**Text Formatting in Action**

****

In this tutorial, you’ll add better styles to headlines, lists, and paragraphs of text, using CSS’s

powerful formatting options.

**Text Formatting in Action:**

**Setting Up the Page**

First, you’ll get your style sheet started, add the @font-face directive to load some

web fonts, and add styles to format the body text.

1. **Launch your web browser and open the file text.html .**

It’s not much to look at—just a collection of headlines, paragraphs and a lone

bulleted list—but you’ll turn it into something far better looking.

2. **Open the file text.html in your favourite text editor.**

Start by adding an internal style sheet to this file.

3. **In the <head> of the web page, click directly after the closing </title> tag.**

Hit Enter (Return), and then type:

<style>

</style>

4. **Open the file reset.css.** Copy all the code from that file and paste it between

the opening and closing <style> tags you added in Step 3.

If you preview the text.html file in a web browser now, you’ll see that all of the

text on the page (headlines, paragraphs, and so on) looks nearly the same—in

other words, all the basic HTML formatting the browser applied has been removed,

so you can start with a clean slate.

Next, you’ll add the necessary @font-face directives to load four web fonts.

Actually, they are all the same font—PTSans—but include the bold, italic, and

bold-italic variants.

5. **Open the at-font-face.css file and copy the contenrts into the style block within text,html:**

In a nutshell, you’ve created a new font family named PTSans. You’ll be able to

use it in any new styles you create.

Next, you’ll create a style that defines some general properties for all text on

the page

.

6. **After the four @font-face directives, press Enter, and type html {.**

This is a basic tag selector that applies to the <html> tag. As discussed in an earlier Lab,

other tags inherit the properties of this tag. You can set up some

basic text characteristics like font, color, and font size for later tags to use as

their starting point.

7. **Press Enter again, and then add the following two properties:**

font-family: 'PTSans', Arial, sans-serif;

font-size: 62.5%;

8. **Complete this style by pressing Enter and typing a closing bracket to mark**

**the end of the style.**

At this point, your completed style should look like this:

html {

font-family: 'PTSans', Arial, sans-serif;

font-size: 62.5%;

}

Your style sheet is complete.

9. **Save the page, and open it in a web browser to preview your work.**

The text on the page changes font…it also gets really small. Don’t worry, that’s the

62.5 percent font size you set in Step 7. That’s just the starting point for all text,

and you’ll easily increase the size of text by defining em sizes for the other tags.

**Formatting the Headings and Paragraphs**

Now that the basic text formatting is done, it’s time to refine the presentation of

the headlines and paragraphs.

1. **Return to your text editor and the text.html file.** Click at the end of the closing

brace of the body tag selector in the internal style sheet, press Enter (Return)

to create a new line, and then type .main h1 {.

This is a descendent selector It provides more specific direction than

a basic HTML tag selector. In this case, the selector tells the web browser “apply

the following formatting to any <h1> tag inside another tag with the class

name main.” If you look at the page’s HTML, you’ll see that there’s a <div> tag

with a class of main (<div class="main">). As you’ll learn later, it’s very common

in CSS-based designs to group HTML tags inside of <div> tags. You can

then position individual <div> tags to create columns and other complex page

layouts. It’s also common to use descendent selectors like this one to pinpoint

your formatting choices by affecting just the tags in certain areas of the page.

2. **Hit Enter, and then type these three CSS properties:**

color: #7F694F;

font-family: "Lobster", Arial, Helvetica, sans-serif;

font-size: 4em;

In order to use the font, add the font link within the head tag element in text.html

<link href='http://fonts.googleapis.com/css?family=Lobster' rel='stylesheet' type='text/css'>

You’ve just changed the color of the <h1> tag as well as the font.

Next you’ll add a text shadow to the headline.

3. **Complete this style by hitting Enter, adding the coded in bold below (don’t**

**forget the closing brace):**

.main h1 {

color: #7F694F;

font-family: "Lobster", Arial, Helvetica, sans-serif;

font-size: 4em;

**text-shadow: 4px 3px 6px rgba(0,0,0,.7);**

}

Here, you’ve added a text shadow which is offset 4 pixels to the

right, 3 pixels below, and is feathered out 6 pixels. In addition, you’re using the

RGBA color to set the shadow to black with 70 percent transparency.

