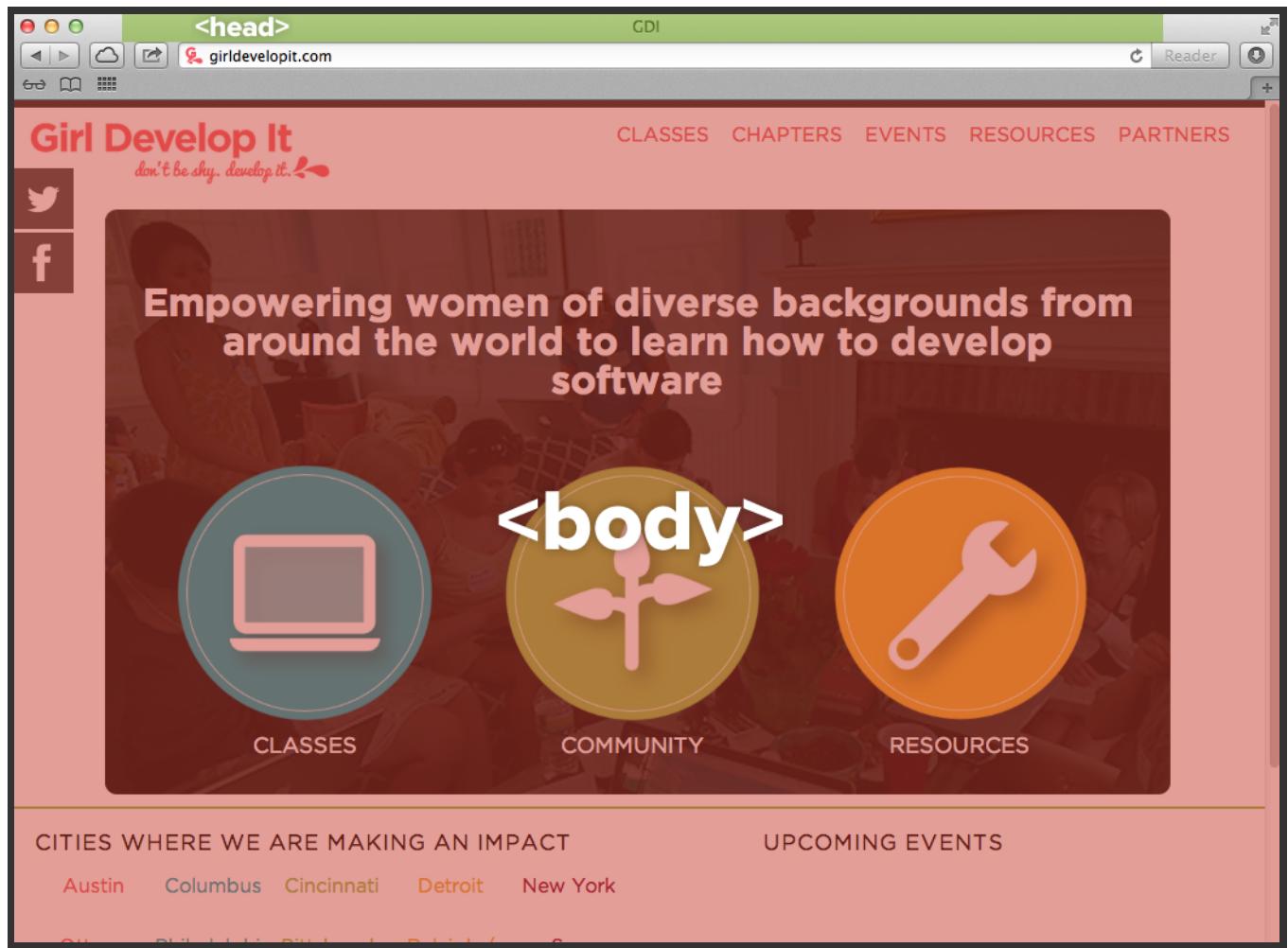
</html>
```

# Head and Body Tags

**Head:** The head contains the title of the page & meta information about the page. Meta information is not visible to the user, but has many purposes One of which is to tell search engines about your page, who created it, and a description.

**Body:** The body contains the actual content of the page. Everything that is contained in the body is visible to the user.

# Head and Body Tags: Example



# Head and Body Tags

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title of the page </title>
  </head>
  <body>
    The page content here.
  </body>
</html>
```

# Let's develop it!

Let's get our web page set up with a doctype, head, title and body.

