

# Introduction to HTML

# What is HTML?

- HTML is the standard markup language for creating Web pages.
- HTML stands for Hyper Text Markup Language
- HTML describes the **structure** of Web pages using markup
- HTML **elements** are the building blocks of HTML pages
- HTML elements are represented by **tags**
- HTML tags label pieces of content such as "heading", "paragraph", "table",...
- Browsers do not display the HTML tags, but use them to render the content of the page

# An Example

A paragraph is your **content**

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

**<p>**This is the content**</p>**

# HTML Elements and Tags

An *element* is an individual component of HTML

- paragraph, image, header,...
- an element name indicates its purpose: *p* for *paragraph*

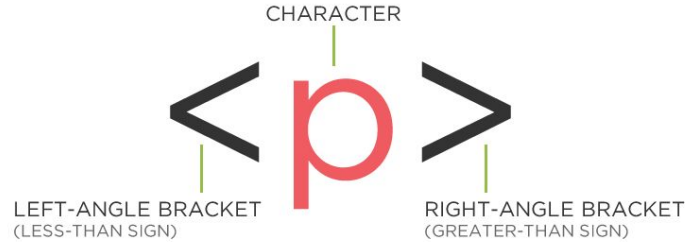
A *tag* marks the beginning and the end of an HTML element

- Opening tag and Closing Tag

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

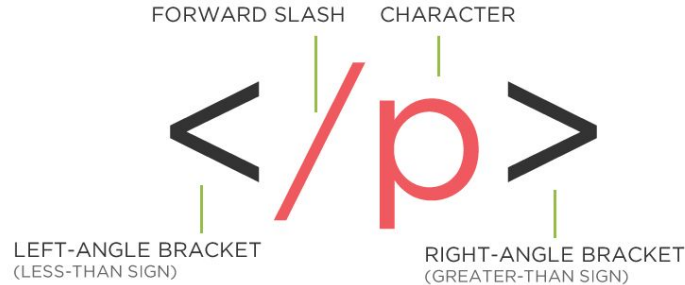
# HTML Tag Breakdown

## OPENING TAG



---

## CLOSING TAG



# Anatomy of an HTML element

## Container Element

- An element that can contain other elements or content
- A paragraph (`<p>content</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.

```

```

# HTML Page Structure

```
<!DOCTYPE html>
<html>

  <head>
    <title>My title</title>
  </head>

  <body>
    <h1>My heading</h1>
    <p>My paragraph</p>
    My image
  </body>

</html>
```



# <!DOCTYPE>

The <!DOCTYPE> declaration represents the document type, and helps browsers to display web pages correctly.

It must only appear once, at the top of the page (before any HTML tags).

The <!DoCtYpE> declaration is not case sensitive.

The <!DOCTYPE> declaration for HTML5 is <!DOCTYPE html>

# <html>, <head>, <body>

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

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, like providing information to search engines.

The body contains the actual content of the page. Everything that is contained in the body is visible to the user. (Some Exceptions!)

# Let's create our first HTML page

open a new file in your text editor and copy this code in it:

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

save the document as myfirstpage.html and open it with a browser.

# Nesting

All elements "nest" inside one another

Nesting is what happens when you put 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>`

```
<body>  
  <p>The paragraph goes here.</p>  
</body>
```

Whichever element **OPENS** first **CLOSES** last

# Headings

HTML headings are defined with the `<h1>` to `<h6>` tags.

`<h1>` defines the most important heading. `<h6>` defines the least important heading

```
<h1>This is heading 1</h1>
```

**This is heading 1**

**This is heading 2**

**This is heading 3**

**This is heading 4**

**This is heading 5**

**This is heading 6**

# Paragraphs

HTML paragraphs are defined with the `<p>` tag:

```
<p>This is a paragraph</p> <p>Paragraph 2</p>
```

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.

# The Poem Problem and Line Breaks

This poem will display in a single line:

```
<p>  
  My Bonnie lies over the ocean.  
  My Bonnie lies over the sea.  
  My Bonnie lies over the ocean.  
  Oh, bring back my Bonnie to me.  
</p>
```

The browser will remove any extra spaces and extra lines when the page is displayed

The `<br>` element defines a line break without starting a new paragraph.

```
<p>  
  My Bonnie lies over the ocean. <br>  
  My Bonnie lies over the sea. <br>  
  My Bonnie lies over the ocean. <br>  
  Oh, bring back my Bonnie to me.  
</p>
```

# Preformatted Text

The HTML `<pre>` element defines preformatted text.

The text inside a `<pre>` element is displayed in a fixed-width font (usually Courier), and it preserves both spaces and line breaks.

```
<pre>  
  My Bonnie lies over the ocean.  
  My Bonnie lies over the sea.  
  My Bonnie lies over the ocean.  
  Oh, bring back my Bonnie to me.  
</pre>
```

This will be shown correctly  
with line breaks preserved.



# Formatted Text

```
<p>
```

```
Here is a paragraph with <em>emphasized</em> text and <strong>important</strong> text.
```

```
</p>
```

```
<p>
```

```
Here is another paragraph with <i>Italic</i> text and <b>Bold</b> text.
```

```
</p>
```

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

Here is another paragraph with *Italic* text and **Bold** text.

# Formatted Text

`<p>Here is a highlighted text and this is smaller text.</p>`

Here is a Highlighted text and this is smaller text.

`<p>Here is some deleted text and this is some inserted text.</p>`

Here is ~~some deleted~~ text and this is some inserted text.

`<p>This is superscripted text and this is subscripted text.</p>`

Here is <sup>superscripted</sup> text and this is <sub>subscripted</sub> text.

# Character Codes

Many mathematical, technical, and currency symbols, are not present on a normal keyboard.

To add such symbols to an HTML page, you can use an HTML entity name or code.

One character that I use most is `&nbsp;`;

[See here for a complete list](#)

©	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	&currency;	&#164;
¥	Yen	&yen;	&#165;
€	Euro	&euro;	&#8364;

# Links

HTML links are defined with the `<a>` tag

The link's destination is specified in the `href` attribute.

`target` attribute specifies where to open the linked document(`_self`, `_blank`,...)

```
<a href="http://www.google.com" target="_blank">This is a link to Google!</a>
```

This is a link to Google!

# Images

HTML images are defined with the `<img>` tag.

The source file (`src`), alternative text (`alt`), `width`, and `height` are provided as attributes.

```

```



# Relative vs. Absolute paths for links & Images

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. "img/filename.jpg"

Absolute paths refer to a specific location of a file, including the domain

- Typically used when pointing to a link that is not within your own domain.
- <http://one.fsu.edu/alumni/image/community/clubs/FSU-Seal-full-color.jpg>

# Lists

## Unordered list (bullets)

- List Item
- Another List Item

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

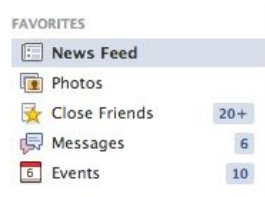
## Ordered list (sequence)

1. List Item
2. Another List Item

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

# Lists

Lists are used everywhere and can be customized to look as you want



- Designer
- Project Manager
- Entrepreneur
- Support Pro
- Consultant
- Blogger
- Documentation Writer
- Teaching & Training



## 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.



# Tables

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

Tables are made up of **rows** and **columns**.

Head	Head
Data	Data








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

# Tables

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

Extra functionality can be added to tables like filtering or sorting rows and 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		●	●	●	●

# Comments

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

Comments can be used to:

- organize your code into sections
- 'comment out' large chunks of code to hide it from the browser.

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