Fonts

**Why Fonts Matter**

Colors are an important aspect of the user experience (UX), but an overview of UX is not complete without a focus on fonts. It's very likely that a lot of the important information a user will see on a web page will be in the form of text. Styling text to make web page content accessible and visually engaging creates a great experience for users.

In this lesson, we'll learn all about fonts and how to style their appearance with CSS.

Let's begin!

**Font Family**

If you've ever used a formatted word processor, chances are that you probably also used a feature that allowed you change the "type of font" you were typing in. The phrase "type of font" refers to the technical term [typeface](https://en.wikipedia.org/wiki/Typeface), or *font family*.

To change the typeface of text on your web page, you can use the font-family property.

h1 {

font-family: Garamond;

}

In the example above, the font family for all main heading elements has been set to Garamond.

When setting typefaces on a web page, keep the following points in mind:

1. The font specified in a stylesheet must be installed on a user's computer in order for that font to display when a user visit the web page. We'll learn how to work around this issue in a later exercise.
2. You've probably noticed that we haven't been specifying a typeface in previous exercises of this course. How exactly does the browser know what typeface to use when displaying the web page? The default typeface for all HTML elements is Times New Roman. You may be familiar with this typeface if you have ever used a formatted word processor.
3. It's a good practice to limit the number of typefaces used on a web page to 2 or 3.
4. When the name of a typeface consists of more than one word, it must be enclosed in double quotes (otherwise it will not be recognized), like so:

h1 {

font-family: "Courier New";

}

# Serif vs Sans-Serif

The practice of [typography](https://en.wikipedia.org/wiki/Typography) has been around for centuries! Over time, typographers have refined their craft and have developed many different typefaces, which has allowed them, in some cases, to classify them as one of the following two types: serif fonts and sans-serif fonts.

1. Serif - the letters in these fonts have extra details on the ends of each letter. Examples include fonts like Times New Roman or Georgia, among others.
2. Sans-Serif - the letters in these fonts do not have extra details on the ends of each letter. Instead, letters have straight, flat edges. Some examples include Arial or Helvetica.

The majority of fonts that we'll study in this lesson will be either serif or sans-serif fonts.

# Fallback Fonts

Earlier, you learned that users must have the fonts specified in the stylesheet installed on their computer in order for their browser to display that font. What happens when a font is not installed on a user's computer?

Most computers have a small set of typefaces pre-installed. This small set includes serif fonts and sans-serif fonts, like Times New Roman and Arial, respectively.

When the stylesheet specifies a font not installed on a user's computer, the pre-installed fonts can be used as fallback fonts for users.

To use fallback fonts, the following syntax is required:

h1 {

font-family: Garamond, Times, serif;

}

The CSS rule above says: "Use the Garamond font for all <h1> elements on the web page. If that font is not available, use the Times font. If both of those fonts are not available, use any serif font pre-installed on the user's computer." The fonts specified after Garamond are the fallback fonts.

Fallback fonts help ensure a consistent experience for the diverse audience of users that visit a site.

# More Fonts

New fonts are constantly being developed. Because there are so many new fonts available, it would be unrealistic to expect users to have all of them installed on their computers.

Fortunately, you don't have to predict which fonts are installed on a user's computer. Many (but not all) of the new fonts that emerge on a daily basis are being centralized in directories made available for public use.

For example, Google offers [Google Fonts](https://fonts.google.com/), a directory of thousands of open-source fonts that are free to use by anyone.

To use these fonts, you can link to a specific Google Font in your HTML code and use it immediately in your stylesheet. Because the HTML file links directly to the Google Font, a user's browser can display the typeface you specify. This avoids having to determine whether or not that font is installed on a user's computer.

To use a Google Font, you can use a <link> element, just like you did for a CSS stylesheet:

<head>

<link href="https://fonts.googleapis.com/css?family=Raleway" type="text/css" rel="stylesheet" >

</head>

In the example above, the href attribute is set to the following URL, which was retrieved from Google Fonts:

https://fonts.googleapis.com/css?family=Raleway

The URL in the example above specifies the Raleway typeface from Google Fonts.

You can use the new font just as you would use any other font:

h1 {

font-family: Raleway, Georgia, serif;

}

You now have access to thousands of new, modern, free-to-use fonts!

# Font Size I

Changing the typeface isn't the only way to customize text. Often times, different sections of a web page and are highlighted by modifying the font size.

To change the size of text on your web page, you can use the font-size property.

p {

font-size: 18px;

}

In the example above, the font-size of all paragraphs was set to 18px. What exactly does px mean?

Measurements require units in order for them to be useful (for example kilograms for weight, or miles for distance). Font size also requires a unit of measurement.

In the next exercise, we'll explore in detail the units of measurement available for font size.

