How to Code in HTML

Summary

Part 1

Hello, and welcome to this extended tutorial on how to code in HTML. This tutorial will be useful when it gets to making advanced effects with your Custom HTML page / HTML file.

1° What is HTML ?

HTML, or HyperText Markup Language, is a Markup language necessary to make Websites and other online features.

We will be learning the basics of HTML, and at the end, you should be able to understand, and hopefully, make your own !

Necessary Software:

1. Notepad / NotePad ++ / TextEdit...whatever. The software must have a simple goal: to enable you to write text!
2. A working internet connection

2° Tags and their attributes

In HTML, anything you would like dispayed needs to be between **TAGS**.

i) Tags

HTML pages are **filled** with what are called tags. These are **invisible on the screen** once the site is made, but they allow the computer to understand what to display.

The tags are easy to spot. They are surrounded by "*angle brackets*", that is to say, '<' and '>' symbols, like this: **<tag>**

What are they for ?

They indicate the nature of the text around them. They mean, for example: "This is the page title", "This is an image", "This is a paragraph of text", etc.. If you feel lost already, just read on, it'll get logical through examples !

There are two types of tags: tags in pairs and orphan tags.

I) Tags in pairs

They open, contain text, and close later. Here's what they look like:

```
<title> This is a title </ title>
```

We distinguish an opening tag (**<title>**) and a closing tag **(</ title>**) indicating that the title ends. This tells the computer that anything that is not between these two tags ... is not a title.

< ! - - Note: Colours here are only used as an indication, real HTML does not have specific colours - ->

```
This is not a title <title> This is a title </ title>
```
This is not a title

II) Orphan tags

These are tags that are most often used to insert an item at a specific location (eg: an image). It is not necessary to delimit the beginning and end of the image, it just tells the computer to "Insert a picture here."

An orphan tag looks like this::

<img />

< ! - - Note that the / end is not mandatory. One could write only <image>. Nevertheless, so as not to be confused with the first type of tag, webmasters recommend to add the / (slash) at the end of the tag. You'll see me put a / for orphan tag and I recommend you do the same, it's a good practice. - ->

ii) Attributes

Attributes are options that you add to tags. They are supplemented to provide additional information. The attribute is placed after the name of the tag, like this:

****

Of course, the example shown here are not complete, as this first section intends to simply explain the syntax

3° The basic structure of an HTML5 page

Returning to our text editor (in my case Notepad + +). I invite you to copy and paste the source code below in Notepad + +. This code corresponds to the elementary code that all web page requires in HTML5::

```
<! DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8" />
        <title> Add page Title </ title>
    </ head>
    <body>


    </ body>
</ html>
```

<! - - I put spaces at the beginning of some lines to "shift" tags. It is not mandatory and it does not affect the display of the page, but you get to make the source code more readable. It's called the indentation. In your editor, just press the Tab key to get the same result. - ->

You'll notice that the tags are **opened and closed in a specific orde**r. For example, the **<html>** is the first one opens, and it is also the last closed. An example:

<html> <body> </ body> </ html>: Correct.

<html> <body> </ html> </ body>: Incorrect

Uh, tis there an explanation for all the tags that we just copied to our editor, Sir?

Of course, do not be impressed seeing all these tags at once, I will explain their role!

I) The doctype

<! DOCTYPE html>

The very first line is called the doctype. It is essential because it indicates that this page is a HTML web page.

It's not really a tag like any other (it begins with an exclamation point).

This line was once incredibly complex. For XHTML 1.0, you had to write:

<! DOCTYPE html PUBLIC "- / / W3C / / DTD XHTML 1.0 Strict / / EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

For HTML5, it was decided to simplify it, to the delight of webmasters. When you see a tag short doctype (<! DOCTYPE html>), it means the page is written in HTML5.

II) The <html>

<html>

</ html>

This is the main tag code. It embraces the entire content of your page. As you can see, the closing </ html> is located at the end of code!

