HTML, CSS, JavaScript and jQuery

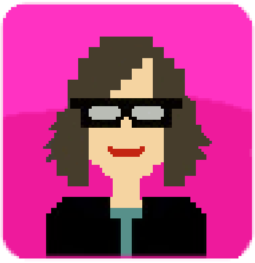
By @codeboom

Contents

Introduction

This is meant to be a quick tour of HTML, CSS, a little JavaScript and jQuery. The booklet was written with Year 12 students in mind, but with a little adaptation I'm sure it would work just as well with most secondary year groups. Please feel free to copy, adapt and distribute as described under this Creative Commons license http://creativecommons.org/licenses/by-nc-sa/3.0/

A quick word about me



I have a degree in Computer Science, and I worked as a professional web developer in 2004 but only for 6 months. I have been using HTML for 12 years, programming for 10 years and I first learnt jQuery three weeks ago. I hope this gives you an idea of what to expect – I'm certainly not a jQuery or CSS guru and there are probably better ways of doing some of the things in this guide. My aim is to get other teachers and students started with the basics so they can do something cool :D

If you need to contact me, tweet me **@codeboom** or email me codergoesforth@gmail.com – I'd love to hear feedback or new ideas!

A quick word about software

You will need to use a text editor to type in all of the code in this document. You could use something really basic like Notepad (or TextEdit on Mac) but it's probably worth downloading a proper code editor, they are invaluable. I like these ones on Windows:

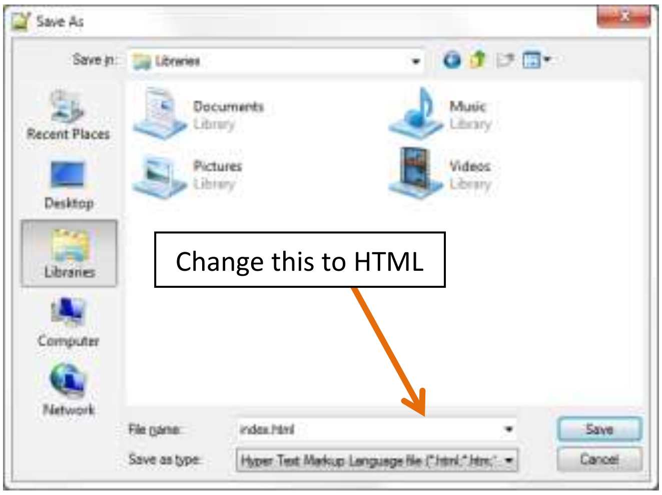
- Notepad++ - free
- PSPad – free
- Dreamweaver (code view) – not free, and a bit overkill, but if you already own it then great!

You will also need an internet browser. I realise that in most schools you don't have a choice and you are probably stuck with Internet Explorer, but if your neighbourhood techies are rather lovely it's worth asking them to install Chrome just for the lovely JavaScript debugging facilities.

HTML

HTML is the language you use to tell a web page what it should look like. You may have covered it before (in which case you could skip this section) or you may need your memory refreshed!

Getting Started



File name: index.html

Save as type: Hyper Text Markup Language file (*.html;*.htm)

Change this to HTML

Start up your favourite text editor (see page 3 for suggestions) with a blank document.

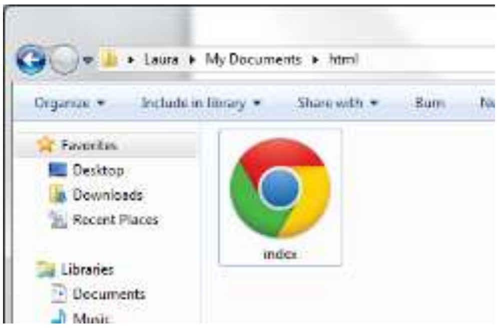
It's a good idea to make a new folder to contain all of your web work.

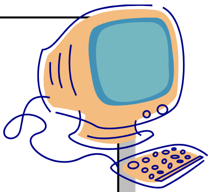
Save your blank document inside this folder as **index.html** – you may need to change the "save as type" to HTML to do this.

