

Web Development

Bsc Applied, Forensics, Entertainment Systems, IOT

Eamonn de Leastar (edeleastar@wit.ie)
Dr. Brenda Mullally (bmullally@wit.ie)

Department of Computing, Maths &
Physics

Waterford Institute of Technology

<http://www.wit.ie>

<http://elearning.wit.ie>

[e](#)



Waterford Institute of Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

Agenda

- CSS
 - Selector rules and the “Cascade” in CSS



Say you have an `` element inside a paragraph. If you change the background color of the paragraph, do you think you also have to change the background of the `` element so it matches the background color of the paragraph?

The elements inside the `<p>` element inherit the font-family style from `<p>`

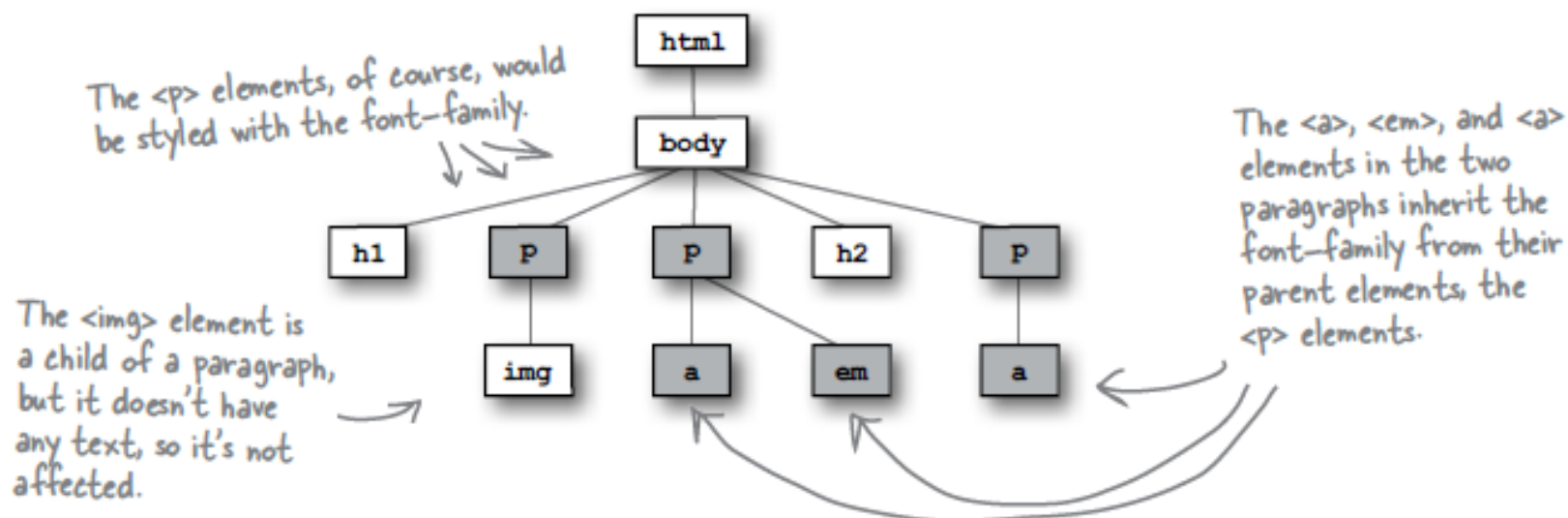
Just like you can inherit your blue eyes or brown hair from your parents, elements can inherit styles from their parents. In this case, the `<a>` and `` elements inherited the `font-family` style from the `<p>` element, which is their parent element. It makes sense that changing your paragraph style would change the style of the elements in the paragraph, doesn't it? After all, if it didn't, you'd have to go in and add CSS rules for every inline element in every paragraph in your whole site...which would definitely be so NOT fun.

Not every style is inherited. Just some are, like `font-family`.

