**HTML and CSS Lab**

**Organizing the content in HTML**

The first thing to do is to distinguish the main blocks on the mock-up. These blocks will form the outline of our page.

To create the outline, we are going to use various HTML tags:

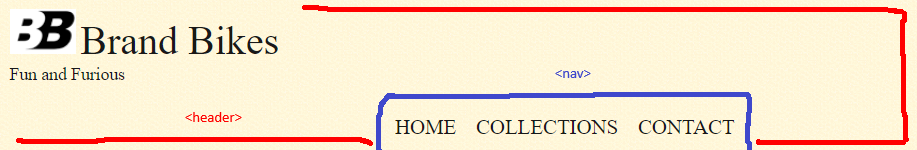
* the HTML5 structural tags, that we know:<header>, <section>, <nav>, etc.;
* the general purpose <div> tag when there is no suitable structural tag.

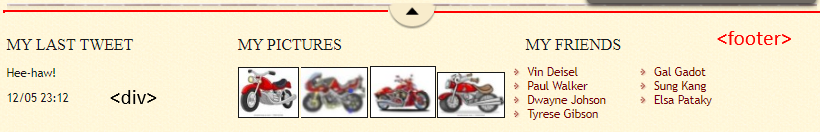
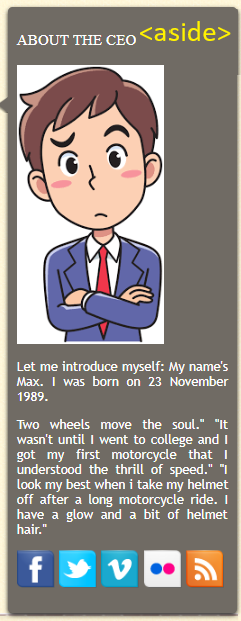
How to know which tag to use?

It’s up to you to decide. Preferably use a tag that makes sense (such as the <header>, <section>, <nav> structural tags), but if no tag seems more appropriate to you, opt for the<div>generic tag.

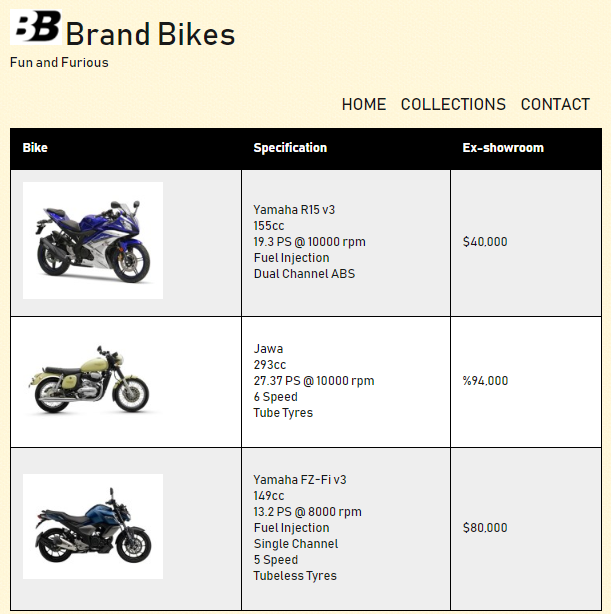
Look at the figure below to a structure.