III) The header <head> and body <body>

A web page consists of two parts:

The header <head>: this section provides some general information on the page, its title, the encoding (for the management of special characters), etc.. This section is usually quite short. The information contains the header is not displayed on the page, these are just general information for the computer. However, they are very important!

The body <body> : this is where lies the main part of the page. Everything we write here will be displayed on the screen. It is inside the body that we will write most of our code.

IV) The encoding (charset)

<meta charset="utf-8" />

This tag specifies the encoding used in your HTML page.

Without going into details, as this can quickly become complicated, encoding indicates how the file is saved. It determines how the special characters will be displayed (accents, Chinese ideograms, Japanese, Arabic symbols, etc..).

There are several encoding techniques: `ISO-8859-1`, `775 OEM`, `Windows-1253` ...

However, the most common and universal is: `UTF-8`. This encoding method is used to display, without any problems, **virtually all symbols** of all the languages.

It also requires that your file is properly saved in UTF-8. This is the case most often in Linux by default, but Windows should generally tell the software.

If you have a display problem accents later on your web page, it is a problem with encoding. Make sure the tag indicates UTF-8 and that your file is saved in UTF-8

V) The main title of the page

```
<title> Title Here... </title>
```

This is the title of your page! Every page should have a title that describes what it contains.

It is recommended that the title stays short enough (less than 100 characters in general).

The title does not appear in your page but on top of it (often in the browser tab). Choosing agood title is essential since it is one of the features the browser will look for.

VI) Comments

We have learned to create our first real HTML page in this chapter. Before concluding, I would like to introduce the principle of comments.

Comments in HTMLare strings that simply serve as memos, or information. It is not read by the computer, and it does not change the page display.

A comment looks like this

```
<!-- My Comment -->
```

Comments can be added ANYWHERE in your page.

```
<html> <!-- I believe this is a HTML page -->
    <head>
        <meta charset="utf-8" />
        <title> Add page Title </ title> <!-- Isn't my title Brilliant ? -->
    </ head>
    <body><!-- Beginning of body tag -->


    </ body>
<!-- End of file --></ html>
```

How to Code in HTML

Part 2

Introduction

Well, the blank page is nice, but your website is likely to have limited success if you leave it to that.

We'll see how to write the content of our web page in this chapter.

Every piece of text must be between tags. These tags tell the computer what the text corresponds to: This is a paragraph This is a title, etc..

We will discover many HTML tags in this chapter. Some have existed since the very first version of HTML, others have more recently been introduced in HTML5.

We will see successively in this chapter:

- How to write paragraphs.
- How to structure your page with titles.
- How to give importance to certain words of your text.
- How to organize information in a bullet point list.

1° Paragraphs

I) Paragraphs

The most common tag in HTML is the paragraph tag ** **

Its a tag in pair, which means that the text must be between ** ** and ** **

Here's an example:

```
<p> Hello and Welcome to my site </p>
```

** ** means "**Begining of paragraph**"

** **means "**End of paragraph**"

Remember: Every code you write must be inside the ** ** tags so as to be visible.

The folloing code is the minimum required for any webpage

Let's write the complete code:

```
<html>
    <head>
        <meta charset="utf-8" />
        <title> Paragraphs </ title>
    </ head>
    <body>
  <p> Hello and Welcome to my site </p>
    </ body>
</ html>
```

Now that we know how paragraphs work, let's move on !

II) Skip a line

As you may have noticed, everything you wrote is on the same line, which means that, esthetically, it isn't brilliant. To be able ot skip a line, frenetically clicking the Enter key does not work (try it out !)

Instead, there are two common methods to do so:

- Use several paragraphs
- use the orphan tag **
**

We will be using the second tag, since you already know the **<p>** tag !

To use it, you must put it IN a paragraph, at the area you want to go back to the line; common sense really...