Not to mention, error-prone, tedious, and time-consuming

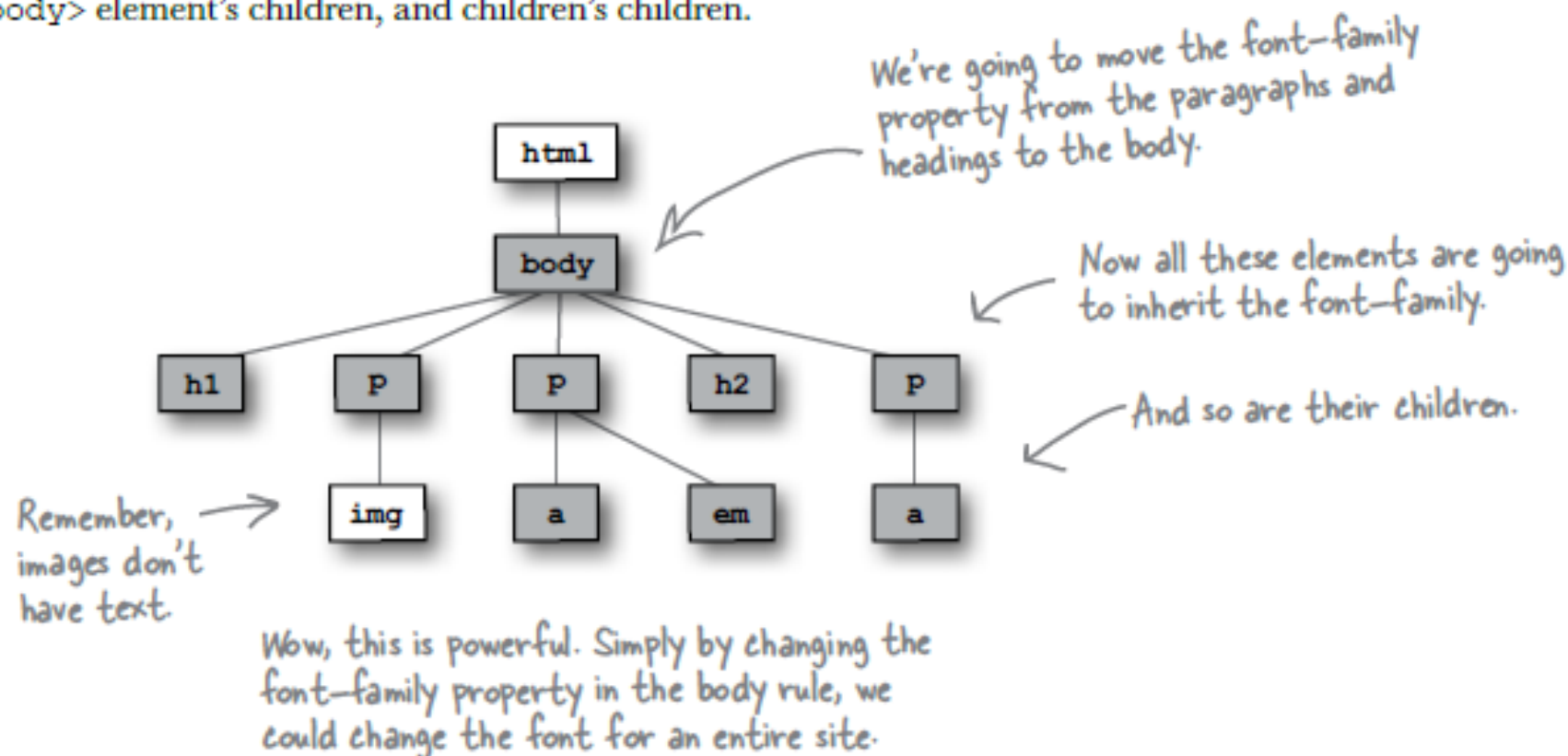
Let's take a look at the HTML tree to see how inheritance works:

If we set the `font-family` of all the `<p>` elements, here are all the elements that would be affected.