The structure of a basic HTML document should start off looking like this:

These are just line numbers – **do not type them in**, they are not part of the code!

```
<html>
<head>
        <title>This is my page title</title>
</head>

<body>
This is where I put the content of my page
</body>
</html>
```





Type this code into your text editor and press save.

Keep your text editor open, and also open the folder where you saved this webpage.

You should see your file - it will probably have the icon of your internet browser. Double click on the file to see your page so far!

Keep both the text editor and the browser open.

HTML tags

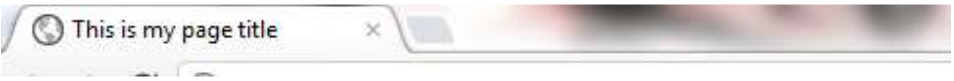
The parts in **bold** type are called HTML ‘tags’

`<html>` This tag tells us that we are starting some HTML code on line 1, and that the page ends on line 9.

`<head>` This tag allows us to put things in for the browser to look at before it begins displaying the page…for example `<title>`

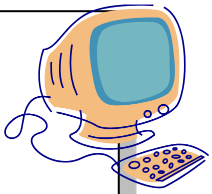
`<title>` Whatever we write between the title tags will appear on the ‘tab’ when you look at it in a browser

Displayed here


This is my page title

`<body>` Whatever we put within the `<body>` tag will be what appears on the main part of the web page

You will notice that to end a tag, we type the same name but with a forward slash /, e.g. `</body>` means this is the end of the body section.



1. Delete *“This is where I put the content of my page”* and write your name on the page instead. Don’t forget that you should write it after `<body>` and before `</body>`. Save the file in your text editor, and then click refresh on your internet browser to see the updated page.
2. Now type in `<b>Hello!</b>` in the body section, save and refresh. What happens?
3. Can you figure out how to add some italic text and some underlined text?
4. What does the `<marquee>` tag do?

Spacing

Leaving a space or moving text onto a new line in your code will **NOT** mean that a space is placed on the page your visitors will see. You must specify where you want spaces using these tags:

`<br/>` Moves on to the next line

`<p>` Leaves a blank line

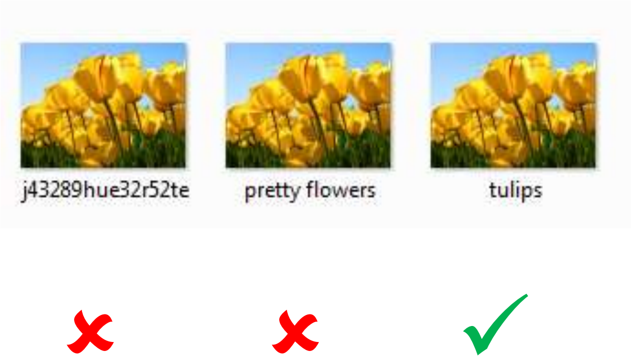
We are being a bit naughty here because really we should use `<p>` like this, to specify text within a paragraph:

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

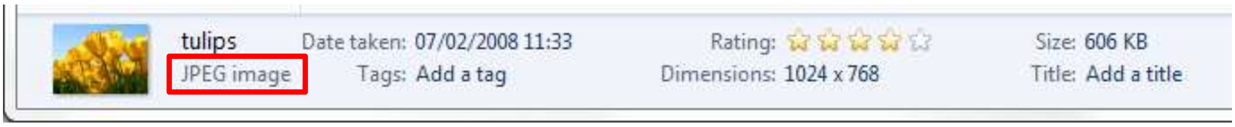
It will still leave a blank line if you only use the opening `<p>` part of the tag though ☺

Images

Let's add a picture to the page! Find a picture on your computer, or search for one on the internet. Save or copy the picture <u>**into the same folder**</u> where your webpage file is saved, making sure that you give it a sensible name with no spaces or capital letters.