4. **Save the file and preview it in a web browser.**

Next, spruce up the appearance of the other headings and paragraphs.

5. **Return to your text editor and the text.html file.** Click after the closing brace

of the .main h1 tag, hit Enter, and add the following two styles:

.main h2 {

font: bold 3.5em "Hoefler Text", Garamond, Times, serif;

color: #ECECEA;

border-bottom: 1px solid rgb(200,200,200);

margin-top: 25px;

text-shadow: 4px 3px 6px rgba(0,0,0,.5);

}

Here you have another descendent selector that only applies to <h2> tags inside

another tag with the class main (you’re probably getting the hang of these now).

The font property used here is shorthand that combines the more long-winded

font-weight, font-size, and font-family. In other

words, this one line makes the headline bold, 3.5ems tall, and specifies a font.

In addition, this style adds a decorative border below the headline and a bit of

space between the headline and the tag above it

Time to tackle more headlines.

6. **Add another style below the one you added in the last step:**

.main h3 {

font-family: "Droid Sans" sans-serif;

color: #D9853C;

font-size: 1.9em;

font-weight: bold;

text-transform: uppercase;

margin-top: 25px;

margin-bottom: 10px;

}

This style dishes out some of the usual formatting—color, size, boldness—and

also uses the text-transform property to make all of the text in the

<h3> headlines uppercase. Finally, it adds a bit of space above and below the

headlines by using the margin properties.

Next, you’ll improve the look of the paragraphs.

7. **Add one more style to the page:**

.main p {

font-size: 1.5em;

line-height: 150%;

margin-left: 150px;

margin-right: 50px;

margin-bottom: 10px;

}

This style introduces the line-height property, which sets the spacing between

lines. A percentage of 150 adds a little more space between lines in a paragraph

than you’d normally see in a web browser. This extra breathing room gives the

text a lighter, airier quality and makes the sentences a little easier to read ( if you speak Latin).

The style also increases the font size to 1.5em and

indents the paragraph from the left and right edges of the page

.

Time to try out a more advanced selector type.

8. **Add the following style to your style sheet:**

.main p:first-line {

font-weight: bold;

color: #ECECEA;

}

The :first-line pseudo-element affects just the first line of a paragraph.

In this case, just the first line of text for each of the paragraphs inside

the main div will be bold and gray.

9. **Save the page and open it in a web browser to preview your work.**

**Formatting Lists Lab**

This page has a single bulleted list. The plan is to move the list over to the right

edge of the page and have the text following it wrap around it. CSS makes this little

trick easy.

1. **Return to your text editor and the text.html file.** Add the following style at

the end of the page’s internal style sheet:

.main ul {

margin: 50px 0 25px 50px;

width: 170px;

float: right;

}

When formatting lists, you’ll usually create styles for two different elements:

the list itself (either the <ul> tag for bulleted lists or the <ol> tag for numbered

lists) and the individual list items (the <li> tag). This style controls the entire list.

There are a few things happening in this style. First, the margin property uses the

shorthand method. This one line sets all four margins around the list, replacing

the four individual margin properties (margin-top, margin-right, and so on).

The four values are ordered like this: top, right, bottom, left. So for this style,

50 pixels of space get added above the list, 0 space on the right, 25 pixels on

the bottom, and 50 pixels on the left.

The width property makes the entire list 170 pixels wide. If any particular list item has more text than will fit within that space, it wraps to another line.

The float property is the real magic—in this case, float: right means move the list over to the right edge of the page.

This property also causes the text following the list to wrap around the left side of the list.

You’ll control the look of the individual list items next.

2. **Add one more style to the internal style sheet in the text.html file:**

.main li {

color: #ECECEA;

font-size: 1.5em;

margin-bottom: 7px;

}

Nothing new here: just changing the color and size and adding space below

each list item. Time to check out your progress.

If you want to add space between list items, you need to add top or bottom margins to the <li>

tag. Adding margins to the <ul> or <ol> tags simply adds space around the entire list.

3. **Save the page and preview it in a web browser**

**Fine-Tuning with Classes Lab**

Sometimes you want even more control over how a style is applied. For example,

while you might want most paragraphs in one section of the page to look the

same, you might also want one or two paragraphs to have their own unique look.

