



# CSS

## Definition

typography

The style, appearance, and arrangement of characters on a page

Typography can be used to add additional depth and organization to the textual content of our website.

# CSS

## Typography - Two Goals

The first goal of good typography is legibility. Legible text is easily scannable and individual letters are easy to distinguish from one another.

## CSS

### Legibility

Legibility is often impacted by the font choice. For example, the letters 'n' and 'u' are difficult to distinguish from one another using the font, 'Brush BQ'.



## CSS

### Legibility

Even with a font choice that is reasonably legible at larger sizes, smaller print may not be as legible.

For example, when using the font 'ITC Juice', the numbers '30' and '80' are reasonably legible at larger sizes. However, the '3' and '8'

\$30 \$30 \$30 \$30 \$30 \$30

\$80 \$80 \$80 \$80 \$80 \$80

## CSS

### Typography - Two Goals

The second goal of good typography is readability. Readability refers to the ease with which text is read. When a user moves from scanning to reading, the textual content of your site shouldn't require the user to strain their eyes. This is especially important for longer sections of text.

# CSS

## Definition

font

A complete set of type of one typeface, size, weight, and style

# CSS

## Definition

font family

A set of one or more fonts with the same typeface but with different sizes, weights, and styles



## CSS

### Font Family

Below is an example of a font family. 11 individual fonts are shown, all of which are part of the same font family.

ABCDEFGHIJKLMNOPQRSTUVWXYZ

ABCDEFGHIJKLMNOPQRSTUVWXYZ

ABCDEFGHIJKLMNOPQRSTUVWXYZ

ABCDEFGHIJKLMNOPQRSTUVWXYZ

ABCDEFGHIJKLMNOPQRSTUVWXYZ

ABCDEFGHIJKLMNOPQRSTUVWXYZ

ABCDEFGHIJKLMNOPQRSTUVWXYZ

ABCDEFGHIJKLMNOPQRSTUVWXYZ

*ABCDEFGHIJKLMNOPQRSTUVWXYZ*

**ABCDEFGHIJKLMNOPQRSTUVWXYZ**

***ABCDEFGHIJKLMNOPQRSTUVWXYZ***

## CSS

### Font Types - Serif

Font family can be classified into a number of categories. One of the oldest classifications is 'serif' fonts. Serif fonts have small finishing strokes (known as 'serifs') at the end of each character. Serif fonts may also be referred to as 'Roman' fonts or 'Gothic' fonts.

Serif fonts provide a horizontal line of reference for the reader. At larger sizes, serif fonts provide a natural horizontal line of reference which can help with increased legibility. That said, it can be argued that serifs add too much clutter at smaller sizes, reducing legibility and readability.

The word "Typography" is displayed in a large, bold, black serif font. Each letter has small, pink, L-shaped brackets and dots at the ends of its horizontal strokes, highlighting the serifs. The 'T' has a top-left bracket, the 'y' has a bottom-left bracket, the 'o' has a top-right dot, the 'g' has a bottom-right dot, the 'r' has a top-right bracket, the 'a' has a top-right dot, the 'p' has a top-right bracket, and the 'h' has a top-right bracket and a bottom-right dot.

# CSS

## Font Types - Serif - Traditional

Serif fonts can be broken down into sub-categories:

- Traditional
- Transitional
- Modern
- Slab

Traditional serif fonts, also known as 'old style' or 'humanist' fonts, date back to the mid-1400s, just after the invention of the printing press.

Traditional serif fonts have the following characteristics:

- More rounded strokes and serifs
- Smooth transitions between thin and thick strokes
- Provide a more historic feel

**CSS**

**Font Types - Serif - Traditional**



# CSS

## Font Types - Serif - Transitional

Transitional serif fonts emerged between traditional and modern serif fonts, hence the name 'transitional'.

Transitional serif fonts have the following characteristics:

- Rounded brackets (curved angle between letters and serifs)
- Straight-edged serifs
- Provides a more modern feel than traditional
- Very versatile

**CSS**

**Font Types - Serif - Transitional**



# CSS

## Font Types - Serif - Modern

Modern serif fonts emerged in the late 18th century and were less ornamental than its predecessors.

Modern serif fonts have the following characteristics:

- Large contrast between thin and thick strokes
- Provides a feeling of elegance and sophistication

Modern serif fonts are much less readable at smaller sizes but work well for page headings and other large textual content.

