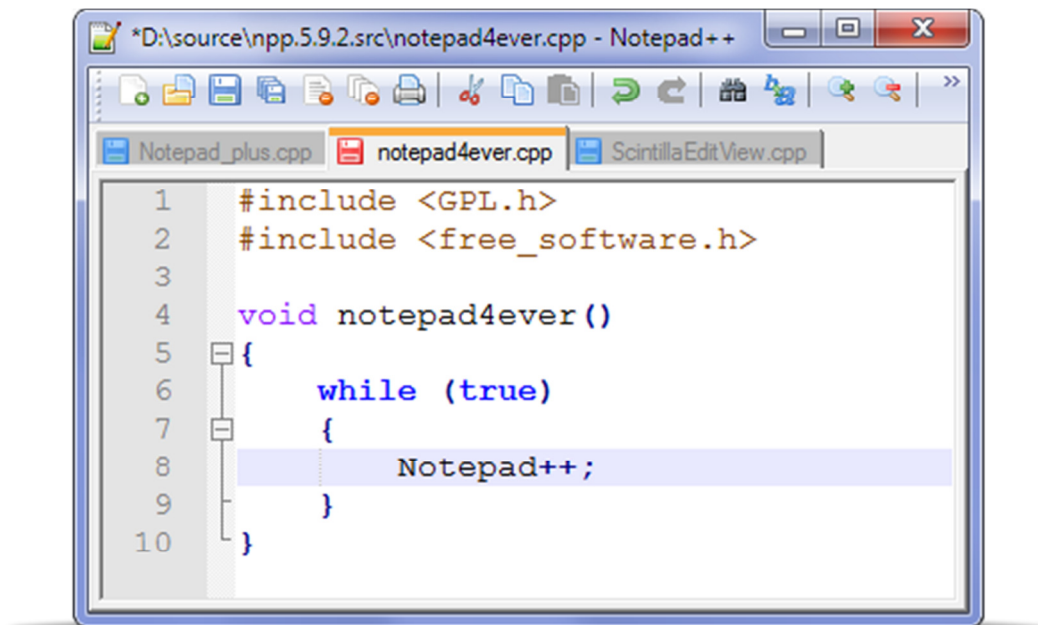


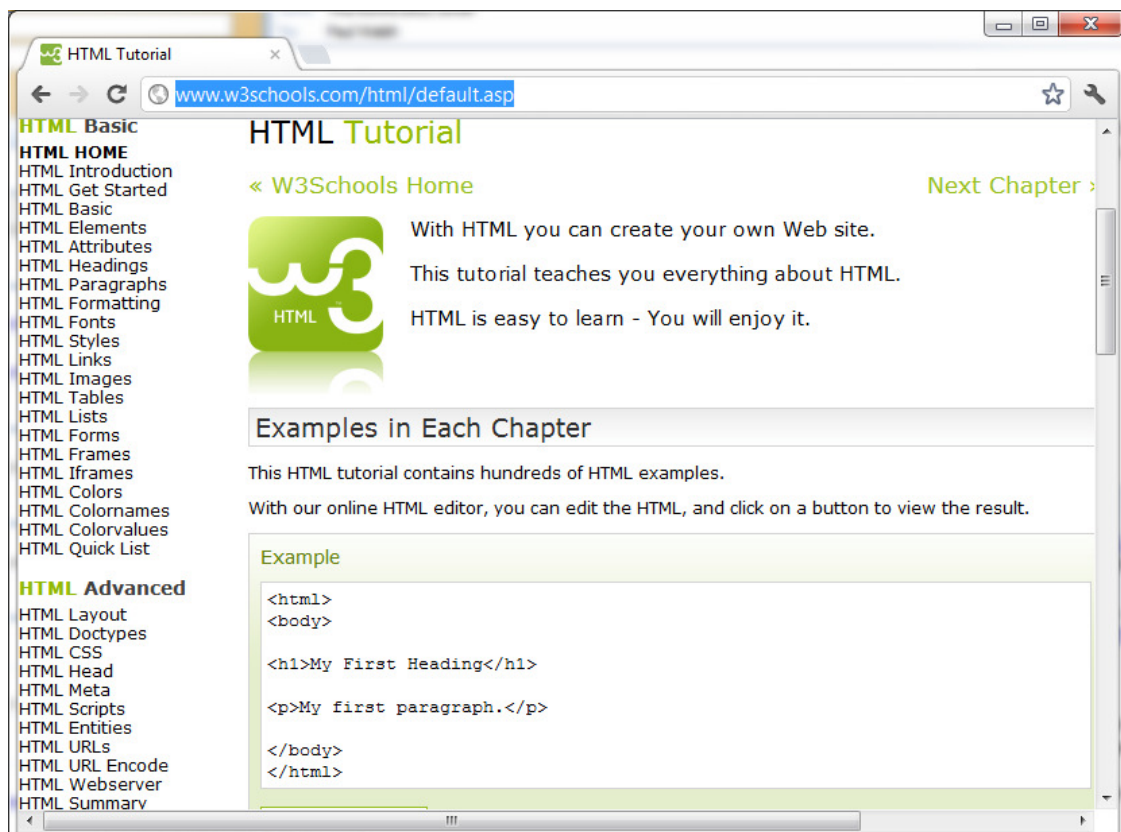
Introduction to HTML 1

To get started.....

Download and install Notepad++ as we will use this to write our code: <http://notepad-plus-plus.org/>



You can use <http://www.w3schools.com/html/default.asp> for reference



All web pages have the same basic structure

The image shows a screenshot of the IMDb website for the movie "Harry Potter and the Philosopher's Stone (2001)". A vertical red line on the right side of the page is annotated with labels: "Head" at the top, "Title" below it, "Heading h1" further down, "Body" in the middle, and "Image" at the bottom. The page content includes a navigation bar with "IMDb" logo and search bar, a "Download the free IMDb Android app" banner, and a "Most-viewed titles on IMDb" section. The main content area features the movie's poster, title, year, genre, and a synopsis. It also displays a "Your rating" section with a star rating of 7.2/10, a "Watch Trailer" button, and a "Watchlist" button. The bottom of the page shows a row of small images related to the movie.

Head

Title

Heading h1

Body

Image

The tools you need

Fortunately, HTML is written in plain text. That means you don't need any fancy software programs like a word processor to create your HTML files. All you need is a simple text editor.

Some rules

