

Code Your Own Website



Intro to HTML + CSS

Class 1

First Things First

Thank you to our wonderful TAs!

Course website: <http://anything.codes>

Instructor contact info: lizs@girldevelopit.com

Get Started: Tools

We'll be using the following tools in class today:

- **Browser**
 - Chrome
- **Development Toolkit**
 - Chrome's Inspector (built-in to Chrome)
- **Text Editor (your preference)**
 - Mac: SublimeText, TextWrangler, jEdit
 - Windows: SublimeText, Notepad++
 - Linux: SublimeText, jEdit, gedit

Terms

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

What is HTML?

- HTML is the code that allows us to build websites
- It adds **structure** to a webpage's content

What is HTML?

- CERN scientist Tim Berners-Lee created "hypertext" to share scientific papers
- **HTML**: Hyper Text Markup Language
- **HTTP**: Hyper Text Transfer Protocol



- First web page August 6, 1991

What's a Markup Language?

A markup language is a set of **markup tags**:

```
<tagname>content</tagname>
```

Each HTML tag **describes** its content:

```
<p>This sentence goes in a paragraph (p) tag.</p>
```

HTML in Action

If you 'View Page Source', you see this:

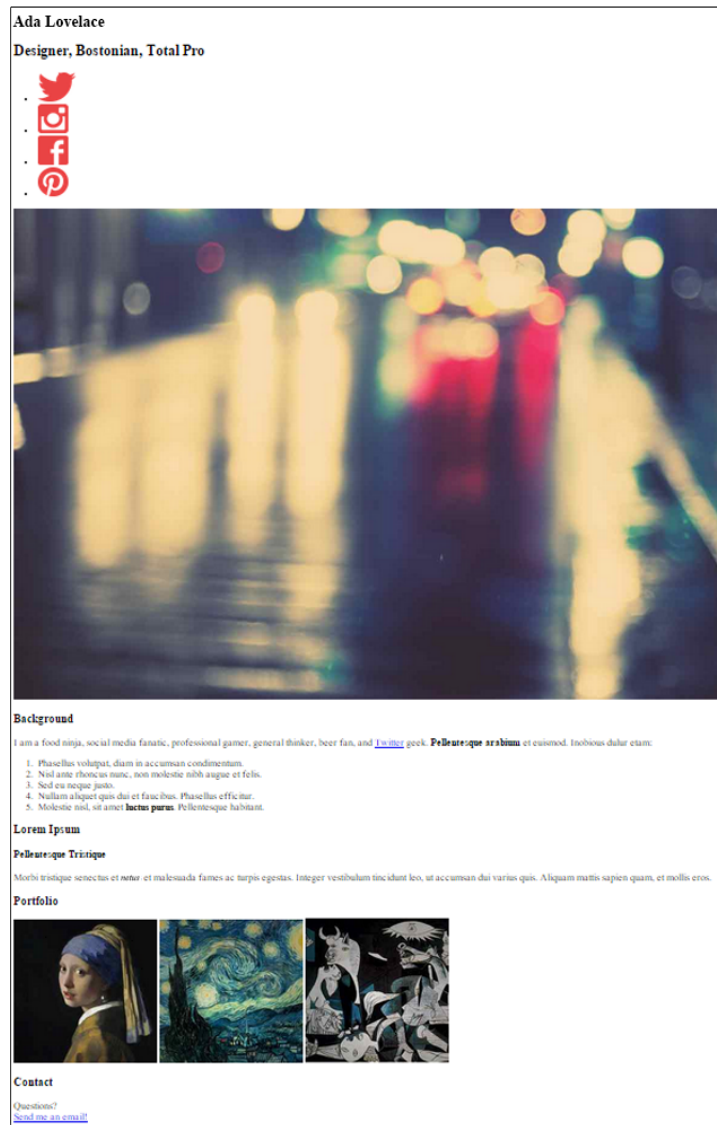
```
132 <section>
133   <h3>HTML in Action</h3>
134   <p>If you 'View Page Source', you see this:
135     <br>
136   </p>
137 </section>
138
139   <!-- Terms -->
140 <section>
141   <h3>Terms</h3>
142   <ul>
143     <li>
144       <div class="green">Web design</div>
145       <div>The process of planning, structuring and creating a website</div>
146     </li>
147     <li>
148       <div class="green">Web development</div>
149       <div>The process of programming dynamic web applications</div>
150       <ul>
151         <li>
152           <div class="orange">Front end development &#128077;</div>
153           <div>The outwardly visible elements of a website or application</div>
154         </li>
155         <li>
156           <div class="green">Back end development</div>
157           <div>The inner workings and functionality of a website or application.</div>
158         </li>
159       </ul>
160     </li>
161   </ul>
162 </section>
163
164   <!-- Get Started -->
165 <section>
166   <h3>Get Started: Tools</h3>
167   <p>We'll be using the following tools in class today:</p>
168   <p>&nbsp;</p>
169   <ul>
170     <li>
```