CSS

Font Types - Serif - Modern

T T T



# CSS

## Font Types - Serif - Slab

Slab serif fonts differ from the other serif fonts in that they don't have as much contrast between thin and thick lines.

Slab serif fonts have the following characteristics:

- Created to catch attention
- Provide a feeling of strength and power
- Can be overwhelming when misused

CSS

Font Types - Serif - Slab



## CSS

### Font Types - Sans-Serif

Sans-serif fonts (derived from the French word 'sans', meaning 'without') are fonts that do not have serifs on the characters and tend to have less contrast between thick and thin lines (if the characters have any line variation at all).

Sans-serif fonts weren't used heavily until the late 1800s. Also known as 'grotesque' fonts, many typographers saw the new font style as controversial, partially due to their departure from tradition.

AaBbCcDd

Sans-serif fonts provide a clean and contemporary feel. They can provide contrast between page elements when used along side serif fonts.

There is much debate about whether sans-serif fonts provide more readable text than serif fonts. While it can be argued that serifs provide a horizontal reference point for reading, at smaller sizes, sans-serif fonts typically provide a better reading experience for users.

## CSS

### Font Types - Script/Cursive

Cursive fonts, also known as script fonts, are designed to look like hand-writing. Just as with hand-writing between individuals, these fonts range from elegant calligraphy to 'less-than-elegant'.

*AaBbCcDd*

AaBbCcDd

Cursive fonts are best used for short bursts of text, such as logos or article headings.

## CSS

### Font Types - Ornamental/Display

Ornamental fonts, also known as 'display fonts', are designed to be very ornamental. While there's lots of variations between these fonts, most of them provide poor readability and often have poorer legibility than other fonts.

Ornamental fonts, like cursive fonts, are best used for short bursts of text, such as logos or article headings.

AaBbCcDd

## CSS

### Font Family

To change the font family using CSS, we can use the 'font-family' property. Values for font-family can be specified with or without quotes. That said, if the font family you want to use has a space in the name, quotes are required.

#### CSS

```
p
{
  font-family: fontName1;
}
```

#### CSS

```
p
{
  font-family: 'Font Name 2';
}
```

## CSS

### Limitation of Web Typography

One of the limitations of typography on the web is that fonts displayed on the user's computer are limited to those fonts that are installed on said computer. For example:

- E.g. If you use a really funky font...



THIS FONT IS  
CRAZY AWESOME!

... your users will likely see this.



THIS FONT IS  
CRAZY AWESOME!

Because of this, we want to limit ourselves to using font families that are commonly found on most computers.

# CSS

## Recommended Fonts

Below are listed fonts that are found on most computers, regardless of operating system.

- Times New Roman
- Georgia
- Verdana
- Arial
- Courier
- Trebuchet MS
- Comic Sans MS
- Impact



## CSS

### Reduced Font Selection

- Fonts only display for your users if they have that font installed on their computer
- Solution? Use multiple font options

Limiting ourselves to only a few fonts is frustrating as a designer. Fortunately for us, we are able to specify multiple values for font-family. The first value specified will be displayed if the user has that font installed. If they don't, then the browser will try to display the second font listed. Best practice would be to list your fonts in the order desired, while keeping in mind that the most popular font listed is the one most likely to be displayed.

While using the fonts listed on the previous slide will work for most users, it may not work on all computers. To address this possibility, it is considered best practice to include a generic font family as the last value for your font-family property.

## CSS

```
p
{
  font-family: fontName1, fontName2, ..., genericFontFamily;
}
```

## CSS

```
p
{
  font-family: 'Palatino Linotype', Georgia, serif;
}
```

Generic font family values are as follows:

- serif
- sans-serif
- cursive\*
- fantasy\*
- monospace

*\* It is worth noting that some generic font family values may have undesirable results depending on the operating system and browser.*

# CSS

## Font Family

### CSS

```
p
{
  /* This will try to make the font Palatino Linotype.
  * If Palatino Linotype isn't installed on this computer,
  * then it will try Georgia. If that's
  * not available then it will use the default
  * serif font.
  */
  font-family: 'Palatino Linotype', Georgia, serif;
}
```