Here's how to useboth in code:

```
<html>
    <head>
        <meta charset="utf-8" />
        <title> Line Breaks </ title>
    </ head>

    <body>
        <p>
            Hello and welcome to my website! <br />
            This is my first test, so please bear with me, I am slowly learning how it works.
<!-- Using the <br /> tag ! -->
        </ p>

        <p><!-- Using the <p> tag ! -->
            For now it's a bit empty, but come back in 2-3 days when I have learned a few more
things, I assure you that you will be surprised!
        </ p>
    </ body>
</ html>
```

You can theoretically put more <br /> tags in a row to make several line breaks, but we consider it a bad practice that makes the code difficult to maintain. To be able to design the page with more efficiency, you will need CSS, coding language that complements HTML and that we will talk of a little later

So what have we learn so far ?

<p> </ p>: to organize text into paragraphs.

**
**: to go to the line.

Now that we know how paragraphs work, let's move on to Titles ! Do not hesitate to re-read this section if you are lost. This is the **FUNDAMENTAL** aspect of HTML: it's **syntax**

2° Titles

When your Buzztouch HTML page will expand, it will get quite complicated to find what you want quickly. This is why titles are so useful.

In HTML, we are very lucky to have a wide variety of titles. There are 6 different titles:

1. **<h1> </ h1>**: Very important title !
2. **<h2> </ h2>**: Important title
3. **<h3> </ h3>**: Usually a sub-title
4. **<h4> </ h4>**: a Little header
5. **<h5> </ h5>**:Not important title
6. **<h6> </ h6>**: Title with hardly any value

Caution: Do NOT confuse these title tags with the **<title>** tag !!! Indeed, the **<title>** is what will be shown in the tab (See Part 1) on your browser. The **<h1> </ h1>** and all the others are to be displayed inside your webpage, in othe words, on the same level as your paragraphs.

Here is an example of the "good" use of titles on a web page.

```
<html>
    <head>
       <meta charset="utf-8" />
        <title> Use of titles </ title>
    </ head>

    <body>
<h1>Biggest </ h1>
<h2>Bigger </ h2>
<h3>Big </ h3>
<h4>Small </ h4>
<h5>Smaller </ h5>
<h6>Smallest </ h6>
    </ body>
</ html>
```

Obviously, the current styling is very poor. When we get to CSS, we will be able to make anything you want (almost !)

3° Adding Importance

I) Adding importance with

To put some value in your text, you can use the tag ** </ em>**.

The use being so self-explanatory, I believe an example is better than words

```
<body> <!-- I have omitted the html, head, doctype tags. They are recursive, and just make
th edisplay more overloaded-->

    <p>

        Hello and welcome to my website! <br />


        This is my first test, <em> so be indulgent please</ em>, I am learning slowly how it
works.

    </ p>

</ body>
```

As you can see, using the **** tag has the effect of putting the text in *italics*. In fact, it's the browser that chooses how to display the words. The **** tag DOES NOT MEAN *Italics* !I am sorry to insist violently on this, but it is important to remember !

II) Adding importance with

Another way of adding importance to text is to use the **** tag. On most browsers, this tag displays the text in bold (including BuzzTouch !). Again, watch and learn:

```
<body> <!-- I have omitted the html, head, doctype tags. They are recursive, and just make
th edisplay more overloaded-->

    <p>

        Hello and welcome to my website! <br />


        This is my first test, <strong> so be indulgent please</ strong>, I am learning slowly
how it works.

    </ p>

</ body>
```

III) Adding importance with <mark>

The **<mark>** tag is the third and final way of making text important. This tag usually highlights the text it defines, making it stand out. Like the two others, this tag has the exact same syntax:

```
<body> <!-- I have omitted the html, head, doctype tags. They are recursive, and just make
th edisplay more overloaded-->

    <p>

        Hello and welcome to my website! <br />


        This is my first test, <mark> so be indulgent please</ mark>, I am learning slowly how
it works.

    </ p>

</ body>
```

Notice: At the momet, we are learning HTML. HTML is a markup language that defines the **Content** and not **Form.**

Therefore, these three tags, being HTML tags, add importance to the text. The display depends on the browser.