Anatomy of a Website

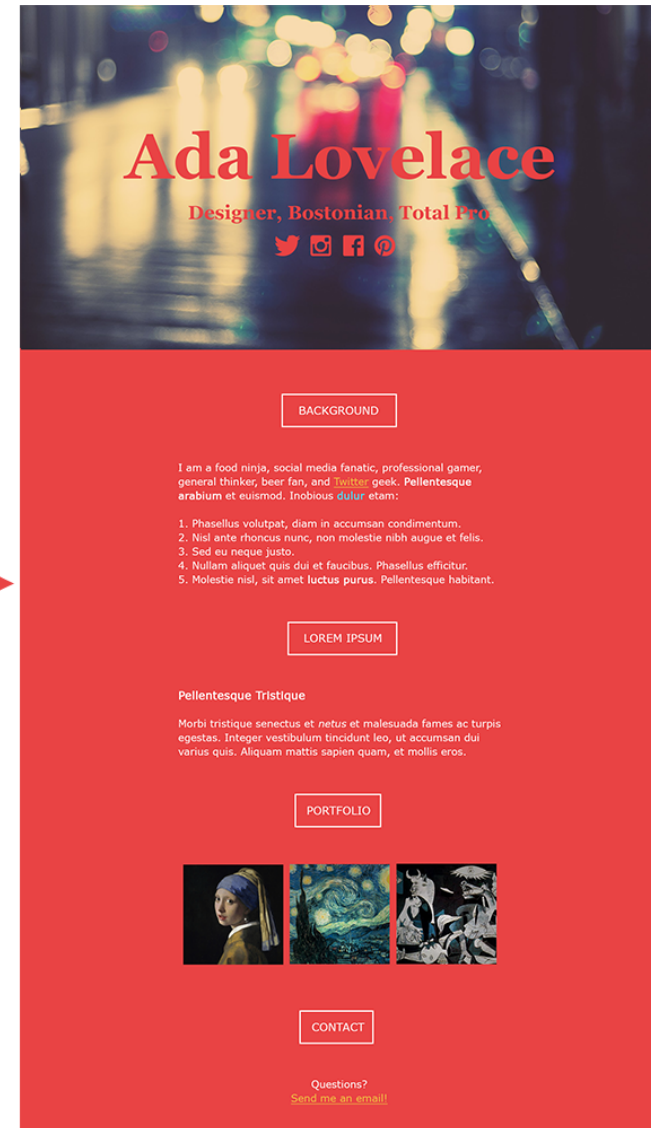
Your Content
+ HTML: Structure
+ CSS: Presentation
= Your Website

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

Our Class Project



BECOMES



Anatomy of a Website

Concrete example

- Content:

```
A paragraph is your content
```

- + **HTML**: Putting your content into an HTML tag to make it look like a paragraph is structure

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

- + **CSS**: Making the paragraph's text red with an 18px font size is presentation
- = **Website**: In the browser, the paragraph looks like:
A paragraph is your content

Anatomy of an HTML element

- Element

- A full block, including opening and closing tags. An element is comprised of its content and tags.

```
<section>Content in the middle</section>
```

- Tag

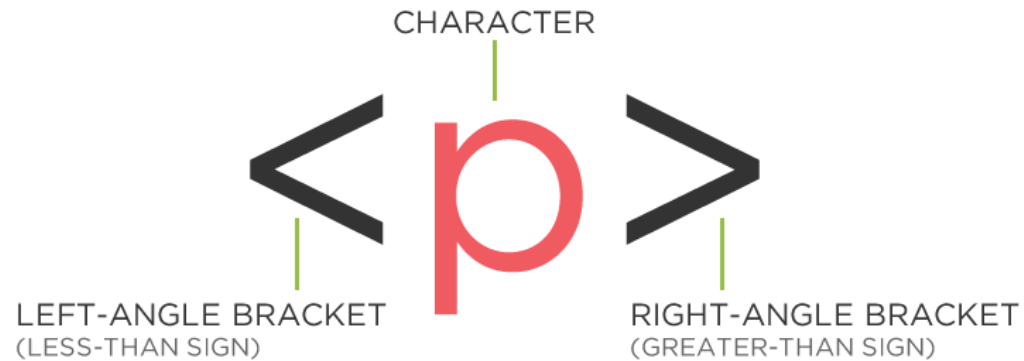
- Tags mark the beginning and end of an element, and indicate the element's purpose. There are opening and closing tags:

```
<p>The p tag means this content is a paragraph.</p>
```