# CSS

## Font Family

### CSS

```
p
{
  /* You can use non-standard fonts.
  * Just be sure you have a good
  * backup and always specify a
  * generic font family.
  */

  font-family: Helvetica, Arial, sans-serif;
}
```

# CSS

## font-family Property

### CSS

```
div { font-family: "Times New Roman", serif; }
```

```
div { font-family: sans-serif; }
```

```
/*  
 * Can take the name of any font family  
 * or a generic font family from the following  
 * list:  
 * 'serif', 'sans-serif', 'cursive', 'fantasy',  
 * or 'monospace'.  
 *  
 * Default Value: N/A  
 * Inherited: Yes  
 *  
 */
```

## CSS

### text-align Property

#### CSS

```
div { text-align: left; }
```

```
div { text-align: center; }
```

```
/*  
 * Can any one of the values 'left', 'right',  
 * 'center', or 'justified'.  
 *  
 * Default Value: left  
 * Inherited: Yes  
 *  
 */
```

## CSS

### Text Alignment

Using 'justify' for the text-align property has the effect of making the text stretch from one end of the element to the other removing the 'ragged edge' on the right side of the text that is found when left-aligned.

This is an effect that is often used in print layouts but is less effective when used on the web. This is due to the fact that when used on smaller elements, browsers may display lots of white space in between words.

## CSS

```
p
{
  text-align: justify;
}
```

## Browser

Lorem ipsum dolor  
sit amet,  
consectetur  
adipiscing elit. Ut  
faucibus est vel  
lacus iaculis  
placerat. Aliquam  
erat volutpat. Mauris  
ipsum velit,  
consequat vel  
pretium quis,  
vestibulum nec nisi.  
Nunc eleifend metus  
sed neque feugiat  
nec cursus turpis  
porttitor. Vivamus  
consectetur lobortis  
nulla, non congue  
nibh ullamcorper  
sed.

## **CSS**

### **Hyphenation**

In order for justified text to be displayed properly, proper hyphenation must be provided. Unfortunately, another limitation of typography on the web is that there is a lack of hyphenation provided by the browser.

## CSS

### Hyphenation - An option

If full justification is a non-negotiable requirement for your website, there is an option, though it requires a fair amount of work and creates a potential maintenance nightmare.

There is an HTML entity that provides us with a 'soft' hyphen. This entity tells the browser that a hyphen can be inserted into a word at a given point. If needed, the hyphen will be added automatically. If it is not needed, then nothing will be displayed.

#### HTML

```
&shy;
```

#### HTML

```
<p>This is par&shy;tic&shy;u&shy;lar&shy;ly long word.</p>
```

#### Browser

This is particu-  
larly long word.



# CSS

## Font Size

The font-size property allows us to change the size of our text.

### CSS

```
p  
{  
  font-size:200%; /* Twice the font-size of the containing element */  
}
```

There are a few options for how we can adjust the size of our text...

# CSS

## Font Size - Units of Measurement

The first option is to use a specific pixel value for our font size. While this provides us with a great deal of control, it has a number of drawbacks.

Using pixels limits us in that it doesn't allow us to create any relationships between our elements. Additionally, pixels creates accessibility issues on older browsers in that it doesn't allow the user to increase/decrease the size of the text.

# CSS

## Font Size - Units of Measurement

Instead of using pixels, we can use percentages. When using a percentage value for font-size, the size of the text is based on the font-size of the containing element.

This approach is well suited for accessibility and also provides us with better cross-device results. It also creates a relationship between our elements, allowing us to easily make changes to an entire web page just by changing the font size of the topmost element (i.e. the <body>)

## CSS

### Font Size - Units of Measurement

Another alternative is to use 'ems'. Technically, an 'em' is equal to the width of the letter 'm' in the current font. Another way to think about it is that it's the current font size. If you want to double the font size of an element, you would specify the value 2em.

This approach is well suited for accessibility and also provides us with a good option for creating well-proportioned spacing around our elements. 'ems' can be used to specify padding and margins, allowing us to create white space that is the equivalent of one line of text (1em).

It is worth noting that while you can use 'em' to specify border width, you may not get consistent results from one browser to the next.