**Home**

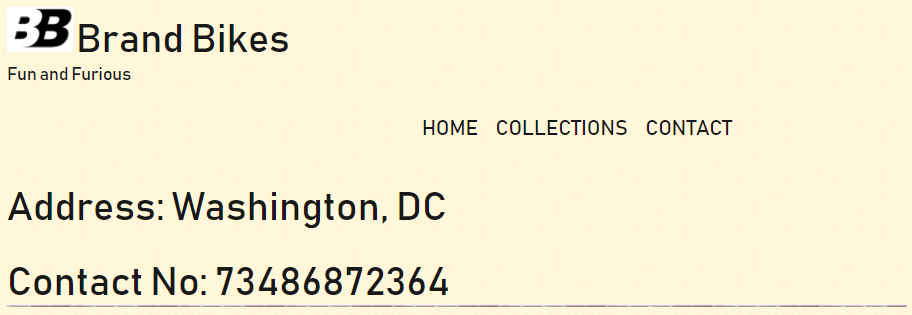


**Collections**

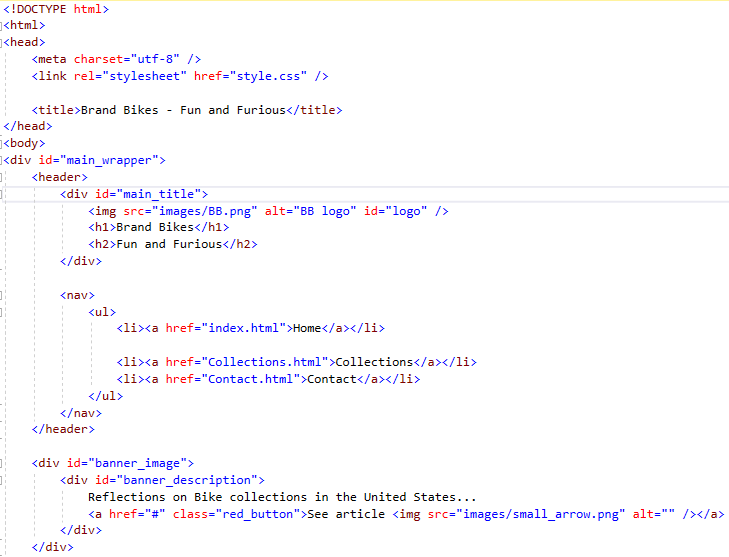


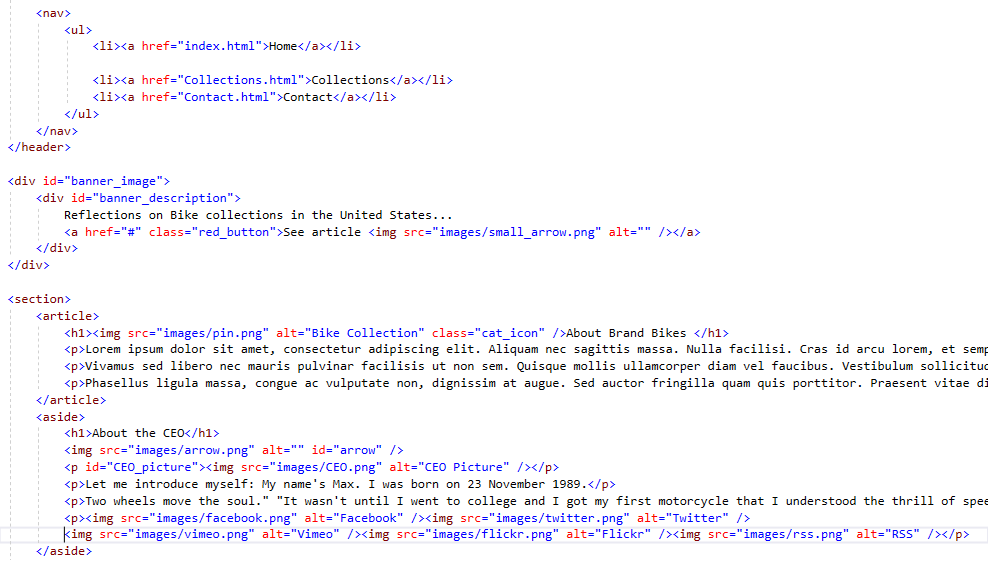
**Contact**



**Note:** Although not all the tags that we're going to use appear in this mock-up, it gives you an idea of the nesting.

**HTML Code: Home**

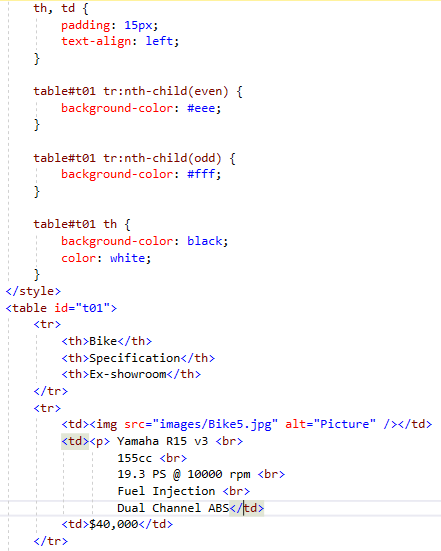


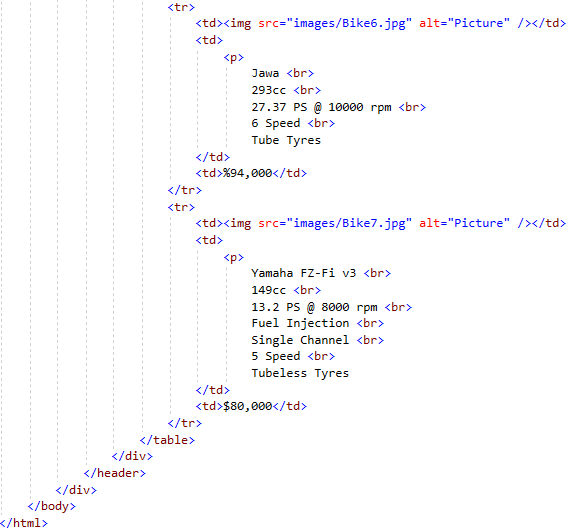




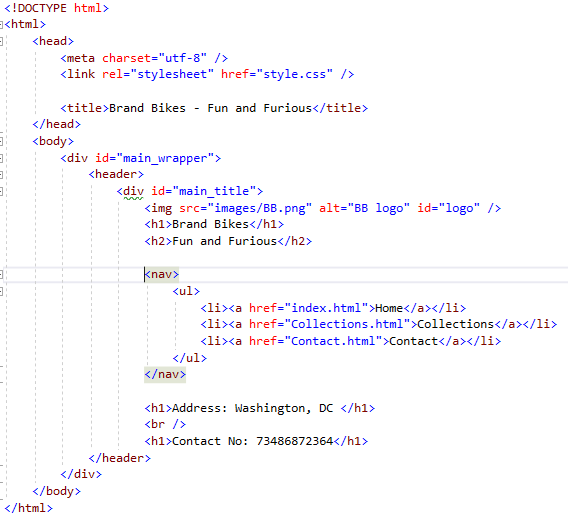