Now, check the file type of the image by clicking on it and looking at the bottom of the window:



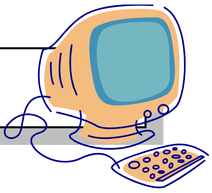
tulips
JPEG image
Date taken: 07/02/2008 11:33
Tags: Add a tag
Rating: ☆☆☆☆☆
Dimensions: 1024 x 768
Size: 606 KB
Title: Add a title

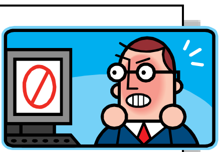
This one is a JPEG image which means it is called **tulips.jpg**
The other usual types are GIF image – **tulips.gif** – or PNG image – **tulips.png**

Here is the code to put the image on your web page:

```
<img src=”tulips.jpg”>
```

Add a picture onto your webpage using the instructions above.





HELP! My picture doesn't show up

- Have you saved the picture into the same folder as the web page?
- Are there any spaces in the file name?
- Are there any capital letters in the file name?
- Did you spell the name of the picture correctly?
- Did you use the correct file type?
- If the picture is a JPEG file, make sure you type .jpg and not .jpeg
- Did you type `img scr` instead of `img src`?
- Can you see anything else that might be wrong with your code?

Annoy your teacher tip #1 – Put up your hand and complain that you are stuck, without even bothering to look at your code and think about what might be wrong. Works every time.

Links

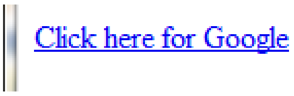
This is the code for a link to Google. The tag is <a>, but it contains something extra called an attribute.

```
<a href="http://www.google.com">Click here for Google</a>
```

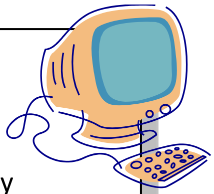
Attribute

href="http://www.google.com" → When this link is clicked, go to http://www.google.com

Click here for Google → This is the text that people click on →



Click here for Google



Add the link to Google as shown above.

1. Now add a link to a different website on your page.
2. Can you change your code so that when someone clicks on a picture on your page, they are sent to Google (or a different website)?

We now know how to add the basic building blocks for web page content – text, images and links. However, our page doesn't look much like something you would see on the web! In order to add colours, sizes, positions and other things to our page, we need to use CSS!

Try the HTML Quiz

1. We start the code with `<html>`. How do we end this tag? ________________

2. Which section comes first, `<head>` or `<body>`? __________________________

3. Which part of this tag is the **attribute**?
   `<img src="car.jpg">` ___________________________________

4. Name three things you should check if you try to put an image on your page but the code isn't working:

   ____

   ____

   ____

5. Which tag do we use to start a new line on the page? _______________________

CSS

With lots of tags, you can add a style **attribute** to change what it looks like. Let's change the background colour of our page:

```
      <body style="background-color: black;">
```

Here is a new tag called <span> which can be used to format pieces of text. The font colour can be changed like this:

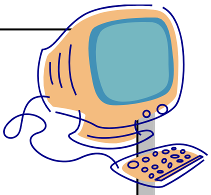
```
      <span style="color: red;">This is red text</span>
```

You can put multiple instructions inside a style by putting a semicolon (;) in between them, for example if you wanted to make red text that was also italic:

```
<span style="color: red; font-style: italic">This is red text</span>
```

There is also an incredibly useful tag we have not yet seen called a `<div>` which creates a box that you can put content in and move about on the page:

```
<div style="background-color: yellow">This is a yellow box</div>
```



Try it yourself!

First of all, type in the styles above and make sure you know what <span> and <div> do.

Now, using the CSS reference on the next page, can you make the following:

1. The word "Peas" in green
2. The word "Tomatoes" in red size 12 Georgia font
3. A box that is 200px high by 100px wide, with the message "Awesome" inside it, and a black border
4. A black box that is 300px square with the message "Difficult" inside it in blinking yellow "comic sans ms" font aligned on the right and a blue 2px solid border.

>_< OMG if you managed to do part 4 of the last exercise you deserve a huge cupcake[1], omnomnom! I bet your style was sooooo long that it went off the end of the page. There must be an easier way of putting in lots of styles, right? RIGHT. Turn to page 10 to find out the clever way.

[1] Unless your teacher is extremely kind, I doubt you'll get one though. Life's not fair, is it? :(

Useful CSS properties

These are some of the most commonly used properties but there are many more, see http://www.w3schools.com/css/ for an exhaustive list.