# CSS

## font-size Property

### CSS

```
div { font-size: 19px; /* Not best practice */ }
```

```
div { font-size: 120%; }
```

```
div { font-size: 1.2em; }
```

```
/*  
 * Font size can be specified in pixels  
 * (not best practice), as a percentage,  
 * in ems, or using one of the following  
 * keywords:  
 * 'xx-small', 'x-small', 'small', 'medium',  
 * 'large', 'x-large', 'xx-large', 'smaller',  
 * or 'larger'.  
 *  
 * Default Value: 'medium'  
 * Inherited: Yes  
 *  
 */
```

## CSS

### Font Size - A tip...

- Most browsers set the default width to approx. 16px.

## CSS

```
body
{
  font-size:62.5%
  /* Now the font size for page is approx. 10px. */
}

p
{
  /* Multiplying by 10 is much easier than by 16. */
  /* Approx. 15px */
  font-size:150%;
}
```

## **CSS**

### **Font Spacing - Kerning**

Kerning is the space adjustment between specific letter pairs for a given font. For example, the space between the letters an 'A' and a 'W' should ideally be less than that of between the letters 'B' and 'E' in many fonts. Proper kerning helps with legibility and readability. Unfortunately, one of the limitations of typography on the web is that web browsers do not offer kerning.

## CSS

### **Kerning Tracking**

Instead of kerning, web browsers provide us with the related concept of 'tracking'.

- Kerning adjusts space between specific letter pairs
- Tracking adjusts space between **all** letters

WAVE

WAVE

WAVE



## CSS

### Tracking

Using the 'letter-spacing' property, we can adjust the spacing **between all letters** using positive or negative values.

## CSS

```
p
{
  /* Applying tracking to paragraphs. */
  letter-spacing: -1px;
}
```

Thisissometcompressedtext

# CSS

## letter-spacing Property

### CSS

```
div { letter-spacing: normal; }
```

```
div { letter-spacing: 0.1em; /* Expanded */}
```

```
div { letter-spacing: -0.1em; /* Compressed */}
```

```
/*  
 * Letter-spacing allows us to specify  
 * the spacing between ALL characters.  
 * Can either be a value or the keyword  
 * 'normal'.  
 *  
 * Default Value: 'normal'  
 * Inherited: Yes  
 *  
 */
```

## CSS

### Tracking

Because 'letter-spacing' doesn't always provide us with an ideal solution for spacing, we also have the option to use the 'word-spacing' property. Rather than adjusting the space between every letter, this adjusts the space between every word.

## CSS

```
p
{
  /* An alternative to tracking */
  word-spacing: .3em;
}
```

### Browser

This is some less expanded text.

# CSS

## word-spacing Property

### CSS

```
div { word-spacing: normal; }
```

```
div { word-spacing: 0.1em; /* Expanded */}
```

```
div { word-spacing: -0.1em; /* Compressed */}
```

```
/*  
 * Word-spacing allows us to specify  
 * the spacing between words.  
 * Can either be a value or the keyword  
 * 'normal'.  
 *  
 * Default Value: 'normal'  
 * Inherited: Yes  
 *  
 */
```

## CSS

### Readability - Line Height

Another important concept in typography is line height, or the space between lines. Browsers set this to 100% by default. However, usability studies have shown that the ideal line height rests closer to 142%.

## CSS

```
p  
{  
  line-height:142%;  
}
```

# CSS

## line-height Property

### CSS

```
div { line-height: normal; }
```

```
div { line-height: 1.42em; }
```

```
div { line-height:142%; }
```

```
/*  
 * Line-height allows us to specify  
 * the amount of spacing between lines  
 * of text. Can either be a value or the keyword  
 * 'normal'.  
 *  
 * Default Value: 'normal'  
 * Inherited: Yes  
 *  
 */
```

# CSS

## font-style Property

The font-style property allows us to specify whether the text of an element should be italicized.

### CSS

```
div { font-style: normal; }
```

```
div { font-style: italic; }
```

```
div { font-style: oblique; /* Often the same as italic */}
```

```
/*  
 *  
 * Default Value: 'normal'  
 * Inherited: Yes  
 *  
 */
```

## CSS

### font-weight Property

The font-weight property allows us to specify whether the text of an element should be bold.