I might sound tedious, but it is very important that we understand well. It is common for beginners to reach this stage, and say they know

how to change the display of text with HTMLe. They saw the tags **, , <mark>** ... and they say: "*Cool, I found out how to italicize, bold, and how to highlight text in HTML*".

And yet ... this is not what these tags are for !

I know, I know you will say "*Yes but when I use the text appears in bold, so it's to bold*.".

The role of tags is to indicate the meaning of the text. Thus, **** tells the computer "This text is important." That' is very important for search engines, who look for key words in texts and web pages.

If you do not seize the concept now, you might understand when we learn CSS ! But for now, let's move on...

4° Bullet points

Bullet points are very useful for basic structuring of a page. There are two types of bullet points:

- `Unordered lists`

1. `Ordered lists`

I) Unordered list

An unordered list looks like this:

- Strawberries
- Raspberries
- Cherries

To create an unordered list, it is very simple. Firstly, we start with the **<ul>** tag

```
<ul>

<!-- List will appear here -->

</ ul><!-- This pair tag is the 'frame' of the list -->
```

And now, to add a bullet point

Using the <li> tag

```
<ul>

    <li> Strawberries </ li>

    <li> Raspberries </ li>

    <li> Cherries </ li>

</ ul>
```

Keep in mind these two tags:

** </ ul>** defines the whole list.

** </ li>** defines a list item (a chip).

You can put as many items as you want in the bulleted list, you are not limited to three items of course.

And now you know how to create an unordered bulleted list! Not that hard once you understand how to nest tags.

For those who need to make complex lists, know that you can nest unordered lists (create a bulleted list into a bulleted list). If you want to do this, open a second beacon **<ul>** within an ** </ li>**.

If you close the tags in the right order, you will not have a problem. Attention, however, this technique is a bit complicated to master.

II) Ordered list

An ordered list works the same way, only one tag changes: **<ul>** becomes **<ol>**

Within the list, nothing is changed: it still uses **<li>** tags **</ li>** to delimit pieces.

However, in an orderedlist, position is important. The 1st <li> will correspond to 1.

As it is particularly intuitive, I will let you admire the simplicity of this example:

```
<h1> My Day </ h1>


<ol>

    <li> I get up </ li>

    <li> I eat and drink </ li>

    <li> I go back to bed </ li>

</ ol>
```

Compared to the previous example, all we had to change was the **<ol>** tag.

For your information, there is a third type of list, much rarer: a list of definitions. It involves **<dl>** tags (to delimit the list), **<dd>** (to define a term) and **<dt>** (to delimit the definition of this term).

Here is a simple example :

```
<dl>
  <dt>Coffee</dt>
    <dd>- black hot drink</dd>
  <dt>Milk</dt>
    <dd>- white cold drink</dd>
</dl>
```

There we are ! You have finished the Part 2 of our tutorial. Proceed to the next part if you would like to know how links wok !

How to Code in HTML

Part 3

Introduction

In the previous chapter, we learnt how to create a simple HTML page.It is true that it was not outstanding, but it was a real HTML page anyhow.

As you know, a website consists of several pages. HThis chapter will teach you hpw to go from one page to another, using links.
I guess you all know what a link is: it is a text you can click on to go to another page.
There are **three types of links**:

1. Link to another site
2. Link to another page in your site
3. Link to an anchor on your page

1° A link to another site

Links to other sites are easily recognizable, since they are usually in blue and underlines, and have a hand cursor when you hover ontop of them. Here is an illustrated and explained example that will link to the Buzztouch Forum:

Links are quite simple. To indicate a link, just use the tag. However, this tag does take attributes to help the computer know what to do with the link.

Here is a link that leads to the Buzztouch Forum, located at http://www.buzztouch.com/forum/:

```
<a href="http://www.buzztouch.com/forum/"> The Official Buzztouch Forum </a>
```

Complete code:

```
<p>

Hello. Would you visit the Buzztouch <a href="http://www.buzztouch.com/forum/"> Forum ? </a>

</p>
```

This is a good site!