As with most things in life, there are rules. In HTML, the rules are fairly simple. For starters, HTML tags are always surrounded by what are called angle brackets < and >. You'll find these brackets on your keyboard just above the comma and period.

Elements

The words/letters between these two angle brackets are called **elements**. These are the coded commands within HTML. Elements tell the browser how to display the web page. For example: <hr> tells the browser to display a horizontal rule;
 tells the browser to skip a line.

Container and empty tags

There are two kinds of tags: container and empty.

The **container tag** always wraps around text or graphics and comes in a set with an opening and a closing:

<html> opening tag
</html> closing tag

Notice the forward slash (/) on the closing tag. This tells the browser that the tag has ended.

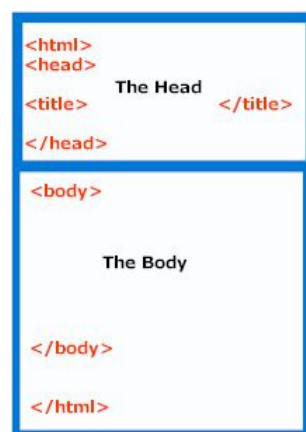
On the other hand, the **empty tag** stands alone. The tag
 is one that adds a line break. Empty tags do not have to be wrapped around copy and do not require a closing.

Case sensitive

HTML is also **not** case sensitive. That means, you can use either lowercase or uppercase. <HTML> is the same as <html>. For consistency, use either one or the other. It's best not to mix and match. For our purposes, I have written our code in lowercase.

HTML structure.

All HTML documents are divided into two main parts: the head and the body. It goes something like this:



You must have the <html>, <head> and <body> container tags in every HTML file.