## CSS

```
div { font-weight: normal; }
```

```
div { font-weight: bold; }
```

```
/*  
 * Font weight allows us to specify  
 * how 'bold' the text is.  
 * Technically, there are a few different  
 * options for what you put here but  
 * most of the time, it's either  
 * 'normal' or 'bold'.  
 *  
 * Default Value: 'normal'  
 * Inherited: Yes  
 *  
 */
```



## CSS

### style/weight vs. em/strong

- Why do we need font-style and font-weight when we have <em> and <strong>?
- You may want to use different styles to emphasize your text

## CSS

style/weight vs. em/strong

### CSS

```
strong
{
  font-weight:normal;
  font-size:200%;
  color:#ff0;
}
```

This is some **important information!**

## CSS

### Font Variant

#### CSS

```
p
{
  font-variant: normal;
}

a
{
  /* Allows for 'copperplating' */
  font-variant: small-caps;
}
```

This is a [COPPERPLATE LINK](#).

# CSS

## font-variant Property

### CSS

```
div { font-variant: normal; }
```

```
div { font-variant: small-caps; }
```

```
/*  
 * Font variant allows us  
 * to implement copperplating.  
 * This is either 'normal' or  
 * 'small-caps'.  
 *  
 * Default Value: 'normal'  
 * Inherited: Yes  
 *  
 */
```

## CSS

### Font Variant

#### CSS

```
p
{
  font-variant: normal;
}

a
{
  /* Allows for 'copperplating' */
  font-variant: small-caps;
}
```

This is a [COPPERPLATE LINK](#).

## CSS

### text-decoration Property

The text-decoration property allows us to specify whether the text should have a line under, over, or through it.

## CSS

```
div { text-decoration: none; }
```

```
div { text-decoration: underline; }
```

```
/*  
 * Text-decoration allows us  
 * to 'decorate' our text. Values  
 * are either 'underline', 'overline'  
 * 'line-through', or 'none'.  
 *  
 * Default Value: 'none'  
 * Inherited: No  
 *  
 */
```

## CSS

### text-indent Property

The text-indent property allows us to specify whether the first line of an element should be indented.

## CSS

```
div { text-indent: none; }
```

```
div { text-indent: 3em; }
```

```
/*  
 * Text-indent allows us to indent  
 * the first line of a block of text.  
 * You can specify values using pixels,  
 * percentages, or ems.  
 *  
 * Default Value: 0  
 * Inherited: Yes  
 *  
 */
```

## CSS

### text-transform Property

#### CSS

```
div { text-transform: none; }
```

```
div { text-transform: uppercase; }
```

```
/*  
 * Text-transform allows us to control  
 * the capitalization of the text.  
 * You can specify one of the following:  
 * 'uppercase', 'lowercase', 'capitalize',  
 * or 'none'.  
 *  
 * Default Value: none  
 * Inherited: Yes  
 *  
 */
```

It is important to note that changing the case of letters can significantly impact the legibility and readability of text. The text-transform property can best be used with small bursts of text. With longer sections of text, it is not recommended.



## CSS

### Sizing

Now that we have explored the properties available to us, it is worth discussing how they can be used to help us achieve our goals of legibility and readability.

## CSS

```
font-size: ...;
```

The size of our type is useful for drawing attention to specific sections of text, improving how easily something is scanned, and also for drawing attention away from less-important text.

- Larger size often improves legibility
- Larger size draws attention
- Smaller size de-emphasizes content
- Useful for headings / footnotes

# CSS

## Sizing

De-emphasized Text

### Emphasized Text

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut in nunc non elit egestas vehicula at sit amet lectus. Nulla nisi dui, venenatis in blandit in, rutrum ac odio. Sed sagittis pulvinar nisi eu lacinia. Nunc ipsum mi, faucibus ac vestibulum sed, rhoncus ut diam. Vivamus nec tincidunt sapien. Pellentesque auctor venenatis velit ac ultrices. Morbi ut mauris eget libero porttitor lacinia. Sed risus urna, dictum sit amet suscipit ut, elementum nec leo.

\* Footnote

# CSS

## Font Family

### CSS

```
font-family:...
```

By using our understanding of various font types, we can draw attention to certain text or create relationships (or distinction) between elements on our page.

- Contrast serif/sans-serif fonts to create distinction
- For example, you can use a serif font for headings and sans-serif for paragraphs
- Script/decorative fonts Can be done with other fonts but remember the fonts on the web

## CSS

# Emphasized Text

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut in nunc non elit egestas vehicula at sit amet lectus. Nulla nisi dui, venenatis in blandit in, rutrum ac odio. Sed sagittis pulvinar nisi eu lacinia. Nunc ipsum mi, faucibus ac vestibulum sed, rhoncus ut diam. Vivamus nec tincidunt sapien. Pellentesque auctor venenatis velit ac ultrices. Morbi ut mauris eget libero porttitor lacinia. Sed risus urna, dictum sit amet suscipit ut, elementum nec leo.