/* Text inside these symbols is a comment and does not have to be typed in! */

General properties

```
background: #000000;
background-image: url("picture.jpg");
background-repeat: repeat-x;
width: 100px;
width: 50%;
height: 100px;
border: 1px solid #000000; /* width, style and colour */
border: 2px dashed #ffffff;
float: left;
float: right;
margin: 0px;
padding: 0px;
```

Text properties

```
text-align: center;   text-align: left;
font-family: "Times New Roman", Times, serif;
font-style: italic;
font-weight: bold;
font-size: 16px;
text-decoration: underline;
text-decoration: blink;
color: #ffffff;
```

Links

```
a:link {text-decoration:none;} /* normal link */
a:hover {text-decoration:underline;}  /* mouse over link */
```

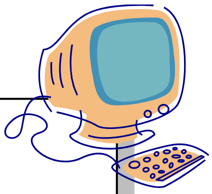
Inline styles are bad!

So you probably found that the answer to CSS exercise number 4 looked a bit like this:

```
<div style=”background-color: black; height: 300px; width: 300px; text-decoration:
blink; color: yellow; font-family: “Comic sans ms”; text-align: right; border: 2px
solid blue”>Difficult</div>
```

That’s 1) rather difficult to find the right bit to edit if you need to change something and 2) really bad if you need more than one black box 300px square with blinking yellow….you get the idea. Using the style attribute is called an **inline style** and it’s not usually a good idea. You can do the same thing a lot more elegantly using a separate file for your styles.

Stylesheet files



1. Make yourself a new blank file. Save it as **style.css**
2. Put this line of HTML code into the head of your document to link to that file
   `<link rel="stylesheet" type="text/css" href="style.css" />`
3. In your **style.css** file, type the following **CSS code** and save it:

```
.myCrazyBox {
      background-color: black;
      height: 300px;
      width: 300px;
      color: yellow;
      /* you can keep typing the rest of the styles here if you like */
}
```

Here we are “defining a class” (basically just specifying a list of styles and giving it the name myCrazyBox)

4. In your **HTML page file** type the following:

```
<div class=”myCrazyBox”>Difficult</div>
```

Here we are applying the “class” myCrazyBox to our element, in this case a div

Now look at your page. You should find that you have the same box, but your code is a lot less confusing!

Now you try creating another different class in your stylesheet and applying that class to a div in your HTML document – pretty easy?

ID vs Class

If we choose to give the element a **class**, we use a dot (.) to define its style in the stylesheet. You can give lots of elements the same class.

```
<head>
   <style type=”text/css”>
       .blueBox {
              background-color: blue;
       }
   </style>
</head>
<body>
   <div class=”blueBox”>This is a blue box</div>
   <div class=”blueBox”>This is another blue box</div>
</body>
```

You may encounter this as another way of defining a stylesheet. If you don’t want to make a separate file, simply use the `<style>` tag and put the classes in the head of your document. Either way is fine.

HOWEVER we can also define styles using an ID.

If we choose to give the element an **id**, then we use a **#** to define its style in the stylesheet. An ID implies that there will be only one of this element on the page.

```
<head>
   <style type=”text/css”>
       #header {
              height: 30px;
       }
   </style>
</head>
<body>
   <div id=”header”>This is the header. There is only one header.</div>
</body>
```

If you know that there will be only one element of a particular style (e.g. most websites have one header, or one footer etc.) then use an ID. If not, use a class.

The distinction between these two is important because of the way that we select elements in jQuery.

JavaScript

What is JavaScript?

JavaScript is a **1.** _________ language which is able to be **2.**__________ by internet browsers. In simple terms, this means that the content of a web page can be changed dynamically according to user input or other variables. It is a **3.**____________ language which means that the code is **4.**___________ on the user's side, as opposed to **5.**_____________ languages such as PHP where the code is executed on the server. It is not the same as Java which is a **6.**___________ language.

Fit these words into the right spaces in the text above:

compiled **server side** **client side**

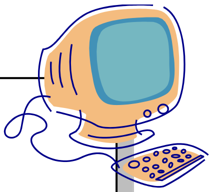
Interpreted **scripting** **executed**

So what can we do with JavaScript? Well, a lot of the things we can do with other programming languages for a start. Our JavaScript code will go inside `<script>` tags, in the `<head>` section of our page.