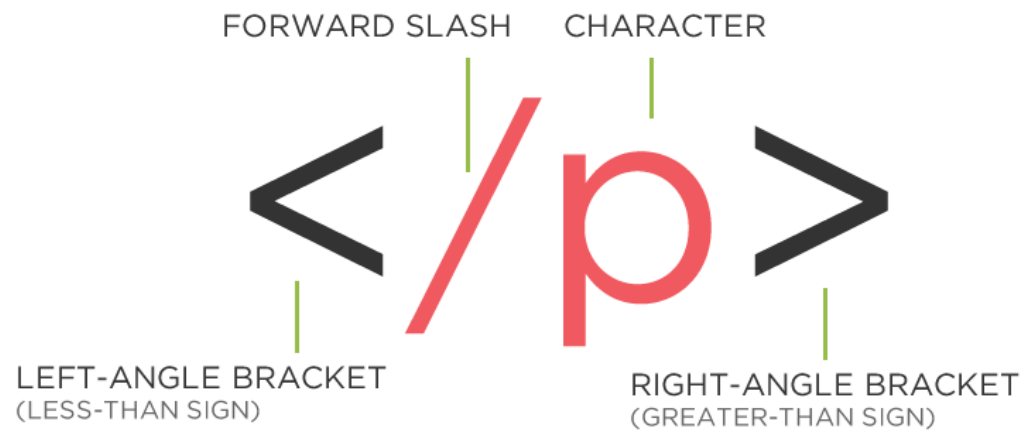
```
<header>The header tag means this content is a website's header.</header>
```

Tag Breakdown

OPENING TAG



CLOSING TAG



HTML Coding Tip #1

Whenever you type an opening tag, immediately type the closing tag, then fill in your content.

1.

```
<strong>
```

2.

```
<strong></strong>
```

3.

```
<strong>Now I can add content!</strong>
```

Anatomy of an HTML element

Shorthand

- **Container Element:** Can contain content within the opening and closing tags:

```
<p>This is content within the opening and closing tags.</p>
```

- **Stand-Alone Element:** When an element does not contain any content within its opening and closing tags, use the `<tagname />` shorthand:

```
<br />
```

```

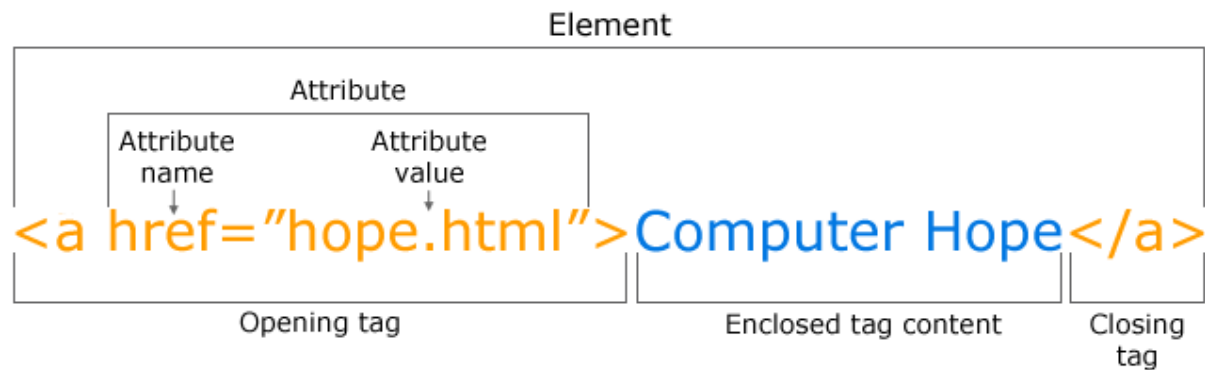
```