By default, the link appears highlighted in blue. If you have already visited the page, the link is displayed in purple.

We'll see how to change this when we study the CSS.

If you want to link to another site, we just have to copy the address (called **URL**) into http://.

Note that some links may begin with **https://** (secure site) or other prefixes (**ftp://** ...).

The links we have just seen are called **ABSOLUTE** links, because they show the full address. We will now learn how to make a link to another page in the same website.

2° A link to another page of a site

We have just learned to create links to existing websites. But I'm sure you would like to make links between different pages of your site, right?

If you are working on a site right now, you might not have a URL assigned to you, which would mean that linking pages is impossible for the moment. Rest assured, there is a solution.

I) Two pages located in the same folder

To begin, we will create two files corresponding to two **different** HTML pages. As I am very inspired, I suggest you call them **page1.html** and **page2.html**. So we'll have two files on your computer in the same folder.

Creating a link is then extremly easy:

```
<p> This links to <a href="page2.html">page2</a> of my website ! </p>
```

II) Two pages located in different folders

If your link is in another folder, or a sub-folder, just write the path, like this !

```
<p> This links to <a href="/content/page2.html">page3</a> located in the subfolder content </p>
```

2° A link to an anchor

This is when our tutorial becomes interesting for BuzzTouch apps !

An anchor is a kind of landmark that you can put in your big HTML pages.

Indeed, if your page is very large, it may be useful to add links that lead to a section further down the same page so the visitor can jump directly to the part that interests him. This is the case of the summary I have coded for this tutorial (View Code Source to check it out !)

There are 2 things to remember when making an anchor.

1. You need to create the benchmark, or location to which the link will lead to.
2. You nned to create the specific **<a>** link

Let's say I have a summary that will link to a title later on. Without HTML code, it will look like this:

Summary:

1. First Chapter <-- Location of <a> tag that links to "**First Chapter**"
2. Second Chapter <-- Location of <a> tag that links to "**Second Chapter**"

```
1. First Chapter    <-- Benchmark, or location to which the 1st link will send to
```

...

```
<!-- My code here -->
```

...

```
2. Second Chapter    <-- Benchmark, or location to which the 2nd link will send to
```

...

```
<!-- More code here ! -->
```

- To make the benchmark, let's take a look at the code:

<h1 id="firstchapter">1. First Chapter</h1>

The **id** attribute can go into any tag, like <p>, or <ul> or virtually anything !

The name you place for the id is up to you. Add one that is convenient.

- To create the link that will send to our Benchmark

First Chapter

I) Adding tooltip upon hover

You can use the title attribute to display a tooltip when the mouse hovers over the link. This attribute is optional.

Here is an example of the use of a title attribute:

```
<p> This links to <a href="#my_anchor" title="Interesting Topic, Really !">an interesting
topic</a> located lower</p>
```

II) Forcing a link to open a new window

it is possible to "force" the opening of a new window. For this, we will add the attribute target="_blank" inside the <a> tag

```
<a href="http://www.buzztouch.com/forum/" title="_blank"> The Official Buzztouch Forum </a>
```

Depending on the configuration of the browser, the page will open in a new window or tab. You can not choose between opening a new window or a new tab.

III) A link to send an email

If you want your visitors to give you feedback, and you would like them to be able to click on a link to do so, you can use links "**mailto**". Nothing changes at the tag, you simply change the value of the href attribute like this:

```
<p> <a href="mailto:buzztouch@info.com"> Send me an email </ a> </ p>
```

IV) A link to download a file

Many of you are wondering what it's like to download a file ... In fact, we must do exactly as if the download package were a link to a web page, but this time indicating the file name to download. The computer will automatically detect the file to download, and then download it !