**Collections**





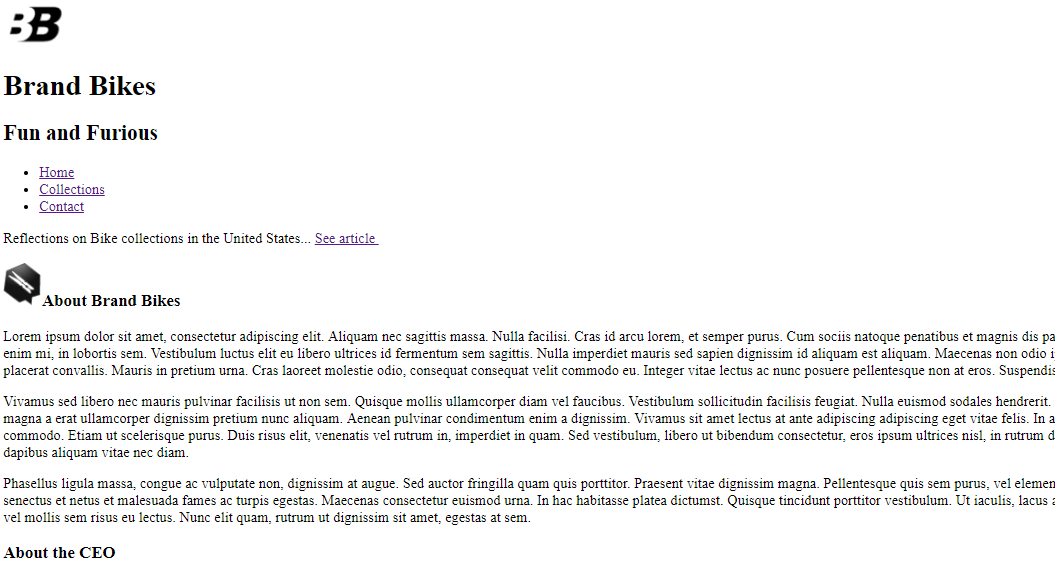


**Contact**



**Note:** As you can see, the whole content of the page is placed in one large<div>tag with a main\_wrapper id. As this tag includes all the content, it will allow us to determine the page size easily and centre our website on the screen.

As you might imagine, the website doesn't look too great yet (to say the least). You can see the current result in the figure below.