The examples are shorthand for `
</br>` and ``

Anatomy of an HTML element

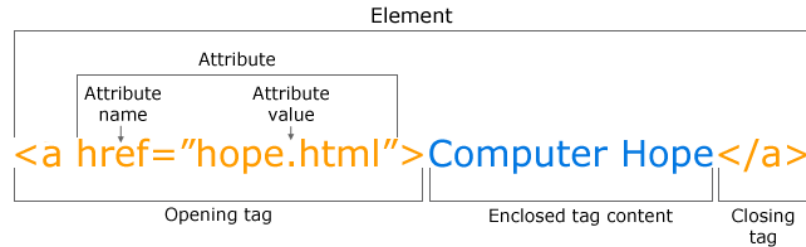
Attributes

Adding **attributes** to an HTML tag provides additional information about the HTML element



Anatomy of an HTML element

Attributes



- **Attribute Name:** class, ID, style, href, etc.
 - Placed inside an opening tag, before the right angle bracket.
- **Attribute Value**
 - Value is the value assigned to a given attribute
 - Values must be contained inside quotation marks:

```

<div id="intro">Lorem ipsum</div>
<a href="http://girldevelopit.com" class="fancy-link">GDI</a>
```

The Fundamental Structure of an HTML File

Doctype

The first thing in an HTML file is the doctype, which tells the browser which language the page is using:

```
<!DOCTYPE html>
```



The doctype is case-insensitive.
DOcType, doctype, DocType and DoCtYpe are all valid.

The Fundamental Structure of an HTML File

HTML Element

After `<!DOCTYPE html>`, the page content must be contained between `<html></html>` tags.

```
<!DOCTYPE html>  
<html>  
  
</html>
```

The Fundamental Structure of an HTML File

Head Element

The head contains information about the page, but does not contain page content. It contains elements that let the browser know:

- The page's **title**
- **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 the page's description
- Where to find the **CSS file** (which styles the page)

```
<!DOCTYPE html>
<html>
    <head>

    </head>
</html>
```

The Fundamental Structure of an HTML File

Body Element

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

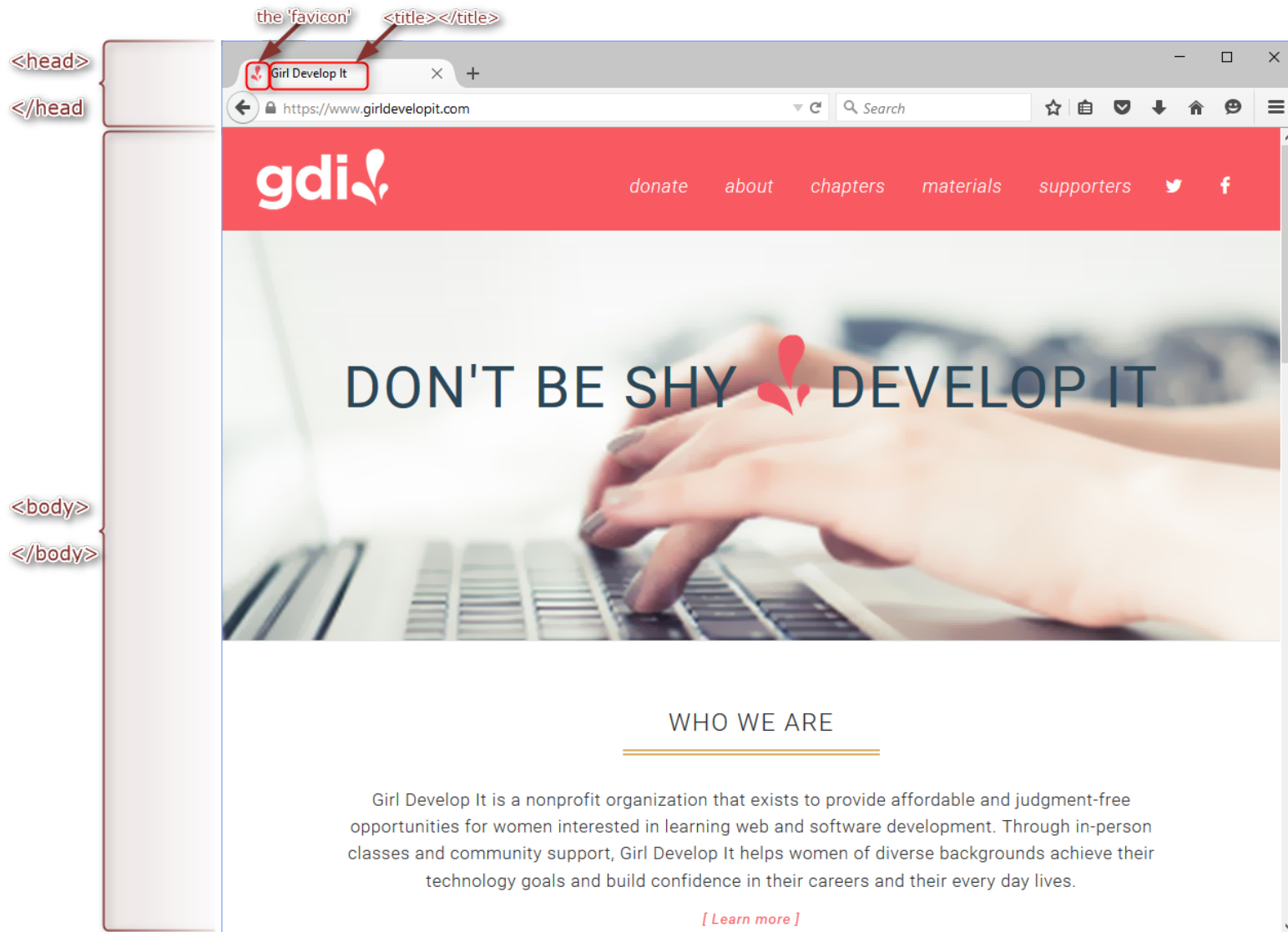
Most of your work will be done within the `<body></body>` tags!

```
<!DOCTYPE html>
<html>
  <head>

  </head>
  <body>

  </body>
</html>
```