Later we'll add some content to it!

# Nesting

All elements "nest" inside one another

Nesting is what happens when you put other containing tags inside other containing tags. For example, you would put the `<p>` inside of the `<body>` tags. The `<p>` is now nested inside the `<body>`



Whichever element OPENS first  
CLOSES last

# Nesting: Example

Elements are 'nested' inside the `<body>` tag.

```
<body>
  <p>A paragraph inside the body tag<
/p>
  </body>
```

Paragraphs 'nested' inside list items.

```
<ul>
  <li>
    <p>A paragraph inside a list ite
m</p>
    </li>
  </ul>
```

# Element: Paragraph

Paragraph 1

```
<p>Paragraph 1</p>
<p>Paragraph 2</p>
<p>Paragraph 3</p>
```

Paragraph 2

Paragraph 3

```
<p>Paragraph 1</p> <p>Pa
ragraph 2</p> <p>Paragrap
h 3</p>
```

```
<p>Paragraph 1</p>
```

```
<p>Paragraph 2</p>
<p>Paragraph 3</p>
```

\* White space is only for humans!

# Example: Paragraphs

Paragraphs allow you to format your content in a readable fashion.

Starting with the higher resolution is my preferred way to design apps for Apple devices, as it allows me to set the appearance of each single pixel, and add details which won't be visible on the scaled-down version. In the worst case, if you have to use bitmap textures for some reason, the scaled-down version will look a lot better than a scaled-up one (which will have blurry textures).

When designing in 2x resolution, you have to think a bit more about what you're doing. For example, you have to avoid sizes like 3 or 5 pixels for border widths, and the same applies to effects like drop shadows — a scaled-down blur ratio of 5 pixels would result in 2 or 3 px blur, and neither would be the result you want to achieve when aiming for real perfection.

I have seen other designers doing it the other way: starting with the normal size, and scaling the whole design up when it is done. Either way is absolutely fine — I think it always depends on the designer's personal preference. However, in this article I will only describe the way to start with the 2x resolution graphics first.

\* You can edit how paragraphs are displayed with CSS

# Element: Heading

```
<h1>Heading 1</h1>
<h2>Heading 2</h2>
<h3>Heading 3</h3>
<h4>Heading 4</h4>
<h5>Heading 5</h5>
<h6>Heading 6</h6>
```

**Heading 1**  
**Heading 2**  
**Heading 3**  
**Heading 4**  
**Heading 5**  
**Heading 6**

\* Heading number indicates hierarchy, not size. Think: Outlines from high school papers

# Example: Headings

## From 63% to 99% Profile Completion

HiredMyWay takes the traditional recruiting process online, connecting candidates, effective hiring process.

A Forbes 2012 Top Career Site.

### Treat your users like humans again.

It's just common sense: a fun game is better than a jumble of letters you can't read. That's why 98% of users prefer PlayThru to CAPTCHA. And the other 2%? Well, some people are gluttons for punishment.

#### USE WORDPRESS

It may seem self-evident, but using WordPress for yourself could be publishing your own personal blog or building a member. Find a way to use WordPress. It's a little crazy to to the top of something when you've never experienced it the user.

### I'm Not A Developer! What Can I Do?

You do not need to know how to write code to get involved in WordPress. If you're not a developer, then WordPress is for you today. Here are some of the things you can do:

# Formatted text

```
<p>
  Here is a paragraph with
  <em>Emphasized</em> text
  and <strong>Important</str
  ong> text.
</p>
```

Here is a paragraph with *Emphasized* text and **Important** text.

- \* Notice: em and strong are meant to indicate meaning through style. If you want to have italicized for appearance and not to communicate meaning, you should use CSS.

# Let's Develop it!

Let's add some content to our site!

Add one of each level of heading with 1-2 short paragraphs of text below each heading.

Italic and bold some text within a few paragraphs.

# Element: Link

Links have three components

- Tag: <a></a>
- Href attribute:  
"http://www.girldevelopit.com"
- Alt attribute: "Girl Develop It"

```
<a href ="http://www.girldevelopit.com" alt = "Girl De  
velop It Homepage">GDI</a>
```

GDI

The <a> tag surrounds text or images  
to turn them into links

# Link Attributes

Links can have attributes that tell the link to do different actions like open in a new tab, or launch your e-mail program.

```
<a href="home.html" target="_blank">Link Text</a>
```

Link opens in a new window/tab with **target="\_blank"**

```
<a href="mailto:detroit@girldevelopit.com">E-mail us!</a>
```

Link opens mail program by inserting **mailto:** directly before the email address.