# Font Size II

There are three units of measurement for font size:

1. px - Represents the unit of pixels. The display of a computer monitor can be measured in pixels. A pixel is a small point on the display. How small? Most computer monitors have a resolution of 72 pixels per inch, so a pixel represents about 1/72nd of an inch. Pixels are sometimes also referred to as points. Pixels are used to set the exact size of an element, in this case, text.

p {

font-size: 18px;

}

2. ems - Pronounced just as it looks, "em." An em is equal to the width of the letter "m". Ems are a relative unit of measurement. They change the size of text relative to the parent element's size of text.

p {

font-size: 1.3em;

}

3. % - Percentages are also a relative unit of measurement. The default size of text in web browsers is 16 pixels, or 16px. When percentages are used, they set the size of text relative to this default size. For example, setting the font size to 200% would be equivalent to setting it to 32px.

p {

font-size: 150%;

}

# Line Height

Text on a web page must also be easy to read. When text is styled to appear larger, the vertical spacing between lines of text can decrease, creating text that is difficult to read, particularly in paragraphs.

To avoid this problem, you can modify the spacing between lines of text with the line-height property.

p {

line-height: 1.5em;

}

When the line height of an element is modified, you are manipulating the leading (pronounced "ledding") of the font. When the line height is increased, the spacing between lines of text increases, which can make text easier to read.

The line height can be modified using pixels or ems, but the unit of ems is preferred, since ems offer a spacing relative to the size of the text on the page.

# Line Height Anatomy

The diagram to the right helps illustrate exactly what the terms "leading" and "line height" mean.

When the line-height property of an element is modified, the leading is increased, resulting in an increase of the vertical spacing between lines of text.

# Word Spacing

You can also increase the spacing between words in a body of text, technically known as word spacing.

To do so, you can use the word-spacing property:

h1 {

word-spacing: 0.3em;

}

The default amount of space between words is usually 0.25em. In the example above, the word spacing is set to 0.3em, which represents an increase of only .05em in word spacing.

It's not common to increase the spacing between words, but it may help enhance the readability of bolded or enlarged text. Note, again, that the preferred unit is ems.

# Letter Spacing

You've learned how to increase the spacing between lines of text and words, but it's possible to get even more detailed: increasing the spacing between individual letters.

The technical term for adjusting the spacing between letters is called "kerning". Kerning can be adjusted with the letter-spacing property in CSS.

h1 {

letter-spacing: 0.3em;

}

Like word spacing, it's not common to increase the kerning in text, but sometimes it enhances the readability of uppercase text.

# Font Weight

You've probably noticed **bolded** text across many different web sites. It's common to bold important headings or keywords.

In CSS, the font-weight property turns bold on or off.

p {

font-weight: bold;

}

In the example above, all paragraphs on the web page would appear bolded.

The font-weight property has a second value: normal. Why does it exist?

If we wanted all text on a web page to appear bolded, we could select all text elements and change their font weight to bold. If a certain section of text was required to appear normal, however, we could set the font weight of that particular element to normal, essentially "shutting off" bold for that element.

In later units, you'll learn how to be more selective about what parts of your site you'd like to style.

# Font Style

You can also italicize words with the font-style property.

h3 {

font-style: italic;

}

The italic value causes text to appear in italics. The font-style property also has a normal value, for the same reasons discussed in the previous exercise.

# Text Transformation

Text can also be styled to appear in either all uppercase or lowercase with the text-transform property.

h1 {

text-transform: uppercase;

}

The code in the example above formats all <h1> elements to appear in uppercase, regardless of the case used for the heading within the HTML code. Alternatively, the lowercase value could be used to format text in all lowercase.

Since text can be directly typed in all uppercase or lowercase within an HTML file, what is the point of a CSS rule that allows you to format [letter case](https://en.wikipedia.org/wiki/Letter_case)?

Depending on the type of content a web page displays, it may make sense to always style a specific element in all uppercase or lowercase letters. For example, a website that reports breaking news may decide to format all <h1> heading elements such that they always appear in all uppercase, as in the example above. It would also avoid uppercase text in the HTML file, which could make code difficult to read.

**Text Alignment**

No matter how much styling is applied to text (typeface, size, weight, etc.), text always appears on the left side of the browser.

To move, or align, text, we can use the text-align property.

h1 {

text-align: right;

}

The text-align property can be set to one of the following three values:

1. left - aligns text to the left hand side of the browser.
2. center - centers text.
3. right - aligns text to the right hand side of the browser.

Later in the course, you'll learn exactly how the browser positions HTML elements by default, which will help you understand how the browser "aligns" text, since "align" is a relative term. For now, it's enough to know that text can be moved to the left, center, or right side of the web page.