The <html> tag tells the browser that this is an HTML document. You must begin and end your files with this tag. The <head> tag contains general information like the title of your document.

The <body> tag holds all your content: words, pictures, artwork and other stuff.

When writing HTML, you add "tags" to the text in order to create the structure. These tags tell the browser how to display the text or graphics in the document. For example, the following document has a simple layout (structure). Notice there are three major parts: a heading, two paragraphs and a marquee.

HEADING

Why I like to go swimming in the summer.

PARAGRAPH

Swimming is my most favorite activity in the summer. When the sun is shining and the air is warm, you will find me dipping into my backyard pool. It's not an impressive pool, only three feet deep, but it's mine.

PARAGRAPH

There are three reasons I like to swim:

Marquee

---> Exercise ---> Freedom ---> Fun

To achieve a similar layout in a WWW browser, you need to add tags. Here is the same document with HTML tags (red) added:

```
<html>
<head>
<title>Why I like to go swimming</title>
</head>
<body>
```

HEADING

```
<h1> Why I like to go swimming in the summer. </h1>
```

PARAGRAPH

```
<p> Swimming is my most favorite activity in the summer. When the sun is shining and the air is warm, you will find me dipping into my backyard pool. It's not an impressive pool, only three feet deep, but it's mine.</p>
```

PARAGRAPH

```
<p>There are three reasons I like to swim:</p>
```

Marquee

```
<marquee>
---> Exercise ---> Freedom ---> Fun
</marquee>
</body>
</html>
```

Headline tag

In HTML, bold copy is created by using the headline tag. There are six levels of headlines, ranging from **<h1>...</h1>** to **<h6>...</h6>**. Here is an example of the code for *all* the headline sizes:

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

Paragraphs & Line Breaks

To add space between paragraphs you use the paragraph tag:

`<p>...</p>`

This is a container tag and must have a beginning and an ending.

To add a single line of space, you use break tag: `
`

This is an empty tag and stands alone. You can use the `
` tag to insert one or more blank lines.

Horizontal Rule

To create a horizontal line on your page you use the empty tag: `<hr>`

Lists come in a variety of forms with most either numbered or bulleted. The numbered lists are called **ordered lists** and the bulleted lists are **unordered lists**.

Lists are nested. There is a tag that identifies the type of list, like numbered or bulleted. Then within that tag there is another tag that itemizes the list. Maybe some definitions would help.

`...`

The ordered list is a container tag and is used for numbered lists.

`...`

The unordered list is a container tag and is used for bulleted lists.

`...`

The listed item tag is a container tag and is **nested** within the ordered or unordered tags.

Here is an example of the differences between ordered and unordered lists.

An ordered (numbered) list goes like this:

```
<ol>
<li>My first item on the list.</li>
<li>My second item on the list.</li>
<li>My third item on the list.</li>
<li>My fourth item on the list.</li>
</ol>
```

In the browser it will appear like this:

1. My first item on the list.
2. My second item on the list.
3. My third item on the list.
4. My fourth item on the list.

An unordered (bulleted) list goes like this:

```
<ul>
<li>My first item on the list.</li>
<li>My second item on the list.</li>
<li>My third item on the list.</li>
<li>My fourth item on the list.</li>
</ul>
```

In the browser it will appear like this:

- My first item on the list.
- My second item on the list.
- My third item on the list.
- My fourth item on the list.

Color Attribute

For instance, let's say you want to have a green background on your Web page with red text, like for Christmas time. You would type this code:

```
<html>
<head>
<title>Color Page</title>
</head>
<body bgcolor="green" text="red ">
Hello. I am a page that can be used for Christmas.
</body>
</html>
```

With this code, your web page appears like this in your [browser](#).

Notice the attributes: **bgcolor** and **text**. They are placed within the `<body>` tag. There's that nesting thing again. Attributes never stand alone. Instead, they always appear inside a body tag.