Head and Body Tags: Example



The Fundamental Structure of an HTML File

Memorize this 😊

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title of the page</title>
  </head>
  <body>
    All of your page content goes here
  </body>
</html>
```

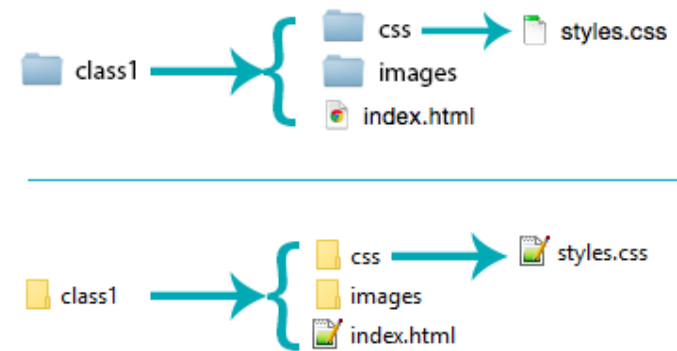
Get Started: Folder Structure

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

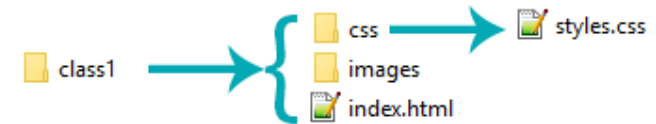
Note: File names should not include spaces or special characters. File names ARE case sensitive.



Get Started: Folder Structure

Go ahead and create
your folders

Ignore the HTML and
CSS files for now

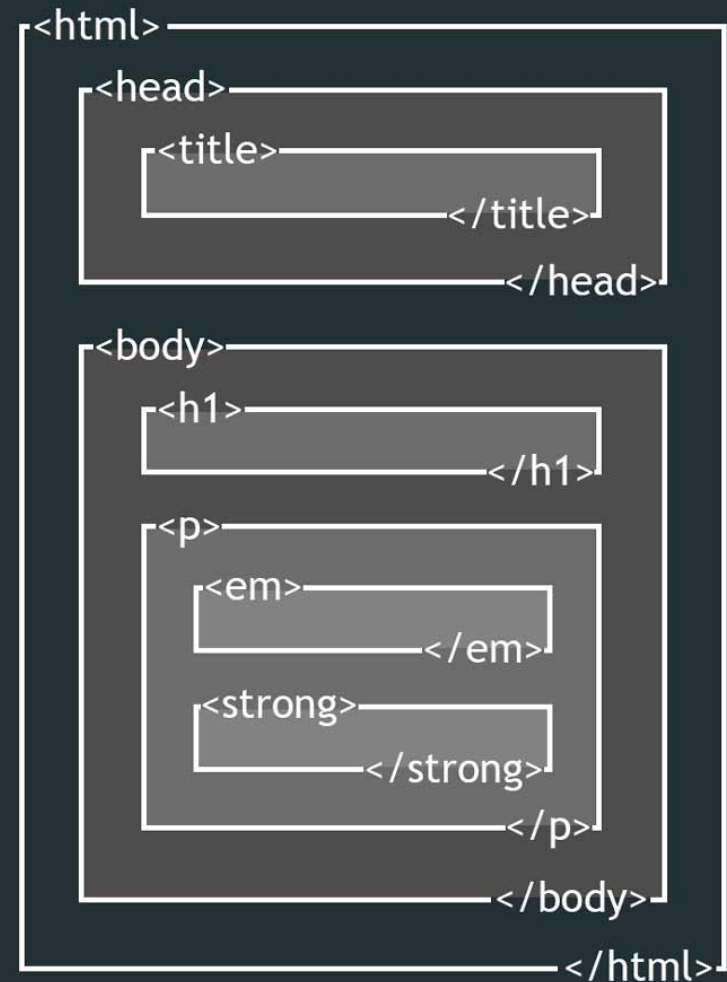


Nesting

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></p>` inside of the `<body></body>` tags. The `<p></p>` is now nested inside the `<body></body>`, and is one of its descendants



The diagram illustrates the nesting of HTML tags using a series of nested rectangles. The outermost rectangle is labeled `<html>` at the top-left and `</html>` at the bottom-right. Inside it is a rectangle labeled `<head>` at the top-left and `</head>` at the bottom-right. Inside the `<head>` rectangle is a smaller rectangle labeled `<title>` at the top-left and `</title>` at the bottom-right. Below the `<head>` rectangle is another rectangle labeled `<body>` at the top-left and `</body>` at the bottom-right. Inside the `<body>` rectangle is a rectangle labeled `<h1>` at the top-left and `</h1>` at the bottom-right. Below the `<h1>` rectangle is a rectangle labeled `<p>` at the top-left and `</p>` at the bottom-right. Inside the `<p>` rectangle are two smaller rectangles: the top one is labeled `` at the top-left and `` at the bottom-right, and the bottom one is labeled `` at the top-left and `` at the bottom-right. The nesting shows that `<title>` is nested inside `<head>`, which is nested inside `<html>`. Similarly, `<h1>` and `<p>` are nested inside `<body>`, which is nested inside `<html>`. The `` and `` tags are nested inside the `<p>` tag.