In this tutorial, the paragraph of text near the top of the page “August 15 **BootCampers**”—contains some unique information—a publication date and author. You

want it to stand out from the other paragraphs, so you’ll add a class to the HTML

and create a class style.

1. **Locate the HTML for that paragraph—<p>August 15 <strong>BootCampers</**

**strong></p>—and add class="byline" to the opening <p> tag.** The HTML

should look like this:

<p class="byline">

August 15 <strong>Bootcampers</strong>

</p>

Now it’s a simple matter of creating a class style that overrides the generic

formatting of the other paragraphs on the page.\

ootcampers

2. **In the internal style sheet near the top of the page, add a style for that**

**paragraph:**

.main .byline {

font-size: 1.6em;

margin: 5px 0 25px 50px;

}

This style tweaks the size and placement of just that one paragraph. Note that

if you’d just named that style .byline—a basic class selector—it wouldn’t work.

Thanks to the rules of the cascade described in the last lesson, .byline is less

specific (less powerful) than the .main p style you created

so it wouldn’t be able to override the size and margins specified by .main p.

However, .main .byline is more specific.

That paragraph still needs some work. It would be great if the name stood out

more. The HTML in this case provides just the hook you need.

3. **Add another style to the style sheet:**

.main .byline strong {

color: rgb(32,126,191);

text-transform: uppercase;

margin-left: 11px;

}

If you look at the HTML in Step 1 above, you’ll see that the text—Bootcampers—is

inside a <strong> tag. The <strong> tag is used to emphasize text and mark

it as important. But that doesn’t mean you have to let it be bold, the way web

browsers normally display that tag. Instead, this descendent selector targets

the <strong> tag but only when it appears inside another tag with the class

.byline, and only if all of that is inside yet another tag with the class main—

whew, that’s pretty specific.

This style turns the text blue, makes it uppercase, and adds a bit of space on

the left side (nudging the name over just a bit from the “August 15” text).

z

**Adding the Finishing Touches Lab**

1. **Return to your text editor and the text.html file.**

First, you’ll add a background color and image to the page.

2. **Locate the html style near the top of the internal style sheet and add one**

**new property so that it looks like this (changes are in bold):**

html {

font-family: 'PTSans', Arial, sans-serif;

font-size: 62.5%;

margin: 0 auto;

background: rgba(86,141,138,1) url(images/retro\_rocket\_bg.jpg) no-repeat center right;

}

The background property is a powerful tool for any web designer. You’ve already

used it a couple of times in earlier tutorials; it lets you add color and insert and

control the placement of an image to the background of any tag.

this line changes

the background color of the page to chalky turquoise.

Next you’ll spruce up the main div.

3. **Add another style in between the html style and the .main h1 style:**

.main {

width: 50%;

margin: 0 auto;

padding: 0px 10px;

}

In other words, click after the closing } for the html style, hit Enter, and type

the code above. You don’t necessarily have to put the style in that spot for it

to work, but for organizational purposes, putting the style that controls the div

before the other styles that format tags inside that div makes sense.

The width property sets an overall width for this div (and the content inside

it), essentially turning this page into a 50%-wide document. The margin

property values—0 auto—put 0 pixels of space above and below the div and

set the left and right margins to auto, which centers the div in the middle of

the browser window. The padding property adds space inside the box, pushing

content inside the div away from the border line

4. **Add another style right after the .main h1 style:**

.main h1 strong {

font-size: 100px;

color: white;

line-height: 1em;

}

The HTML for the headline looks like this:

<h1><strong>Games &amp; Design</strong></h1>

The text size is pumped way up, its color changed, and line height is adjusted so

that it fits inside the top of the page. You’ll notice that the line height is set to

1em so in this case, the line height will translate to 100 pixels—that’s the

font size of this style.

5. **Add an image above the first h1 element**

Add the following content immediately within the first div below the <body> tag

<img src="images/rocket\_large.png" class="logo" alt="Gordons Alive!">

This adds a small rocket-ship logo image above the branding text.

**6.** **Save the file and preview it in a web browser.**

You’ve explored many of the text formatting properties offered by

CSS and turned basic HTML into an attractive design.