**Zeldman Chapter 12** CSS bugs and CSS3

Again: “Create once, publish everywhere.”

Forward compatibility is Zeldman’s term/campaign [others call that “progressive enhancement” meaning add new stuff but be sure it still is backward compatible.]

**CSS Bugs**: [I would not worry much about these except for the history.]

Note shorthand, p. 232: (“less verbose” as Zeldman says)
background: #fcc url(“alert.png”) no-repeat 0 0.3em;

Same as

background-color: #fcc;

background-image: url(“alert.png”);

background-repeat: no-repeat;

background-position: 0 0.3em; [that’s the X-offset, then Y-offset; recall we used that to position the firefox.gif in the first exercise]

A place to test in old browsers: at [evolt.org](http://browsers.evolt.org/).

double float margin bug,

the star HTML hack \* html {}

How to embed Flash, Quicktime. Used to be you used both <embed> and <object> even though the latter was the W3C standard. One approach is still to use both, another is Google's [SWFObject](http://code.google.com/p/swfobject/) JavaScript which Zeldman vouches for, another is to start using the HTML5 <video> element.

Conditional comments [recommended and Zeldman "can't overstate the value" p. 245]. Only IE understands these anyway. Here's [their page](http://msdn.microsoft.com/en-us/library/ms537512.aspx).

<!—[if lt IE 7]>

<link rel="stylesheet" href="oldiestyles.css" type="text/css" />

<![endif]-->

**CSS3** [see link on Schedule, Extra resume Notes2]

[Note the process for CSS3 is to develop modules, not to wait for a whole new spec to come out. Also the four main browser engines have proprietary extensions (see below) as they test things, which you'll need to use. The hope is that when the W3C formalizes CSS3 those will go away.]

This is only a sample. See my Resume notes and this [link](http://www.webcredible.co.uk/user-friendly-resources/web-dev/css3.shtml).

**1) Alpha channels**

body {

 background: #9178c4;

 color: rgba(30, 40, 30, 0.9)

}

You could always use rgb() for colors—now you need to learn those values. Each color is 8-bit so range from 0 to 255 (sum= 2\*\*8 = 256) and obviously the order is red, green blue + alpha (like alpha-level transparency in Flash, PNGs, elsewhere.

a= 1.0 is fully opaque/not transparent

a= 0.0 is fully transparent

There's also #somediv {opacity: 0.7} on the same model but that makes everything in the element—child elements, etc. transparent/less opaque.

**2) Text-shadow** [**box-shadow** works the same way]

body {

 font: normal 13px/20px "lucida sans", "lucida sans unicode", verdana, sans-serif;

 text-shadow: 0 1px 0 #123456;

}

[Those values are for: **x**-offset **y**-offset **blur** radius which he never explains. Note also font/line-height 13/20 and how quotation marks are placed for font names]

Note also that text-shadow: is generally written without the vendor prefixes, as it’s well supported.

See [10 to 1](http://www.10to1.be/) site for this, also font: and background: shorthands, negative margins, floats, how he comments out his </div> tags, etc. {Q: where's he from? Compare to Sitepoint on [embossed text](http://www.sitepoint.com/using-css-text-shadow-to-create-embossed-text/) techniques.

**3) Rounded corners and vendor/proprietary extensions**

Safari 3+ [and also the rendering engine for Google Chrome] is Webkit: [note the leading hyphen)

#somediv {

 border: 1px solid #123;

-webkit-border-radius: 1em .5em;

-webkit-top-left-border-radius: .7em;

}

For Mozilla (Firefox 3.5+)

#somediv {

 border: 1px solid #123;

-moz-border-radius: 1em .5em;

-moz-border-radius-topleft: 0.7em;

}

For Opera 10+ up to about Opera 20 [it then became –webkit]

#somediv {

 border: 1px solid #123;

-o-border-radius: 1em .5em;

-o-border-top-left-radius: 0.7em;

}

where one value is all four, two values is x/horizontal and y/vertical so you can get elliptical shapes and then you can specify each corner but the syntax varies among rendering engines [notw(for now).

to be safe Internet Explorer is –ms (for Microsoft):

-ms-border-radius: 20px;

When standardized, it should all become

border-radius: 1em;

Others (see Extra resume notes2):

* gradient:
* column-count: column-width: column-gap: column-rule:
* multiple backgrounds
* background-clip
* new selectors nth-child: nth-of-type: etc.
* transition: transform: to create animations
* media queries See [Quirksmode.org](http://www.quirksmode.org/blog/archives/2010/08/combining_media.html)
@media all and (max-device-width: 600px) {
.webstuff {
display: none;}
.sidebar {
float: none;
width: auto;
}
}

And maybe you add some styles just for desktop here as well:
@media all and (min-device-width: 600px) {
 body {
 // some more styles here
 }
}
OR <link rel="stylesheet" type="text/css" href="phone.css" media="all and (max-device-width: 600px)" />

Quirksmode says [here](http://www.quirksmode.org/blog/archives/2010/09/combining_meta.html) you must also add this <meta> tag, for mobile devices:

 <meta name="viewport" content="width=device-width, initial-scale=1">

(to set width to whatever the vendor has determined, and to not allow pinch-zoom)