```
<html>  
  <head>  
    <title>  
  </head>  
  <body>  
    <h1>  
    <p>  
      <em>  
      <strong>  
    </p>  
  </body>  
</html>
```

Nesting: Example

All your page's content is 'nested' inside the body element:

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

And other elements can be nested inside of that:

```
<body>  
  <p>      A paragraph inside the body element  
            <em>which has some italic text</em>  
            and  
            <strong>some bold text</strong>  
  </p>  
</body>
```

⇒ A paragraph inside the body element *which has some italic text* and **some bold text**

Nesting

HTML elements are often looked at as a family tree. Developers will often refer to elements as "siblings", "immediate children", and "descendants".

Can you name any siblings?

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title of the page</title>
  </head>
  <body>
    All of your page content goes here
  </body>
</html>
```

How about immediate children?

HTML Coding Tip #2

1. Whenever you add a 'child' element:

```
<ul>  
</ul>
```

2. ... indent it on a new line!

```
<ul>  
    <li>I'm indented!</li>  
    <li>I'm also indented!</li>  
</ul>
```

This will make your life much easier down the road, as you add more content and style your pages.

HTML Coding Tip #3

HTML ignores multiple whitespaces in a row. You can add consecutive white space characters (spaces, tabs, & carriage returns) into your HTML, but when you view that page — *all but one disappears!*

HTML:

```
<p>Two carriage returns  
Three carriage returns  
  
Plus multiple spaces    ...</p>
```

Result:

Two carriage returns Three
carriage returns Plus multiple
spaces ...

To force consecutive spaces to show, use ` ` (stands for "no-break space") in your HTML instead of spaces.

Common HTML Elements

Paragraph

HTML:

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

Result:

Paragraph 1
Paragraph 2
Paragraph 3

White space outside of any tags won't render
(that's just for us humans!):

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

Common HTML Elements

Paragraphs in Action

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

Common HTML Elements

Headings

HTML:

```
<h1>Heading 1</h1>
```

```
<h2>Heading 2</h2>
```

```
<h3>Heading 3</h3>
```

```
<h4>Heading 4</h4>
```

```
<h5>Heading 5</h5>
```

```
<h6>Heading 6</h6>
```

Result:

Heading 1

Heading 2

Heading 3

Heading 4

Heading 5

Heading 6

Common HTML Elements

Headings in Action

THIS IS THE MAIN HEADING OF THE DOCUMENT

We choose to go to the moon. We choose to go to the moon in this decade and do the other things, not because they are easy, but because they are hard, because that goal will serve to organize and measure the best of our energies and skills, because that challenge is one that we are willing to accept, one we are unwilling to postpone, and one which we intend to win, and the others, too.