Let's look a little closer at the body attribute:

```
<body bgcolor="#bee3c2">...</body>
```

1. The bracket and tag appear first (**<body**).
2. Always add a space between the tag and attribute.
3. Then enter the attribute (**bgcolor**).
4. Equal sign goes next (**=**).
5. Next are quotation marks that contain a description of how the attribute should look like (**"#bee3c2"**). In this case, it's a code for the color green.
6. Close with a bracket (**>**).
7. Then, add your closing tag **</body>**.

Align Attribute

Another attribute that comes in handy is: **align**. Often used with headlines and graphics, this attribute let's you place text or pictures to the left, center or right side of the margin.

Here's a simple example for the align attribute:

```
<html>
<head>
<title>My Spring Vacation</title>
</head>
<body bgcolor="black " text="white">
<h4 align="center">My Spring Vacation<br>
by Russ Peabody</h4>
```

```
<p>My spring vacation was wonderful, except for the terrible storms we had during the first part of
the week. Snuggle, my dog, enjoyed the walks we had on the beach and Jimmy, my teddy bear, liked
the hotel room. He said the maid was very nice to him.</p>
</body>
```

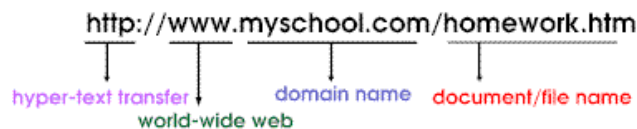
Introduction to HTML Links

Back in 1945, an engineer predicted that someday we would have super computers talking to each other. Today, this vision is a reality through hypertext links (or what others call hyperlinks or just plain links).

Links are the power behind the World Wide Web. Through links, millions of pages, filled with information and knowledge, are only a click away. Every Web site can become a virtual library—a place to learn and grow! And it's all so very simple to accomplish.

Uniform Resource Locator

Before we get into how to create a hyperlink in HTML, we have to cover some "techy" stuff. Every Web site has an address call the **Uniform Resource Locator** or URL (pronounced like "earl"). Think of an URL as an address. If you've done any Web surfing at all, you have probably used an URL or two. But what does it mean? Let's examine one more closely:



http:// (hypertext transfer protocol) is a code or what is technically called a protocol that helps one computer talk to another computer. **www or world wide web** lets the server (the computer) know the file is located on the World Wide Web.

myschool.com is a domain name or where the Web site is hosted. **homework.html** is the file or Web page you are seeking.

Hypertext Link

Okay, now that you understand URLs, you'll find it easier to understand links. There are two things you need to create a link:

1. the name of a file or the URL you want to link to
2. the link hotspot—the highlighted text or graphic that will be clicked on to get to where you want to go.

To create a link in HTML, you need the **anchor** tag: `<a>...`

Inside the tag, you need the attribute: **HREF** (hypertext reference).

An example of a link looks like this:

Harry uses the browser called [Netscape](#).

The code looks like this:

Harry uses the browser called

```
<a href="http://www.netscape.com">Netscape</a>.
```

Let's take a closer look at the code:



Notice that the anchor tag must surround the hotspot and that the attribute HREF must describe where the browser should look for the Web site (the URL).

The Image tag

To place an image onto a Web page you will need to use the image tag. There are two very important things to remember about the image tag:

1. It's an empty tag (remember, that means there's no closing tag, only a beginning).
2. It requires attributes to be effective.

The image tag `` has several attributes:

src--identifies the image and tells the browser where to get the image

alt—gives alternative text for those who are not viewing their pages as images.

height & width—tells the browser the size of the graphic (speeds up the downloading process)

Let's go through each one of the attributes in more detail.

SRC attribute

To help the browser identify and find an image, you use the following command:

```

```

I'd like to cover a couple of important things here. While the name of the file is important, it is equally important to direct the browsers to where the file is located. The browser will not show your image if it can't find it.

Alt attribute

Sometimes people turn off their image loading option in their browser, leaving a Web page with lots of blank boxes. So all the work you have done to place those important images is gone, leaving in its wake only blank, boring boxes.

The **alt attribute** lets the user know what's in the picture by providing descriptive text.