(Don't make a new `<head>` section, find your existing one and put this inside it. We only want one head, this isn't Classics.)

```
<script type="text/javascript">
      alert("Hi!");
</script>
```

This pops up a box that says Hi inside it. Wow.



I'm not being lazy, but I don't see the point of making a whole huge guide on how to do basic JavaScript when there's a really good interactive one already available. Open up an internet browser, go to www.codecademy.com and follow the first few JavaScript tutorials!

Annoy your teacher tip #2 – Wait until your teacher asks you to email him/her your JavaScript work, then email them a document containing this code – I'll let you figure out why it is so annoying ;)

```
<script type="text/javascript">
  while(true){
    alert("Ha ha!");
  }
</script>
```

jQuery

It’s time to ride with the Big Dog!

The first thing you need to do is to add a line of code to the `<head>` of your document in order to import the jQuery library. I’m using the current version of jQuery hosted by Google. (So make sure you have internet access, or this will not work!)

```
<script type="text/javascript"
src="http://ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js"></script>
```

There are two main things to know about when using jQuery – selectors and events.

Selector – a way of referring to an element or elements on the page

Event – something that could happen to an element on the page, e.g. it could be clicked, or hovered over etc.

An example

Suppose I have the following code for a page with a blue box and a button – most of this should look familiar:

```
<html>
<head>
        <link rel="stylesheet" type="text/css" href="style.css" />
        <script type="text/javascript"
        src="http://ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js"></script>
</head>
<body>
        <button id=”myButton”>Click me</button>
        <div id=”blueBox”>This is a box!</div>
</body>
</html>
```

```
#blueBox{
      background-color: blue;
      height: 300px;
      width: 300px;
}
```

BTW here’s what’s inside my **style.css** document…

When I click the button, I want to make the box disappear. When I click it again, I want to make it appear. I will add this code below the jQuery import line, still inside the `<head>` section:

1. I want to make sure the document is ready before I do anything, otherwise my code may not work:

```
<script type=”text/javascript”>

    $(document).ready(function(){
         /* This is where I write the rest of my code */
    });

</script>
```

I can write all of my code within this ready check, so that I know the whole page has loaded properly.

2. I need to tell the button to check whether it has been clicked on. I have to select the button to tell it what to do. Because I gave the button an ID (myButton), I can **select** the button like this:

```
        $("#myButton")
```

Don't forget it has the # because myButton is an ID

3. Now I want to tell the button to look out for when someone clicks on it, known as a "click event". I can add the click **event** part – so that the page will "listen" out for when the button is clicked:

(For the time being, I will just ask the button to pop up an alert box when it is pressed, so that I can check whether the button is checking for clicks properly before I add more complicated code.)

```
    $(document).ready(function(){
         $("#myButton").click(function(){
               /* When the button is clicked, a box will pop up */
              alert("Is this working?");
         });
    });
```

View your page and click the button. If you get a box popping up, you know that you have selected the button correctly, and created the click event correctly. If not, you may wish to use the JavaScript console in Chrome (Spanner > Tools > JavaScript console) to find out if there is an error.

4. Now let's make the box disappear and reappear. (Get rid of the alert, that was just for debugging.) We can now select the blue box, and use the built in effect `toggle()` on the box to alternate between showing it and hiding it.

```
    $(document).ready(function(){
         /* When I click the button... */
         $("#myButton").click(function(){
               /* ...I want this stuff to happen */
               $("#blueBox").toggle();
         });
    });
```

Refresh your page and click the button – voila!