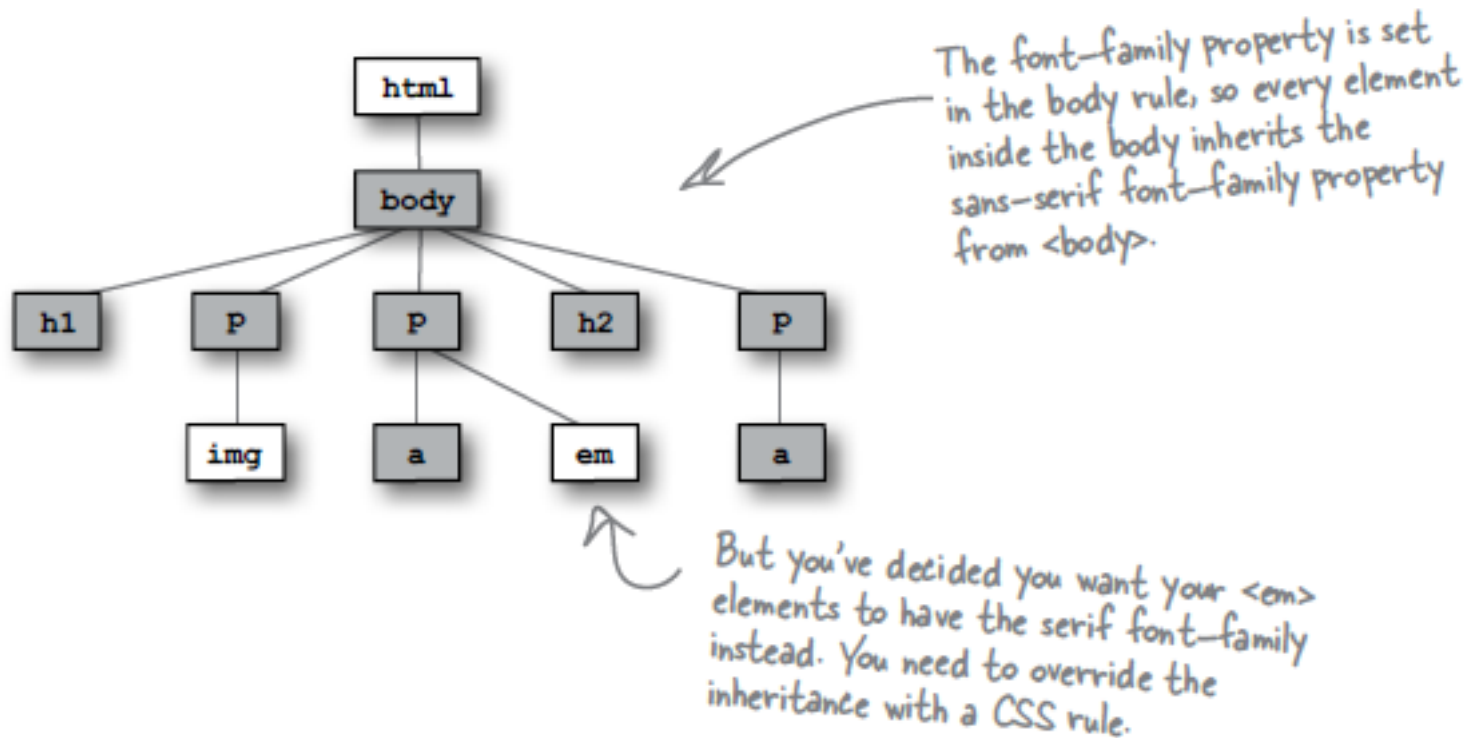
What if we move the font up the family tree?

If most elements inherit the `font-family` property, what if we move it up to the `<body>` element? That should have the effect of changing the font for all the `<body>` element's children, and children's children.