For example, suppose you want to download myFile.zip. Simply place this file in the same folder as your webpage (or a subfolder) and put a link to this file:

```
<p> <a href="myFile.zip"> Download </ a> </ p>
```

How to Code in HTML

Part 4

Introduction

Insert an image into a web page? You'll see, it is incredibly easy.

Well, almost. There are different image formats that can be used on websites, and we should not choose them randomly. Indeed, sometimes the images are large to download, which slows down the loading time of the page (much more than the text!).

To ensure that your pages are readable and quick to download, I will breifly explain the different types of images.

1° Types of images

What do you know of image formats?

When you have an image, you can save it to several different "formats". The weight (in KB or even MB) of the image will be higher or lower depending on the **format** chosen, and the **image quality** will change. image quality

I) The JPEG

The images in the JPEG (`Joint Photographic Expert Group`) are widespread on the Web. This format is designed to reduce the size of photos, which can contain more than 16 million different colors.

JPEG images are saved with the extension .jpg or .jpeg.

II) PNG

The PNG (Portable Network Graphics) is the newest of all. This format is suitable for most graphics (I am tempted to say "anything that is not a picture"). PNG has two big advantages: it can be made transparent and it does not affect the quality of the image.

III) GIF

The GIF format is quite old, and extremly heavy. It is depricated

To sum up which to choose:

- **A picture**: using a JPEG.
- **Any graphics with few colors (less than 256):** Use an 8-bit PNG.
- **Any graphics with many colors:** Use a 24-bit PNG.
- **A moving picture**: use an animated GIF.

2° Inserting images

To insert an image, we will be using an orphan tag: **<img />**

It is a type tag orphan (like **
**). That means we do not need to write it in like most other tags we have seen so far. Indeed, we do not need to define a portion of text, we just want to insert an image at a specific location.

The tag must be accompanied by ***two required attributes***:

1. **src**: Defines the image you are going to insert. It works EXACTLY like href, that we used for links. You can either put an **absolute path** (eg http://www.site.com/fleur.png) or a realtive path. So if your image is in a subfolder pictures you would type: src = "images / tree.png"
2. **alt:** it means "**alternate text**". This is not obligatory, but strongly recommended. It displays a text in case the picture cannot load. It could be a warning, or just a description.

Her's how we add an image:

```
<p>
     Here is a picture I took during my recent vacation in the mountains: <br />
     <img src="images/tree.jpg" alt="My favorite tree" />
</ p>
```

Tip: Avoid using complex names for the files and file paths. Avoid using symbols that are not internation (like "é", or "~")

I) Add a tooltip

Similarily to the link tooltip, the image tooltip is displayed using the title="" attribute
Here is another example:

```
<p>
     Here is a picture I took during my recent vacation in the mountains: <br />
     <img src="images/tree.jpg" alt="My favorite tree" title="An epic tree !" />
</ p>
```

Hover over the picture with the mouse to see the tooltip appear.

3° Exercice:

Before this tutorial ends, I would like to give you guys a little task. Using the skills recently acquired, you could build a biasic site with the following instrctuins. If you want me to check on it, just send it to me (information at the bottom of the page !):

The task is the following.

Build a website with 3 pages.

In the 1st page, I would like to have abasic cooking class. Requirements:

- Use lists.
- Send a link to a URL of a professional cooking site (this site for example http://www.smileycooking.com/)
- Use all three kinds of lists.

In the second page, I would like you to make a summary of my tutorials

In the 3rd page, I would like you to create a art gallery. Take a compilation of your three favorite images, of three different formats, and include them in your website.

Make sure the three pages are linked together, so we can go from the cooking class to the photo gallery.

Of course, you are not obliged to do so, you can even do the exercice you like

By Anthony a.k.a Skyr0s. Special thanks to 'site Du Zero', and w3. If you need help, just contact me at this email address

For extra reference, check out this amazing referral site. You have acquired the basics, and the syntax of HTML. You can do almost anything in HTML now. Move on to CSS to enhance your WebSite experience !