This is a simple title of level 2

The surface is fine and powdery. I can kick it up loosely with my toe. It does adhere in fine layers, like powdered charcoal, to the sole and sides of my boots. I only go in a small fraction of an inch, maybe an eighth of an inch, but I can see the footprints of my boots and the treads in the fine, sandy particles. There seems to be no difficult in moving around, as we suspected.

Another one

First, I believe that this nation should commit itself to achieving the goal, before this decade is out, of landing a man on the moon and returning him safely to the earth. No single space project in this period will be more impressive to mankind, or more important for the long-range exploration of space; and none will be so difficult or expensive to accomplish.

THIS ONE GOES DEEPER

A good rule for rocket experimenters to explode. The vehicle explodes, literally shakes you a little bit, but the actual liftoff follow is this: always assume that it will explodes, off the pad. The simulator shakes your entire body and soul.

Even deeper

The path of the righteous man is beset on all sides by the iniquities of the selfish and the tyranny of evil men. Blessed is he who, in the name of charity and good will, shepherds the weak through the valley of darkness, for he is truly his brother's keeper and the finder of lost children.

AND A LEVEL-3 TITLE AGAIN

It has been said that astronomy is a humbling and character-building experience. There is perhaps no better demonstration of the folly of human conceits than this distant image of our tiny world. To me, it underscores our responsibility to deal more kindly with one another, and to preserve and cherish the pale blue dot, the only home we've ever known.

Common HTML Elements

Formatted text

HTML:

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

Result:

Here is a
paragraph with
Emphasized
text and
Important text.

Common HTML Elements

Links

Standard links have three components:

1. The **tag**:
`<a>`
2. The **content** (the clickable portion within the `a` element):
GDI Boston
3. The **href** attribute (the destination of the link):
`href="http://www.girldevelopit.com/chapters/boston"`

```
<a href="http://www.girldevelopit.com/chapters/boston">GDI Boston</a>
```

Common HTML Elements

Additional Link Options

1. The `title` attribute for descriptive 'hover' text:
title="Read more about GDI Boston"
2. The `target` attribute:
target="_blank" (opens link in a new tab)

```
<a href="http://www.girldevelopit.com/chapters/boston" title="Read more  
about GDI Boston" target="_blank">GDI Boston</a>
```

Common HTML Elements

Additional Link Options

3. Surround another element, such as a heading or images, within `<a>` tags to link it

HTML:

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

Result:



Common HTML Elements

Additional Link Options

4. **Make an email link**, which launches a user's mail program, by inserting **mailto:** directly before the email address

```
<a href="mailto:lizs@girldevelopit.com">Email me!</a>
```

Common HTML Elements

Images

Images have three components:

1. The **tag**:
``
2. The **src** attribute:
`src="http://lorempixel.com/200/200/city/"`
3. The **alt** attribute:
`alt="Picture of a city"`

```

```



File Paths

Relative vs. absolute paths for links, images, etc.

- **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/boston"`
- Typically used when pointing to a link that is not within your own domain.

Common HTML Elements

Line Break

```
<p>
  You spin me right round, baby<br />
  Right round like a record, baby<br />
  Right round round round
</p>
```

You spin me right round, baby
Right round like a record, baby
Right round round round

Common HTML Elements

Unordered and ordered lists

HTML:

```
<ul>  
  <li>Unordered List Item</li>  
  <li>Another List Item</li>  
</ul>  
  
<ol>  
  <li>Ordered List Item</li>  
  <li>Another List Item</li>  
</ol>
```