# CSS

## Color

### CSS

```
color: ...;  
background-color: ...;
```

As discussed in our lecture on color theory, color can be used to draw attention to certain text, de-emphasize less important text, communicate or reinforce meaning (e.g., using red to indicate danger), or to indicate that something isn't available (using light gray for out of stock items).

# CSS

De-emphasized Text

## Emphasized Text

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut in nunc non elit egestas vehicula at sit amet lectus. Nulla nisi dui, venenatis in blandit in, rutrum ac odio. Sed sagittis pulvinar nisi eu lacinia. Nunc ipsum mi, faucibus ac vestibulum sed, rhoncus ut diam. Vivamus nec tincidunt sapien. Pellentesque auctor venenatis velit ac ultrices. Morbi ut mauris eget libero porttitor lacinia. Sed risus urna, dictum sit amet suscipit ut, elementum nec leo.

\* Footnote

## CSS

### Capitalization

#### CSS

```
text-transform:...
```

As previously mentioned, capitalization, when used appropriately, can be used to emphasize certain elements. That said, this approach should be limited to headings and/or small bursts of text.

# CSS

de-emphasized text

## EMPHASIZED TEXT

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut in nunc non elit egestas vehicula at sit amet lectus. Nulla nisi dui, venenatis in blandit in, rutrum ac odio. Sed sagittis pulvinar nisi eu lacinia. Nunc ipsum mi, faucibus ac vestibulum sed, rhoncus ut diam. Vivamus nec tincidunt sapien. Pellentesque auctor venenatis velit ac ultrices. Morbi ut mauris eget libero porttitor lacinia. Sed risus urna, dictum sit amet suscipit ut, elementum nec leo.

\* Footnote



## CSS

### Style/Decoration/Weight

#### CSS

```
text-decoration:...;  
font-style:...;  
font-weight:...;
```

Adjusting the style of text can also be used to emphasize text. It is important to limit using italics or bold text to just a few words at a time. Otherwise, the intended effect can be lost and the user left more distracted by the emphasized text than intended.

It is worth noting that users will automatically associate short sections of underlined text with hyperlinks. Use underlining cautiously!

# CSS

de-emphasized text

## EMPHASIZED TEXT

**Lorem ipsum dolor sit amet, consectetur adipiscing elit.** Ut in nunc non elit egestas vehicula at sit amet lectus. Nulla nisi dui, venenatis in blandit in, rutrum ac odio. Sed sagittis pulvinar nisi eu lacinia. Nunc ipsum mi, faucibus ac vestibulum sed, rhoncus ut diam. Vivamus nec tincidunt sapien. Pellentesque auctor venenatis velit ac ultrices. Morbi ut mauris eget libero porttitor lacinia. Sed risus urna, dictum sit amet suscipit ut, elementum nec leo.

\* Footnote

# CSS

## White Space

### CSS

```
padding:...;  
margin:...;  
line-height:...;  
word-spacing:...;  
letter-spacing:...;  
text-indent:...;
```

Proper use of white-space is one of the best tools in a designer's arsenal. Users look for spacing to help tell them where to start, where to pause, where to stop, and where to go next.

By applying appropriate spacing between elements, lines, words, and characters, we can naturally guide the users eye through our website, pointing them to the important elements on our page without requiring much effort on their part.

# CSS

## Heading 1

Posted on February 20, 2012

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut in nunc non elit egestas vehicula at sit amet lectus. Nulla nisi dui, venenatis in blandit in, rutrum ac odio. Sed sagittis pulvinar nisi eu lacinia. Nunc ipsum mi, faucibus ac vestibulum sed, rhoncus ut diam. Vivamus nec tincidunt sapien. Pellentesque auctor venenatis velit ac ultrices. Morbi ut mauris eget libero porttitor lacinia. Sed risus urna, dictum sit amet suscipit ut, elementum nec leo.