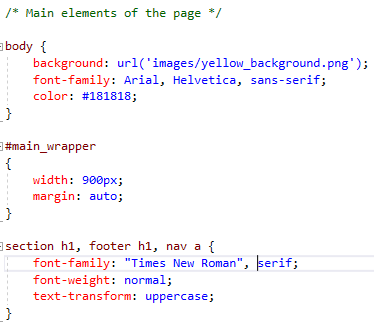
**Formatting in CSS**

To format the design, we’re going to proceed in several steps. we going to deal with the following items, in the order indicated:

* fonts.
* Define the page's main styles (website width, background colour, default text colour).
* Header and browsing links.
* Banner (representing the San Francisco bridge).
* Main section of the page's body, in the centre.
* Footer.

**Defining the main styles**

We can now define a few global styles for the whole design of our page. We're going to define a default background image, font and text colour, and more especially we're going to size our page and centre it on the screen.



With #bloc\_page, the block covers the entire page, so we set the width limits to 900 pixels. With automatic margins, the design will be cantered.

We have used the CSS property text-transform: uppercase;(that we haven't seen before) to ensure that titles are always written in uppercase. Indeed, this property changes the text to uppercase (it can also do the opposite with lowercase). Note that we could also have written the titles directly in uppercase in the HTML code.

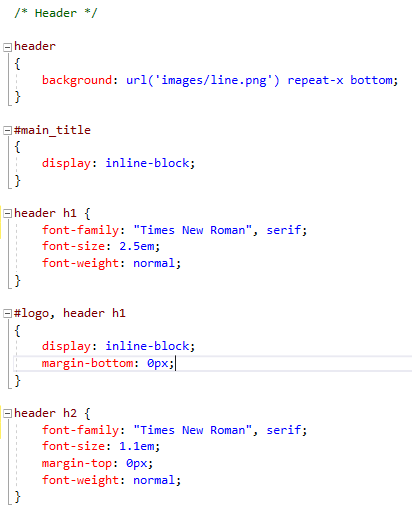
The figure below shows, what you get for now with the CSS code. Although we're still a long way from the result, it already feels a bit like we're "getting there".



**Header and browsing links**

According to the structure we've suggested, the header also contains browsing links. Let's begin by defining the header and the logo at the top left. We'll then see how to format the browsing links.

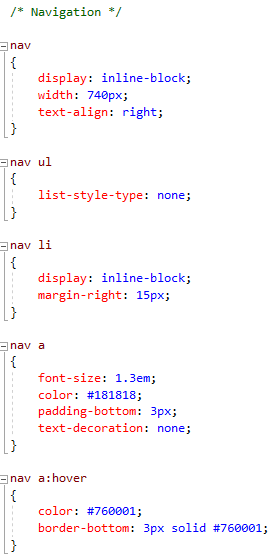
**The header**



We going to make a distinction between the header and the page body with a background image. The items are positioned using inline-block and we're going to customize the fonts and sizes. Nothing out of the ordinary for the time being.

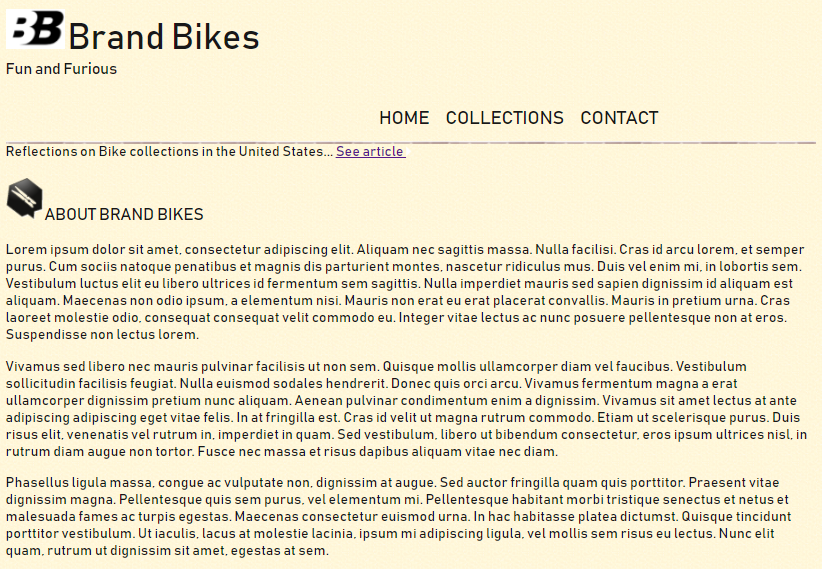
**The browsing links**

The formatting of the browsing links is a bit more interesting. As you saw, we created a bulleted list for the links... but such a list is usually displayed height-wise, not width-wise. Fortunately, it's easy to change, as you'll see:



The main new thing is the CSS definition list-style-type: none; which removes the round image used as a bullet. Each list item (<li>) is positioned as an inline block, which allows us to place the links side by side as we wanted.

