

insurance.com
Real quotes. Real savings. Real fast.



Car insurance made easy.
We search. You **save**.

GO

insurance.com
Real quotes. Real savings. Real fast.



Car insurance made easy.
We search. You **save**.

GO

Web Developer Class: Learn the Basic HTML Tags!

By [Joe Burns](#)

<http://www.htmlgoodies.com/primers/html/article.php/3478151/Web-Developer-Class-Learn-the-Basic-HTML-Tags.htm> ([Back to article](#))

Quality
applications faster.
Why DevOps
is the new normal.

IBM DevOps Symposium
Live Stream
November 20, 2014

REGISTER NOW ▶



(with updates by editorial staff)

Hello and welcome to our next class, learning the basic Hypertext Transfer Markup Language (HTML) tags! No doubt you've attempted to write a small document in your favorite text editor, such as Windows Notepad, and saved it as TEXT for MAC or TEXT for your PC. You also remembered to save the document with the .htm or .html suffix, I'm sure. Good, now let's move on to today's lesson, for today we actually write some HTML code!

HTML Tags

HTML works in a very simple, very logical, format. It reads like you do, from top to bottom, and left to right. That's important to remember. HTML is written with normal old text. What you use to set certain sections apart as headings, subtitles, bold text, underlined text, etc is a series of what we call "tags".

Think of tags as making your structure. Let's say you want a heading. You will put a tag at the exact point you want the heading to start and another tag where you want the heading to stop. If you want a specific word to be emphasized, you will place a start emphasis tag at the beginning of the word and an end emphasis tag at the end of the word. Is this making sense so far? Later we will discuss using [Cascading Style Sheets](#) (CSS) to do the same thing, but for now we are concentrating on HTML, so back to the lesson!

Tag Format

All tag (I sometimes call them *command*) formats are the same. They begin with a less-than sign: < and end with a greater-than sign: >. Always. No exceptions. What goes inside the < and > is the tag. Learning HTML is learning the specific tags needed to perform whatever command you want to do. Here's an example:

The tag for a paragraph is "p". That makes sense. For example:

```
<p>Joe</p>
```

In the old HTML standards, we used to use B for bold, and I for italics, etc. With the latest standards it is now accepted practice to separate content from presentation - by which we mean you set out the **structure** of your document in HTML, and control how it displays using a CSS file (again, more on CSS [here!](#))

This means we can mark our keywords using **strong** and *em* tags, which have the same effect, but comply with the latest standards.

```
<strong>Joe</strong> and <em>Burns</em>
```

Look At What's Happening:

1. is the beginning strong tag.
2. "Joe" is the word being affected by the tag.
3. is the end strong tag. Notice it is exactly the same as the beginning tag except there is a slash in front of the tag command.
4. This is what the strong tags above produced: **Joe** Nice, huh?

Some Questions

Q. *Is the end tag for other commands simply the begin tag with the added slash?*

A. Yup.

Q. *Will the tags show up on my page?*

A. No. As long as your commands are inside the < and > marks, the tag is used to alter the text, but the actual code is hidden from the viewer.

Q. *Do I use capitals or lower case? I've seen people using both.*

A. In HTML, the browser doesn't care. However, should you move on to XHTML, they will have to be lower case, so you may as well just use lower case--it can't hurt and can only help.

Q. *Must everything have a tag to show up on the page?*

A. No. If you just type in text, it'll show up. But it will not have any special look.

Q. *What if I forget to add the end tag or forget to add the slash to the end tag command?*

A. That's trouble, but easy-to-fix trouble. It will be obvious if you've not placed an end tag when you look at the document in your browser. The entire document will be affected after the point where you forgot the end tag. Just go back into the document, add the slash, and reload the document into the browser.

Q. *Do all HTML tags require both a begin and end tag, like above?*

A. No. There are exceptions to the rule, such as break
 tags and image tags , but let's stay on the ones that do require both tags to work for now. Moving along...

Open and Close Tags

The majority of HTML tags do require both an open and a close tag (a begin and end tag). Most are very easy to understand because the tag is obvious. Here are a few and what they do to text:

Affect	Code	Code Used	What It Does
Strong	Strong	Bold	Bold
Emphasis	em	Italic	<i>Italic</i>
Paragraph	p	<p>Paragraph</p>	Plain

Note: the strong and em tags are normally found inside a paragraph. There is a technical reason for this which we'll cover later.

Can I Use Two Tags at Once?

Yes. Just make sure to begin and end both. Like so:

`Strong and emphasis` gives you ***Strong and Italic***

If you do use multiple tags to alter text, make a point of not getting the end tags out of order. Look at this:

`Strong and emphasis`

In terms of format, the example above is not correct. The end tags are out of order in relation to the start tags.

Follow this rule:

Always set the beginning and end tags at the same time, always placing them on the farthest end of the item being affected.

Here, again, is the example above in correct form:

`Strong and emphasis`

Notice the strong tags are on the far ends. Next in line are the emphasis tags. Just keep setting commands at the farthest ends each time you add them and you'll stay in good form.

Single Tags

The open and close tag format dominates the majority of the available HTML tags, but there are tags that stand alone. Here are two useful ones:

tag	What It Does
<code><HR></code>	This command gives you a line across the page. (HR stands for Horizontal Reference.) The line right above the words "Single tags" was made using an <code><HR></code> tag.
<code>
</code>	This BR eaks the text and starts it again on the next line. Remember you saved your document as TEXT so where you hit ENTER to jump to the next line was not saved. In an HTML document, you need to denote where you want every carriage return with a <code>
</code> .

Writing Your First Page

So, here we go... you're going to write your first HTML page using what you have learned above plus

a few other items. And these other items are important to every page you will ever write. Why? Because they will be on every page you ever write.

For a properly formed document, you need a 'doctype'. For example:

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

This tells the browser exactly what version of HTML you are using. While this won't make any difference to you early on, when you get into CSS and positioning it will have a huge impact, so get into good habits now to avoid the problems later!

You will always have this tag: <HTML>

That makes sense. You are denoting that this is an HTML document.

Your next tags will always be these: <TITLE> and </TITLE>

See the very top of this page? I mean way up top. Above the FILE -- EDIT -- VIEW menus. The colored bar up there. Right now it reads "Web Developer Class: Learn the Basic HTML Tags" That's the title of the page and that's what you are denoting here. Whatever you put between these two tags will show up in the title bar way at the top.

Finally, you'll end every page you write with this tag: </HTML>

Get it? You started the page with HTML and you will end the page with /HTML. That makes sense again.

So, Here We Go!

I want you to play around with these commands. Just remember that HTML reads like you do, top to bottom, left to right. It will respond where you place the start tag and stop where you place the end tag. Just make sure your tags are within the < and > items.

Here's a sample page to show you what I mean for you to do tonight: <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">

```
<html>
  <head>
    <title>My first HTML document</title>
  </head>
  <body>
    <p>Hello <strong>world!</strong></p>
```

```
<br>
```

```
<p>This is my very first HTML page.<p>
```

```
</body>
```

```
</html>
```

[Click here to view the sample page.](#)

Notice I only used the tags I showed you on this page. Yes, it's a simple page, but you're just starting out. Notice the <HTML> and </HTML>. Notice the <TITLE> and </TITLE>. See how there's a beginning and end tag when I alter the text and that the P and BR commands are used to go to new lines?

Look at the program above and then what it produced. Look at the source code when you open the page. See how the HTML tags denoted where text was affected? Good! I knew you would. Now go! Go into the world -- or at least to your text editor -- and create. Follow the instructions in [HTML Primer #1](#) to help you save and then display your first HTML page.

You Can Do This!