Vivamus vel lectus eu justo consequat congue eu et enim. Sed vitae mi lorem. Donec elementum, magna vitae eleifend sodales, purus mi blandit sapien, et congue neque tellus vitae arcu. Fusce faucibus est at metus ornare at vulputate sapien tristique. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vivamus vitae facilisis erat. Fusce eget nibh libero. Ut rutrum accumsan leo et consectetur. Phasellus venenatis ligula ut nisi laoreet ac commodo ipsum cursus.

## Heading 2

Posted on February 20, 2012

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut in nunc non elit egestas vehicula at sit amet lectus. Nulla nisi dui, venenatis in blandit in, rutrum ac odio. Sed sagittis pulvinar nisi eu lacinia. Nunc ipsum mi, faucibus ac vestibulum sed, rhoncus ut diam. Vivamus nec tincidunt sapien. Pellentesque auctor venenatis velit ac ultrices. Morbi ut mauris eget libero porttitor lacinia. Sed risus urna, dictum sit amet suscipit ut, elementum nec leo.

Vivamus vel lectus eu justo consequat congue eu et enim. Sed vitae mi lorem. Donec elementum, magna vitae eleifend sodales, purus mi blandit sapien, et congue neque tellus vitae arcu. Fusce faucibus est at metus ornare at vulputate sapien tristique. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vivamus vitae facilisis erat. Fusce eget nibh libero. Ut rutrum accumsan leo et consectetur. Phasellus venenatis ligula ut nisi laoreet ac commodo ipsum cursus.

## Heading 3

Posted on February 20, 2012

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut in nunc non elit egestas vehicula at sit amet lectus. Nulla nisi dui, venenatis in blandit in, rutrum ac odio. Sed sagittis pulvinar nisi eu lacinia. Nunc ipsum mi, faucibus ac vestibulum sed, rhoncus ut diam. Vivamus nec tincidunt sapien. Pellentesque auctor venenatis velit ac ultrices. Morbi ut mauris eget libero porttitor lacinia. Sed risus urna, dictum sit amet suscipit ut, elementum nec leo.

Vivamus vel lectus eu justo consequat congue eu et enim. Sed vitae mi lorem. Donec elementum, magna vitae eleifend sodales, purus mi blandit sapien, et congue neque tellus vitae arcu. Fusce faucibus est at metus ornare at vulputate sapien tristique. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vivamus vitae facilisis erat. Fusce eget nibh libero. Ut rutrum accumsan leo et consectetur. Phasellus venenatis ligula ut nisi laoreet ac commodo ipsum cursus.

# CSS

## Pseudo Elements

Pseudo elements are elements that aren't explicitly defined in the HTML but are identifiable by the browser. We can target these pseudo elements by using any valid CSS selector, followed by a colon, followed by a valid pseudo element keyword.

It is worth noting that not all browsers (particularly older browsers) support pseudo elements.

## CSS

```
CSSselector:pseudoElement  
{  
  /* CSS properties here */  
}
```

## CSS

### Pseudo Elements

#### :first-letter

Allows you to style the first letter of an element.

## CSS

```
p:first-letter
{
  font-weight:bold;
  font-size:300%;
}
```

## Browser

**L**orem ipsum dolor sit amet, consectetur adipiscing elit. Nunc blandit tortor at lectus ornare consequat. Morbi ornare ipsum sed nisl pretium at commodo ante eleifend. Integer nibh nisi, ornare sed sodales id, congue a nulla. Mauris blandit vulputate blandit. Nullam ut diam dui. Curabitur non dictum dolor. Donec id urna nec lectus bibendum sagittis eget a justo. Donec vitae neque nisi, vel vestibulum odio. Aenean a est lectus, ac tempus neque.

## CSS

### Pseudo Elements

:first-letter

## CSS

```
p:first-letter
{
  font-size:300%;
  line-height:1em;
  font-family:'Times New Roman', serif;
  border:1px solid #000;
  padding:5px 10px;
  margin:0 5px 0;
  float:left;
}
```

## Browser

**L**

orem ipsum dolor sit amet, consectetur adipiscing elit. Nunc blandit tortor at lectus ornare consequat. Morbi ornare ipsum sed nisl pretium at commodo ante eleifend. Integer nibh nisi, ornare sed sodales id, congue a nulla. Mauris blandit vulputate blandit. Nullam ut diam dui. Curabitur non dictum dolor. Donec id urna nec lectus bibendum sagittis eget a justo. Donec vitae neque nisi, vel vestibulum odio. Aenean a est lectus, ac tempus neque. Lectus ornare consequat. Morbi ornare ipsum sed nisl pretium at commodo ante eleifend. Integer nibh nisi, ornare sed sodales id, congue a nulla. Mauris blandit vulputate blandit. Nullam ut diam dui. Curabitur non dictum dolor. Donec id urna nec lectus bibendum sagittis eget a justo. Donec vitae neque nisi, vel vestibulum odio. Aenean a est lectus, ac tempus neque.