# Relative vs. Absolute paths for links & images

- **Relative**

- Relative paths change depending upon the page the link is on.
  - Links within the same directory need no path information. "`filename.jpg`"
  - Subdirectories are listed without preceding slashes. "`images/filename.jpg`"

- **Absolute**

- Absolute paths refer to a specific location of a file, including the domain.  
`"http://www.girldevelopit.com/chapters/detroit"`
- Typically used when pointing to a link that is not within your own domain.

# Let's Develop It

Let's add links to our site!

Add links that open in the same window, a new window and link to an e-mail address.

# Element: Image

Images have three components

- Tag: <img/>
- Src attribute:  
"http://girldevelopit.com/assets/pink-logo.png"
- Alt attribute: "Girl Develop It logo"

```
<img src ="http://girldevelopit.com/assets/pink-logo.png" alt = "Girl Develop It Logo"/>
```



\* Notice: This tag is our first example of a stand-alone or "self-closing" element.

# Element: Line Break

```
<p>
  Imagine there's no Heave
n <br/>
  It's easy if you try <br
/>
  No hell below us  <br/>
  Above us only sky
</p>
```

Imagine there's no  
Heaven  
It's easy if you try  
No hell below us  
Above us only sky

# Let's Develop It!

Let's add some images and line  
breaks to our page.

We can even turn our images into  
links!

# Element: Unordered and ordered lists

```
<ul>
  <li>List Item</li>
  <li>AnotherList Item</
li>
</ul>
```

Unordered list  
(bullets)

- List Item
- AnotherList Item

```
<ol>
  <li>List Item</li>
  <li>AnotherList Item</
li>
</ol>
```

Ordered list  
(sequence)

1. List Item
2. AnotherList Item

# Lists: Examples

Lists can be used to organize any list of items.

The screenshot shows a web-based application interface with several sections:

- PROGRAM**: A section containing numbered items 1 through 11, each with a small icon and text: DAY 1 / THURSDAY 8TH NOV., DAY 2 / FRIDAY 9TH NOV., DAY 3 / SATURDAY 10TH NOV. (highlighted in blue), and DAY 4 / SUNDAY 11TH NOV.
- FAVORITES**: A sidebar with a list of items: News Feed (20+), Photos, Close Friends, Messages (6), and Events (10).
- Job Titles**: A list of bullet-pointed job titles: Designer, Project Manager, Entrepreneur, Support Pro, Consultant, Blogger, Documentation Writer, and Teaching & Training.
- Languages**: A list of language icons with their names: English, Czech, English (Canada), and Filipino.
- Add PlayThru to existing Contact Form**: A callout box with three numbered steps:
  - Select the contact form you would like PlayThru to appear on.
  - Select "Are You a Human" from the "Generate Tag" drop down.
  - Follow the onscreen instructions and save the form.
- Sidebar Categories**: A vertical sidebar on the right side listing various topics: Android, Branding, Business, Business Cards, Captcha, Cheat Sheets, Coding Essentials, Content Strategy, Conferences, Copyright, CSS & CSS3, Design Best Practices, Design Legacy, Downloads, and E-Commerce.

You'd be surprised how often lists are used in web design.

# Let's Develop it!

Let's add one of each ordered and unordered lists to our page.

We can make a list of links or even a list of images!

# Comments

You can add comments to your code that will not be seen by the browser, but only visible when viewing the code.

```
<!-- Comment goes here -->
```

Comments can be used to organize your code into sections so you (or someone else) can easily understand your code. It can also be used to 'comment out' large chunks of code to hide it from the browser.

```
<!-- Beginning of header -->
    <div id="header">Header Content </div>
<!-- End of header -->

<!--
    <ol>
        <li>List Item</li>
        <li>Another List Item</li>
    </ol>
-->
```

# Tables

Tables are a way to represent complex information in a grid format.

Tables are made up of rows and columns.

```
<table>
  <tr>
    <th>Head</th>
    <th>Head</th>
  </tr>
  <tr>
    <td>Data</td>
    <td>Data</td>
  </tr>
</table>
```

Head	Head
Data	Data

# Tables: Examples

Tables can be styled with CSS to add zebra striping or to highlight important rows/columns.