Overriding inheritance

By moving the `font-family` property up into the `body`, you've set that font style for the entire page. But what if you don't want the sans-serif font on every element? For instance, you could decide that you want `` elements to use the serif font instead.



Rules, Classes, Elements - which ones get selected?

(1) *Explicit Match*: Do any selectors select your element?

- Examine CSS rules for *explicit match* for element.

(2) *Inheritance Match*: What if no rules match the element:

- Rely on *inheritance*.
- Look at the element's parents, and parents' parents, and so on, until you find the property defined.

(3) *Default Match*: still no explicit or inherited match

- use the *default* value defined by the browser

(4) *Most Specific Match*: What if more than one match? ?

- select rule that is the *most specific*

Examples

```
p { color: black; }
```

Here's a rule that selects any old paragraph element.

```
.greentea { color: green; }
```

This rule selects members of the greentea class. That's a little more specific.

```
p.greentea { color: green; }
```

And this rule selects only paragraphs that are in the greentea class, so that's even more specific.

```
p.raspberry { color: blue; }
```

```
p.blueberry { color: purple; }
```

These rules also select only paragraphs in a particular class. So they are about the same in specificity as the p.greentea rule.

Specific & Ordering Match

- If we had an element that belonged only to the greentea class
 - there would be an obvious winner:
 - the p.greentea selector is the most specific, so the text would be green.
- But you have an element that belongs to all three classes: greentea, raspberry, and blueberry.
 - So, p.greentea, p.raspberry, and p.blueberry all select the element, and are of equal specificity. What do you do now?
 - You choose the one that is listed last in the CSS file.

(5): *Ordering Match*: - If you can't resolve a conflict because two selectors are equally specific

- use the ordering of the rules in your style sheet file. That is, you use the rule listed last in the CSS file (nearest the bottom). And in this case, that would be the p.blueberry rule.

Example

p	{	color: black;	}
.greentea	{	color: green;	}
p.greentea	{	color: green;	}
p.raspberry	{	color: blue;	}
p.blueberry	{	color: purple;	}

1- Explicit Match

2- Inheritance Match

3- Default Match

```
<p>
  My Normal Tea <br>
  Customers say they <q>really like</q> this one!
</p>
<blockquote>
  All of the best teas
</blockquote>
<p class="greentea">
  My Green Tea
</p>
<p class="greentea blueberry">
  My Mixed Tea - what colour is it?
</p>
```