The figure below shows the result obtained using the latest CSS additions.

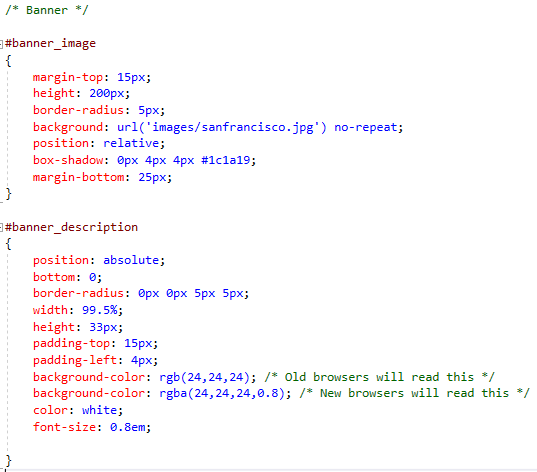


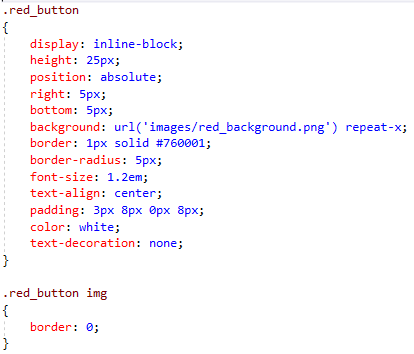
**The banner**

The banner is interesting for several reasons:

* it has rounded edges;
* the description is written on a slightly transparent background;
* the "See the article" button is implemented in CSS, with rounded corners;
* a shadow gives volume to the banner.

Here's the code we used to produce the whole banner:





This code is technical and full of CSS features. It's perhaps the most difficult part in building this page.

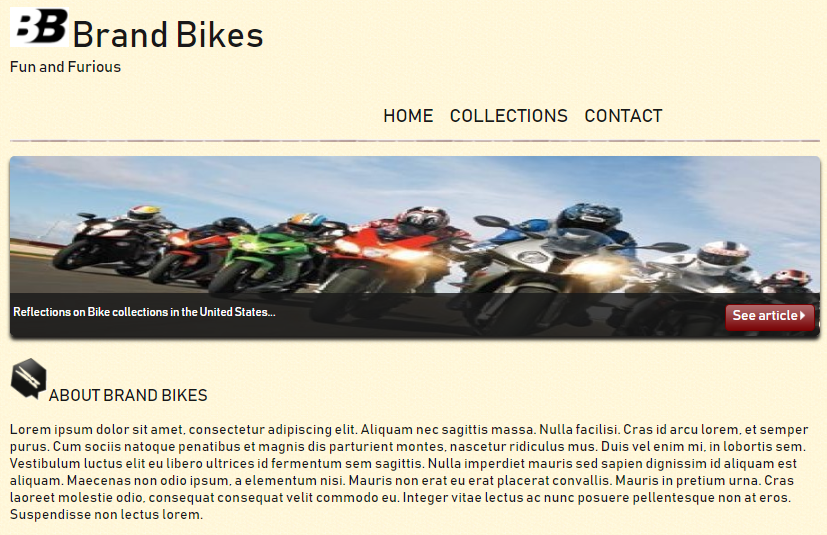
You can see that we've chosen to display the image of the bridge as a background image in the banner<div>block in index.html.

Banner has a relative position (without any offset). As the button has an absolute position inside, it's thus placed at the bottom right of the banner!

This technique is particularly useful and powerful in building a design. and should be remembered!

**Final detail:** for the banner legend, I've chose to use transparency with the RGB annotation rather than the opacity property. Indeed, opacity would have made the entire contents of the block transparent, including the "See article" button inside. I found it preferable to make only the background colour transparent rather than the whole block.