Name	Format	Uniques	Eng. Rate	CTR	CE	VTR	Visit time	Conv. rate
Campaign total	all	810K	26%	0.48%	7.6/10	0.20%	4s	0.09%
Burt Rich	980x120	300K	30%	0.64%	7.9/10	0.20%	20s	0.06%
Burt Meme Machi...	768x150	180K	12%	0.14%	6.0/10	0.16%	25s	0.02%
Burt Copybox	300x250	68K	7.5%	0.14%	6.2/10	0.10%	7s	0.01%
Burt Copybox	468x400	65K	35%	0.30%	8.1/10	0.19%	16s	0.01%
Burt Rich	350x250	64K	11%	0.39%	6.4/10	0.08%	12s	0.08%
Burt Meme Machi...	768x150	64K	18%	0.22%	8.9/10	0.10%	14s	0.06%

Products	Price	US\$ 1,299	US\$ 1,899	US\$ 1,899	US\$ 2,599	US\$ 49.99/mo†
Photoshop	US\$699	•				
Photoshop Extended	US\$999		•	•	•	•
Illustrator	US\$599	•	•	•	•	•
InDesign	US\$699	•	•		•	•
Adobe Muse	US\$14.99/mo†					•
Acrobat X Pro for Creative Suite	US\$449	•	•		•	•
Flash Professional	US\$699		•	•	•	•

# Character codes

There are character codes for many different characters in many different languages

- Delta: &delta; δ
- Copyright symbol  
&copy; ©
- Grave: &grave; `
- An grave a: &agrave;  
à
- A full list is available  
at

[htmlandcssbook.com](http://htmlandcssbook.com)

©	Copyright	&copy;	&#169;
®	Registered trademark	&reg;	&#174;
™	Trademark	&trade;	&#8482;
<	Less than	&lt;	&#60;
>	Greater than	&gt;	&#62;
&	Ampersand	&amp;	&#38;
"	Quotation mark	&quot;	&#34;
¢	Cent	&cent;	&#162;
£	Pound	&pound;	&#163;
¤	Currency	&curren;	&#164;
¥	Yen	&yen;	&#165;
€	Euro	&euro;	&#8364;

# Questions?

We're here to help!



# Break

Let's eat!

# Anatomy of a website

Your Content

+ HTML: Structure  
+ CSS: Presentation  
= Your Website

A website is a way to present your content to the world, using HTML and CSS to present that content & make it look good.

# **CSS: What is it?**

CSS = Cascading Style Sheets

CSS is a "style sheet language" that lets you style the elements on your page.

CSS works in conjunction with HTML, but is not HTML itself.

# CSS: What can it do?

All colored text, position, backgrounds, and size

The screenshot shows the homepage of the Girl Develop It website. At the top left is the logo "Girl Develop It" with the tagline "don't be shy. develop it." Below the logo are social media icons for Twitter and Facebook. At the top right are navigation links: CLASSES, CHAPTERS, EVENTS, RESOURCES, and PARTNERS. The main banner features a dark background image of people working at computers, overlaid with white text: "Empowering women of diverse backgrounds from around the world to learn how to develop software". Below the banner are three circular icons: a teal icon with a laptop labeled "CLASSES", a green icon with two stylized paintbrushes labeled "COMMUNITY", and a yellow icon with a wrench labeled "RESOURCES". At the bottom of the page are two sections: "CITIES WHERE WE ARE MAKING AN IMPACT" and "UPCOMING EVENTS".

# CSS: What does it look like?

```
}->
body {-
>   font: 100% "Gotham", sans-serif;-
>   width: 100%;-
>   padding: 0;-
>   margin: 0;-
}->
header{-
>   clear:both;-
>   width: 100%;-
}->
h3{-
>   font-size: 110%;-
>   font-weight: normal;-
>   letter-spacing: 1px;-
>   margin-bottom: 12px;-
>   margin-top: 10px;-
>   text-transform: uppercase;-
}->
a{-
>   color: #01A9B4;-
>   text-decoration: none;-
>   background-color: transparent;-
}->
a:hover{-
>   color: #222;-
}->
#page, #site_bar_content, .header, .footer {-
>   max-width: 1200px;-
>   min-width: 480px;-
>   margin: 0px auto;-
}->
.header{-
>   padding: 10px 15px;-
}->
```

# The CSS Rule



# The CSS Rule

```
selector {  
  property: value;  
}
```

A block of CSS code is a rule.

The rule starts with a selector.

It has sets of properties and values.

A property-value pair is a declaration.

# CSS Syntax

Declarations: Property and value of style you plan use on HTML element.

