## You’ve got the talent

Anybody can build a webpage, the only reason they don’t, or aren’t, is because the skills for showing their talents are handicapped by bad documentation or lazily written tutorials.

To help you bring your web-developer skills up to the level of your creative talents, the following short statements will articulate the relations and purposes of the core web-development tools.

### A webpage is a document, like any text file, that is expressed according to:

* Structure: the rules for organization that are described in the page’s **Hyper Text Markup Language (HTML)** file(s)
* Appearance: Stylistic descriptions contained in **Cascading Style Sheets (CSS)**
* Behavior: when you click on things, type into boxes, hover your mouse over a link, and anything else that causes the page’s state to change, its done according to functionality described in the **JavaScript** **(JS)** files associated with that page.

You can think of html as the webpage’s skeleton, it holds everything in the right place and provides a means of describing context for all of the aesthetic devices shown on the page.

Where html files make the skeleton, the css files are the clothing it wears. CSS files describe color, shape, size and position of elements on a page; the html sets of the rules for how those elements should interact and where their boundaries are.

Think of how your skeleton defines the size and shape your torso; no matter how hard you try, you’ll never be able to properly wear pants on your torso. So, as css defines how your pants and shirt fit on your skeleton, it’s the html that defines the shape and proportionality of that skeleton.

So you’ve got a dressed up skeleton, that’s still not going to get anything done, so you need to put some meat on them bones. That’s where