5. What happens if you change `toggle()` to `fadeToggle()` or perhaps `slideToggle()` ?

Obviously this is just the start of what you can do! I find the jQuery reference section on w3schools to be by far the most useful resource:
http://www.w3schools.com/jquery/default.asp

See next page for the full HTML code for this exercise

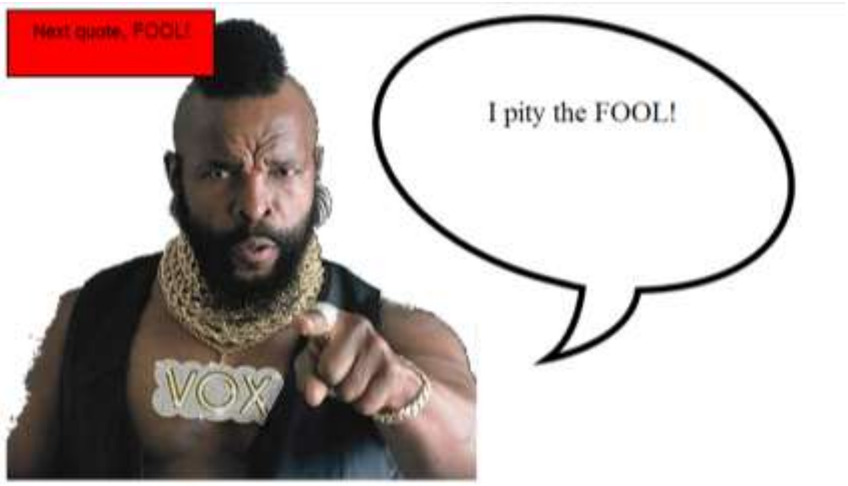
```
<html>
<head>
        <style type="text/css">
                #blueBox{
                          background-color: blue;
                          height: 300px;
                          width: 300px;
                }
        </style>

        <script type="text/javascript"
        src="http://ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js">
        </script>

        <script type="text/javascript">

                $(document).ready(function(){
                        /* When I click the button... */
                        $("#myButton").click(function(){
                                /* ...I want this stuff to happen */
                                $("#blueBox").toggle();
                        });

                });

        </script>

</head>
<body>
        <button id="myButton">Click me</button>
        <div id="blueBox">This is a box!</div>
</body>
</html>
```

jQuery with a purpose

It’s sometimes quite hard to see how this could apply to writing something more substantial. Let’s make a fun app!

The idea

We have a picture of a celebrity with a speech bubble, and a button. When the user clicks the button, a new quote appears in the speech bubble.



Picture shamefully stolen from http://userserve-ak.last.fm/serve/_/27372765/MrT.jpg

How to do it

Easy stuff

1. Find a picture of your favourite celeb
2. Using your favourite image editor, add a speech bubble to the image and save it
3. Create a HTML page and put this image on the page inside `<div id=”picture”>`
4. Add a button

Layering the message above the picture

5. Now add `<div id=”message”>Message here</div>`
6. In the stylesheet, give the **message** div the style
   `background-color: red; position: absolute; z-index: 2;`
   The z-index property is the stack order, so elements with a higher z-index will appear above elements with a lower z-index. We want the message box to appear in front of the picture.

7. In the stylesheet give the **picture** div this style to make it appear below the message
   `position: absolute; z-index: 1;`

Moving the message to the right place

8. Add `top: 100px; left: 100px;` to the style for the **message** div. This should position the div 100 pixels from the left of the page and 100 pixels from the top of the page – which is probably not inside your speech bubble! Alter these values until the message appears in the right place.
9. You may also want to specify height and width properties for the message box to make it fill the speech bubble. Once it is in place, get rid of the `background-color: red` property – this was only so we could see where it was to position it!

When the button is clicked, change the message

10. In the head, start a <script> section and put in the jQuery code to ensure the document is ready. See the exercise we did before for help on this.
11. Inside here, add jQuery code to tell the button to look out for when it is clicked
12. When the button is clicked, we want the message div to change to something else. You can achieve this by selecting the message div, and altering the html inside it:

    ```
    $("#message").html("This is a different quote");
    ```

    Put this code inside the brackets for the part where we tell the button to look out for clicks.

13. This should work…when you click the button, the message should change. It's not very exciting! See if you can figure out how to do these things:

    - Create an array of quotes
    - When the button is clicked, a random quote is selected from the array and appears in the box (hint: try searching for how to select a random item from an array in JS)
    - Have each quote come up in a random colour as well?

For a full solution and more jQuery game ideas including Noughts and Crosses and Pairs, check out http://codeboom.wordpress.com soon! ;)