4- Most Specific Match

5- Ordering Match

p	{	color: black;	}
.greentea	{	color: green;	}
p.greentea	{	color: green;	}
p.raspberry	{	color: blue;	}
p.blueberry	{	color: purple;	}

```

<p>
  My Normal Tea <br>
  Customers say they <q>really like</q> this one!
</p>
<blockquote>
  All of the best teas
</blockquote>
<p class="greentea">
  My Green Tea
</p>
<p class="greentea blueberry">
  My Mixed Tea - what colour is it?
</p>

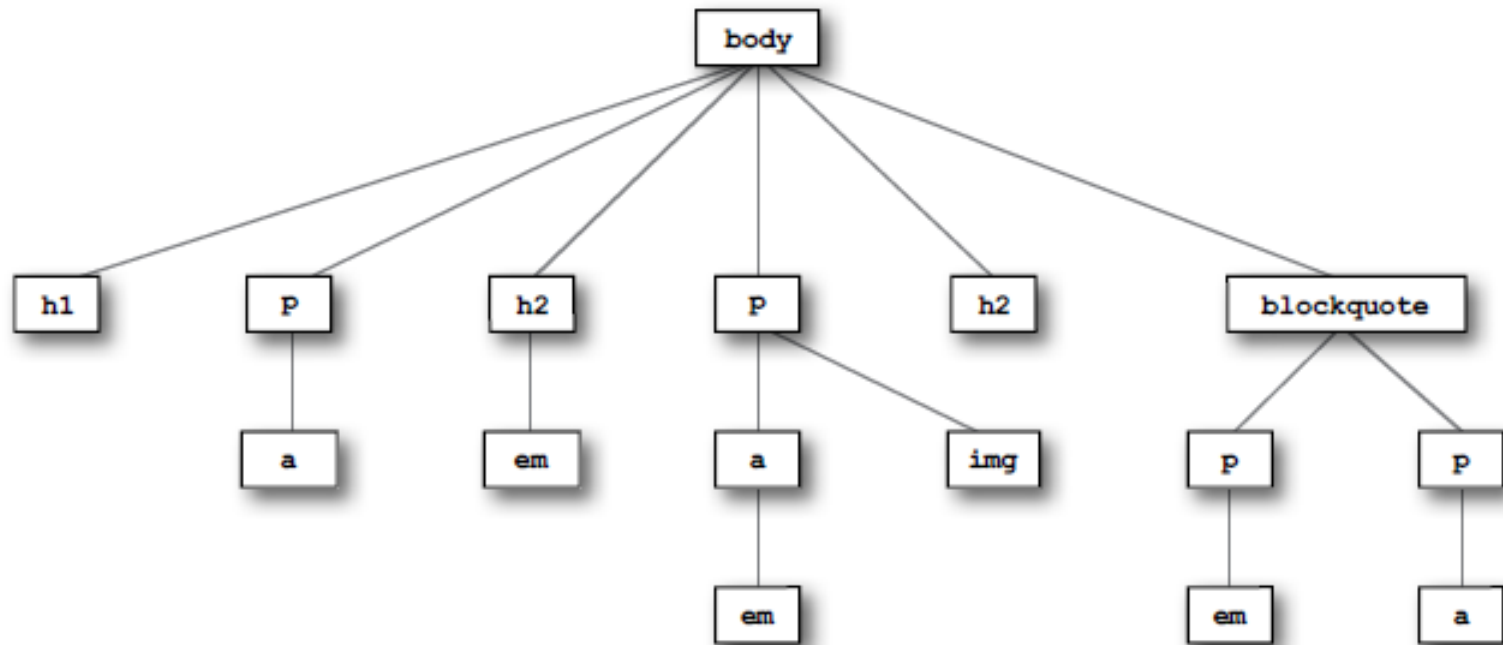
```

My Normal Tea
 Customers say they "really like" this one!

All of the best teas

My Green Tea

My Mixed Tea - what colour is it?



```
body {  
  color: green;  
}  
  
p {  
  color: black;  
}
```

← Here's the CSS. Use this to determine which of the above elements hit the jackpot and get the green (color).

More Advanced Rules

Selector	Applies to
p	All paragraphs in the document
.about	All elements in the document with a class value of about
#corporatehistory	The element in the document with an id value of corporate history (if present)
h1,h2,h3	All first-, second-, and third-level headings in the document
.privacy, .copyright	All elements with a class of privacy or copyright
#header,#footer	The element assigned an id of header, and the element assigned an id of footer
p.footnote	All paragraphs assigned a class of footnote

Selector	Applies to
#bodycopy.usergenerated	An element that has been assigned both an id of bodycopy and a class of usergenerated
.navigation a	All links with an ancestor parent assigned a class of navigation
#primarynavigation li.current	All list items with a class of current and an ancestor parent with an id of primarynavigation
.about #bodycopy	Any element on the site with an id of bodycopy and an ancestor parent assigned a class of about
body#personalproducts, body#proproducts, body#enterpriseproducts	The body elements within the site assigned the ids personalproducts, proproducts, and enterpriseproducts
body#personalproducts #bodycopy, body#proproducts #bodycopy, body#enterpriseproducts #bodycopy	The elements assigned an id of bodycopy, within the documents suggested by the previous example
ol li ol li ol li	A list item in the third level of a nested ordered list