## CSS

### Pseudo Elements

#### :first-line

Allows you to style the first line of an element.

## CSS

```
p.article:first-line
{
  font-weight:bold;
  color:#a00;
}
```

## Browser

**Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc blandit tortor at lectus ornare consequat. Morbi ornare ipsum sed nisl pretium at commodo ante eleifend. Integer nibh nisi, ornare sed sodales id, congue a nulla. Mauris blandit vulputate blandit. Nullam ut diam dui. Curabitur non dictum dolor. Donec id urna nec lectus bibendum sagittis eget a justo. Donec vitae neque nisi, vel vestibulum odio. Aenean a est lectus, ac tempus neque.**



# CSS

## Pseudo Classes

Pseudo classes are based on the users interaction with elements on the web page. The pseudo class describes a behavior state for a given element.

We can target these behavior states by using any valid CSS selector, followed by a colon, followed by a valid pseudo class keyword.

## CSS

```
CSSSelector:pseudoClass  
{  
  /* CSS properties here */  
}
```

# CSS

## Pseudo Classes

Most of the time, pseudo classes are used with hyperlinks. For example, we can change how hyperlinks look when 'hovered over' (i.e., when the mouse cursor is positioned over the element).

### HTML

```
<a href="index.html">My Link</a>
```

### CSS

```
a  
{  
  background-color:black;  
  color:#fff;  
}  
  
a:hover  
{  
  background-color:#ff0;  
  color:#000;  
}
```



My Link

# CSS

## Pseudo Classes

Valid CSS 2.1 pseudo classes are as follows:

- :link
- :visited
- :focus
- :hover
- :active

These should be added to your CSS in this order!

# CSS

## Pseudo Classes

### :link

References links in their default state

## CSS

```
a:link
{
  color:red;
}
/** These are the same **/
a
{
  color:red;
}
```

# CSS

## Pseudo Classes

### :visited

References links that have already been visited by the user

## CSS

```
a
{
  color:#00f;
}

a:visited
{
  background-color:#000;
  color:#fff;
}
```

**This link** has been visited but [this link](#) has not.

## CSS

### Pseudo Classes

#### :focus

References elements that currently have the keyboard input focus (is ready to receive input via the keyboard).

## CSS

```
a
{
  background-color:#000;
  color:#fff;
}

a:focus
{
  background-color:#ff0;
  color:#000;
}
```

## Visit Google!

If the user were to press the tab key, they could navigate around your website without using the mouse. Using the :focus pseudo class, we can indicate how elements should behave when they have the 'focus' of the keyboard. In the above example, the background color of the link will turn from black to yellow, while the text will change from white to black.

## CSS

### Pseudo Classes

#### :hover

References elements that currently have the mouse pointer over them

## CSS

```
a
{
  background-color:#000;
  color:#fff;
}

a:hover
{
  background-color:#ff0;
  color:#000;
}
```

[This is my link!](#)

# CSS

## Pseudo Classes

### :hover

:hover and :focus should be used together to provide the most consistent (and accessible) user experience.



## CSS

### Pseudo Classes

#### :active

References a link that is currently being clicked.

## CSS

```
a
{
  background-color:#fff;
  color:#000;
}

a:active
{
  color:#a00;
}
```

### This is my link!

'active' describes the state between when a user clicks on a link and when they let up the mouse button. While this typically is only a split second, it is a noticeable amount of time from a user's perspective. Using the :active class, we can provide a visual indication that the user's mouse click was, in fact, effective.

## CSS

### Pseudo Classes

#### :active

References a link that is currently being clicked.

## CSS

```
a:focus
{
  background-color:#000;
  color:#fff;
}

a:active
{
  color:#000;
  /* Active links also have focus!!! */
}
```

This is my link!