The image tag for this graphic looks like this:

```

```

Height & Width attribute

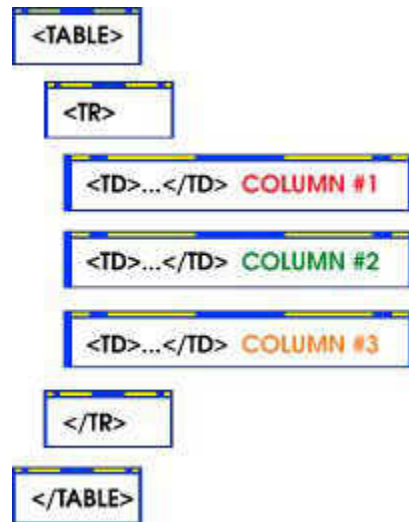
When placing your images onto the screen, the browser looks for the height and width of your images. This is an excellent feature because it allows the browsers to set the space size aside while laying out the balance of the page. If you do not specify the size, the browser will stretch it or reduce it to fit.

Height and width are listed in pixels--those little dots that make up an image on a computer screen.

Building Web pages with tables.

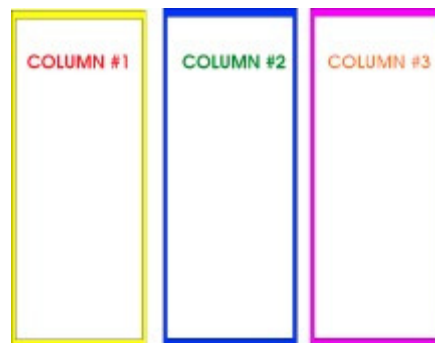
Tables have literally changed the look of the Web page. Originally, tables let people present data (mostly numbers) in a column format. Designers quickly figured out ways to improve the layout of their pages using tables. Although a bit complicated and sometimes difficult to understand, tables do improve your layout and let you present your material in a more eye-appealing way.

For example, this page--in fact, this whole tutorial--has been developed using a four-column table. The best way to understand tables is to look at the following illustration.



The structure of a table

Every table is wrapped by the `<table>...</table>` tag. Then, the columns are surrounded by the `<tr>...</tr>` tag. You can have one column or as many as you want. In the above illustration, we are looking at three columns. Now each column has its own tag: `<td>...</td>`. Although these columns appear horizontally in code, they actually represent vertical columns. So the above illustration represents a set of columns like this:



Tables can be very complex. However, we will create a simple column format. Using some of the basic HTML tags you have learned so far, let's create a new Web page with tables.

Step 1 Open up a text editor (remember, Simple Text for Mac and Notepad for Windows).

Step 2 Enter the following code:

```

<html>
<head>
<title> This is a page using tables </title>
</head>
<body bgcolor="ffffff" text="000000">
<h1>A Web Page Using Tables</h1>
<table border="1">
<tr>
<td>This is column one</td>
<td>This is column two</td>

```

```
<td>This is column three</td>
</tr>
</table>
</body>
</html>
```

Notice that I've included a border with a value of 1. When designing a page, it helps to keep the border attribute in the code just so you know where things are. Then after you are pleased with the design, you can take out the border attribute. Some folks like the border on all the time. It's your choice--you're the designer.

Now let's add some copy to our columns. And see what happens. Add the code listed in red.

Step 1 Open the file table1.html

Step 2 Add the code highlighted in red:

```
<html>
<head>
<title> This is a page using tables </title>
</head>
<body bgcolor="#ffffff" text="#000000">
<h1>A Web Page Using Tables</h1>
<table border="1">
<tr>

<td bgcolor="#000000">
<font color="#ffffff">
<b>This is column one</b>
<br>
I enjoy working on HTML code. It gives me a chance to be creative.</font>
</td>

<td bgcolor="#bee3c2"> <b>This is column two</b></td>

<td bgcolor="#ff8000">
<font color="#804000">
<b>This is column three<br>Notice what happens to the column when you
add copy. It gets larger. This can cause a problem. But we do have a way to
control the margins in tables.</b>
</font>
</td>

</tr>
</table>
</body>
</html>
```

Taken from <http://www.goodellgroup.com/tutorial/toc.html>