Declarations end with a semicolon

Declaration groups are surrounded by curly brackets.

```
selector {  
    property: value;  
    property: value;  
    property: value;  
}
```

# Selector: Element

```
p {  
    property: value;  
}
```

Selects all paragraph elements.

```
img {  
    property: value;  
}
```

Selects all image elements.

# Selector: ID

```
#footer {  
    property: value;  
}
```

Selects all elements with an id of "footer".

```
<p id="footer">Copyright 2011</p>
```

The associated HTML.

# Selector: Class

```
.warning {  
  color: red;  
}
```

Selects all elements with a class of "warning".

```
<p class="warning">Run away!</p>
```

The associated HTML.

# IDs vs. Classes

**ID** -- Should only apply to one element on a webpage. I.E. A webpage only has one footer. The "#" is how you tell CSS "this is an id."

**Class** -- Lots of elements can have the same class. I.E. There can be many warning on one webpage. The "." is how you tell CSS "this is a class name."

# Selector: Child

```
p em {  
  color: yellow;  
}
```

Selects all em elements that are  
within a paragraph

```
<p>This is <em>important.</em></p>
```

The associated HTML.

# Property Values

Each property can have one or more comma separated values.

```
p{  
  color: white;  
  background-color: red;  
  font-family: Arial, sans-serif;  
}
```

# Property: Color

The color property changes the color of the text.

```
p {  
  color: red;  
  color: #ff0000;  
  color: rgb(255, 0, 0);  
}
```

Color name

Hexadecimal value

RGB value

The 17 standard colors are: **aqua**, black, **blue**, **fuchsia**, gray, grey, **green**, **lime**, **maroon**, **navy**, **olive**, **purple**, **red**, **silver**, **teal**, **white**, and **yellow**.

# Property: Background-color

The background-color property changes the color of the background.

```
p {  
background-color: black;  
background-color: #000000;  
background-color: rgb(0,0,0);  
}
```

# Property: Font-family

The font-family property defines which font is used.

```
p {  
  font-family: "Times New Roman";  
  font-family: serif;  
  font-family: "Arial", sans-serif;  
}
```

Specific font name

Generic name

Comma-separated list

# Property: Font-size

The font-size property specifies the size of the font.

```
p {  
    font-size: 12px;  
    font-size: 1.5em;  
    font-size: 100%;  
}
```

Pixels  
"em"  
Percentage

# Property: Fonts (shorthand)

```
p {  
    font-style: italic;  
    font-weight: bold;  
    font-size: 10px;  
    font-family: sans-serif;  
}
```

OR

```
p {  
    font: italic bold 10px sans-serif;  
}
```

# Connecting CSS to HTML

3 ways

"Inline"

"Embedded"

"External"

# Connecting CSS to HTML: Inline

```
<p style="color:red">Some text.</p>
```

Uses the HTML attribute style.  
Difficult to use in large projects  
Not preferred.

# Connecting CSS to HTML: Embedded

```
<head>
  <style type="text/css">
    p {
      color: blue;
      font-size: 12px;
    }
  </style>
</head>
```

Inside `<head>` element.

Uses `<style>` tag.

Can only be used in one html file

# Connecting CSS to HTML: Linked

```
<head>
  <link rel="stylesheet" type="text/css" href="style.css">
</head>
```

Shared resource for several pages.

Reduced file size & bandwidth

Easy to maintain in larger projects.

Preferred by nerds everywhere!

# Let's develop it

- Create a new .css file
- Add a link to the file in the head of the portfolio made last time
- Add styles to change the colors, background colors or fonts of different parts of our page
- Try using ids and classes to change specific elements

# Cascading

Styles "cascade" down until changed

```
p{
  color:blue;
  font-family: 'Helvetica';
}
.red{
  color:red;
}
#special{
  font-family: Arial;
}

<p>Paragraph</p>
<p class ="green">Paragraph</p>
<p class ="red">Paragraph</p>
<p class = "red" id ="special">Paragraph</p>
```

# CSS Properties

Many CSS properties have self-explanatory names:

- background-color
- font-family
- font-size
- color
- width
- height

**Comprehensive list of all CSS properties**

# Homework for tomorrow

- View the source of some of your favorite web sites and see if you can start to recognize the parts of code we talked about today.
- Find parts of the site you're unsure of how they're built and bring those examples to class. We will have a section of class where we can pick apart the code to see how it's built.

# Questions?

We're here to help!



E-mail me with questions you have  
tonight (or any day) at  
**[corrinajo@gmail.com](mailto:corrinajo@gmail.com)**