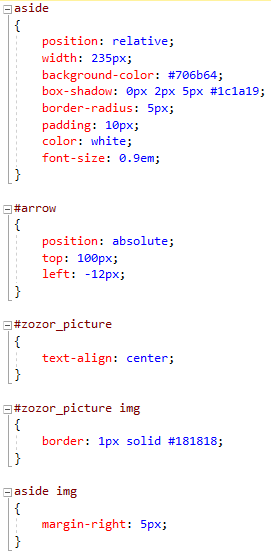
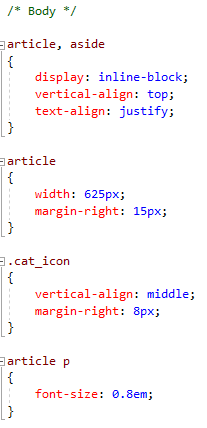
**The result** (figure below).



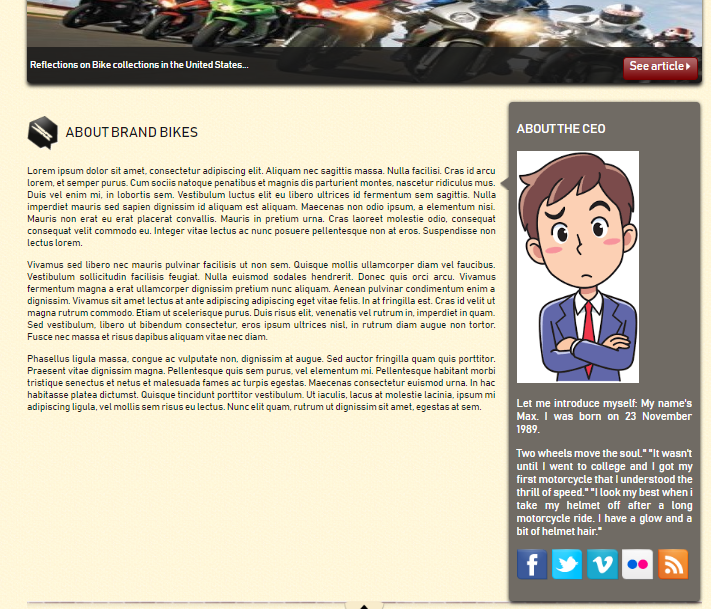
**The Body**

The body, in the centre of the page, in our case consists of a single<section>tag (but there could be several tags, of course).

The "About the author" block is positioned as an in inline block. We'll play with the rounded corners and shadows and adjust the text margins and size a bit, and here we are!