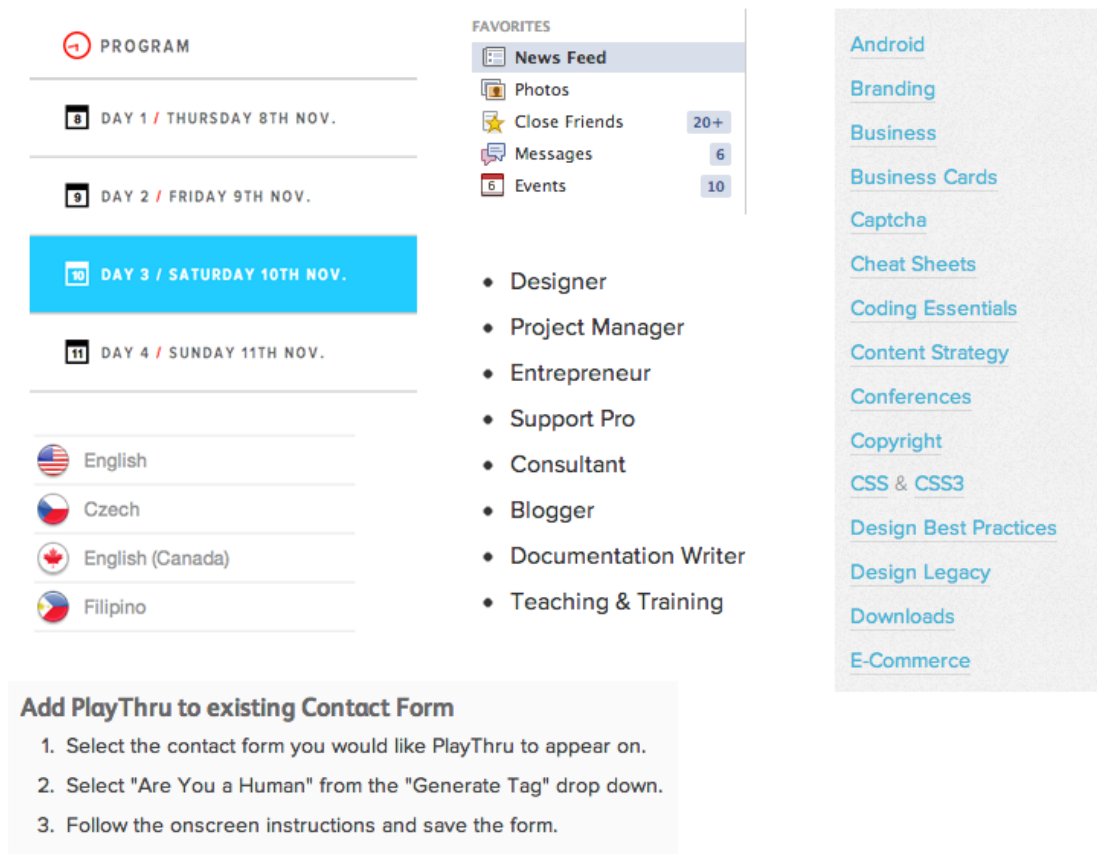
Result:

- Unordered List Item
 - Another List Item
-
1. Ordered List Item
 2. Another List Item

Common HTML Elements

Lists in Action

Lists can be used to organize any list of items.



The image displays three distinct examples of HTML lists used in web development:

- Vertical List of Dates:** A list of four days, with the third day highlighted in blue. The items are: PROGRAM, DAY 1 / THURSDAY 8TH NOV., DAY 2 / FRIDAY 9TH NOV., DAY 3 / SATURDAY 10TH NOV. (highlighted), and DAY 4 / SUNDAY 11TH NOV.
- Horizontal List of Favorites:** A list of five items with corresponding icons and counts: News Feed, Photos, Close Friends (20+), Messages (6), and Events (10).
- Vertical List of Links:** A list of twelve links, each preceded by a bullet point: Designer, Project Manager, Entrepreneur, Support Pro, Consultant, Blogger, Documentation Writer, Teaching & Training, 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.

Add PlayThru to existing Contact Form

1. Select the contact form you would like PlayThru to appear on.
2. Select "Are You a Human" from the "Generate Tag" drop down.
3. Follow the onscreen instructions and save the form.

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

Common HTML Elements

Comments

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

```
<!-- 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>-->
```

Common HTML Elements

Tables

Tables are a way to represent complex information in a grid format. Tables are made up of rows (`tr`), which contain cells (`th` for table header cells, or `td` for standard table data cells).

HTML:

```
<table>
  <tr>
    <th>Head 1</th>
    <th>Head 2</th>
  </tr>
  <tr>
    <td>Data 1</td>
    <td>Data 2</td>
  </tr>
</table>
```








Result:

Head 1	Head 2
Data 1	Data 2

Common HTML Elements

Tables in Action

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		●	●	●	●

Common HTML Elements

Character Codes

- Space (to make consecutive spaces):

` `

- Copyright symbol: ©

`©`

- Double arrow: »

`»`

- Less than, greater than: < and >

`<` and `>`

- A full list is available [here](#)

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