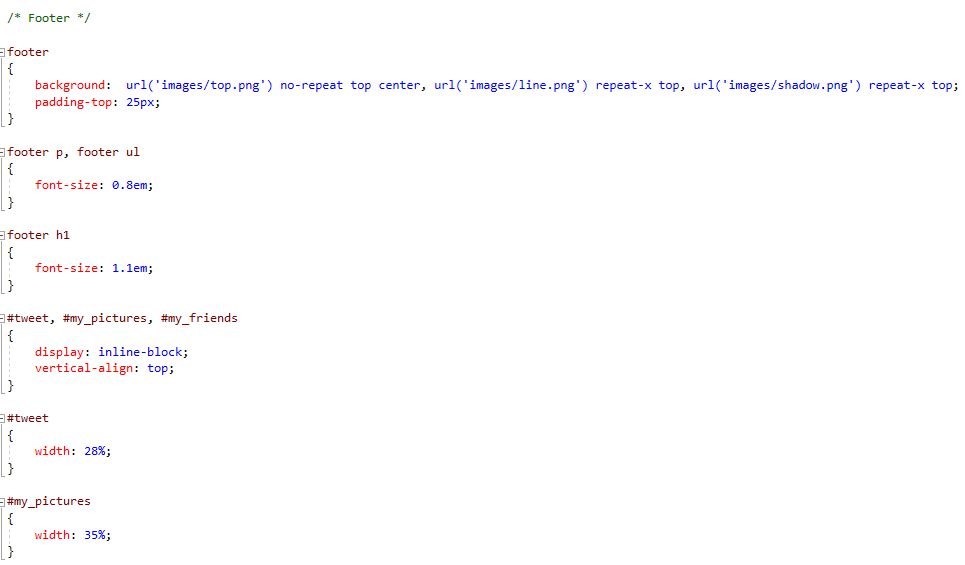


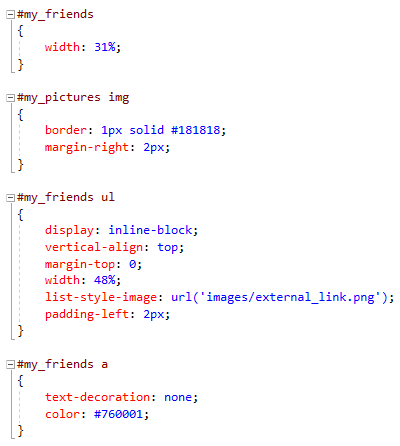
We need to manage to place the arrow to the left of the "About the author" <aside> block to give the effect of a bubble. The technique is the same: we positioned the <aside> block relatively (without inserting an offset), which then allows us to position the arrow image relative to the <aside> block (and not relative to the whole page). By adjusting the image offset, we can accurately place it where want to the nearest pixel (figure below)!



**The footer**

It consists of three sub-blocks that we've produced by<div>to which we've assigned id to identify them better. These blocks are positioned with inline-block side by side.





A couple of small points about the footer:

* We've used CSS3' multiple background images feature to separate the body from the footer. It includes three images: the separator, the small upwards arrow and a slight graduation.
* We've changed the bullet of the "My Friends" list at the bottom right, using the list-style-image property that allowed to use a custom image rather than the standard bullets. There are many specific CSS properties like this one and we can't look at them all one by one in this Lab. But now that you're used to CSS, you'll have no trouble learning how to use them simply by reading the annex listing the main CSS properties.

Our design is now completed.



**Checking the validity**

W3C provides a tool called the "Validator" on its website.

The validator is a type of program that analyses your source code and tells you whether it's correctly written or contains errors you need to fix.

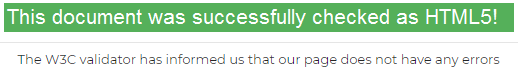
Remember: W3C set up standards that have to be complied with to make sure that all websites speak the same "language".

The HTML validator, however, is going to be useful to us: here's its address [**http://validator.w3.org**](http://validator.w3.org).

You can validate your web page in three different ways, which is why there are three tabs:

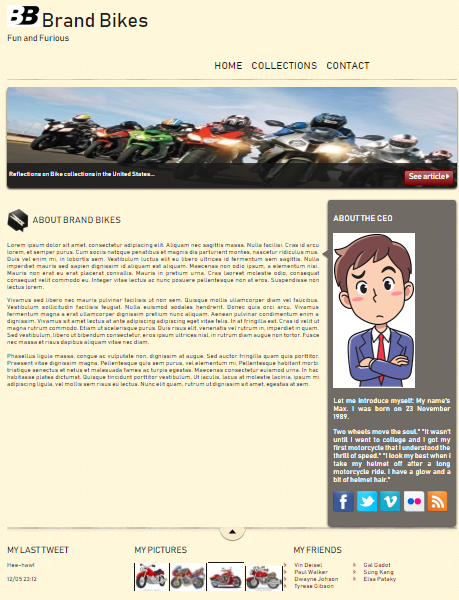
* address (URL);
* send the file.html;
* copy and paste the HTML code.

If you send your HTML code and everything goes well, the validator will reply with the message shown in the figure below.

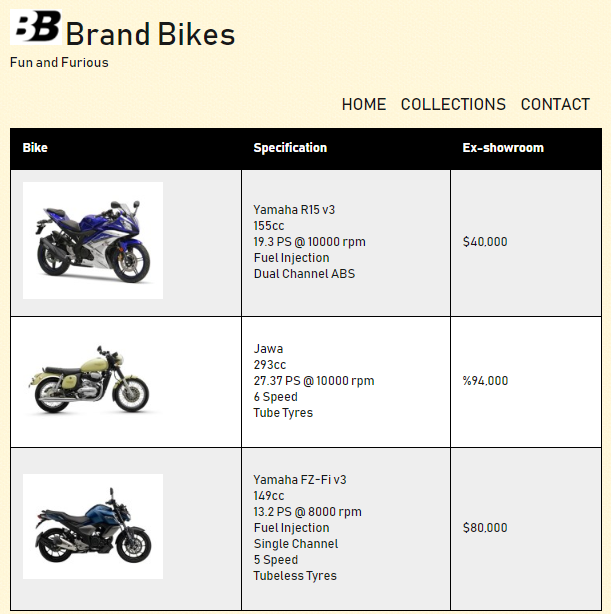


**The Final Code**

We provided the final code of the web page we've built. You can also view the result (figure below) online through a web code.



**Collections**



**Contact**

