

CSS

CSS for Styling

- introduce the basics of CSS
 - different visual styles to HTML elements,
 - change their preset default appearance.

Lesson outline

- setting CSS properties
 - color, font, text, background, borders
- how rule sets are combined through cascading and inheritance
- selecting elements with id and class attributes

Content vs. Presentation

- HTML is for content, the information on the page
- CSS is for presentation, how to display the page
- Keeping content separate from presentation is a very important web design principle
- If the HTML contains no styles, its entire appearance can be changed by swapping **.css** files
- <http://www.csszengarden.com/>

Basic CSS rule syntax

- A **CSS** file consists of one or more rules
- A rule's selector specifies HTML element(s) and applies style properties
 - The * selector, selects all elements
 - To add a comment we use: /* */

```
selector {  
  property: value;  
  property: value;  
  ...  
}  
  
p {  
  font-family: sans-serif;  
  color: red;  
}
```



CSS properties for colors



```
p {  
  color: white;  
  background-color: blue;  
}
```



This paragraph uses the style above.

Linking external stylesheet (preferred way)

- Put all CSS rules in separate .css file
- Link in head section of html page using <link> tag.



```
<head>
  <link href="style.css" type="text/css" rel="stylesheet" />
</head>
```

```
<head>
  <style>    <!-- less preferred -->
    p {
      color: white;
      background-color: blue;
    }
  </style>
</head>
```

Exercise

- Apply the previous p css rule to the html demo from yesterday.
 - First use the <style> element on the page
 - Second change to use the preferred <link> to a separate file



Specifying colors

- **Color names:** aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white (white), yellow
- **RGB codes:** red, green, and blue values from 0 to 255
- **HEX codes:** RGB values in base-16 from 00 (none) to FF (full)



```
h1 { color: red;    }  
h2 { color: rgb(128, 0, 196); }  
h3 { color: #FF8800; }
```

CSS properties for font



```
p {  
    /* can specify multiple fonts from highest to lowest priority */  
    font-family: Garamond, "Times New Roman", serif;  
    font-size: 14px;  
    font-weight: bold;  
    font-style: italic;  
}
```

```
p {  
    /* The font property is a shorthand for font-style, font-variant,  
    font-weight, font-size/line-height, font-family*/  
    font: 14px Arial, sans-serif;  
}
```

The font-size and font-family values are required. If one of the other values is missing, their default value are used.



Size Units

- **Units:** pixels (**px**), point (**pt**), m-size (**em**), viewport width/height(vw, vh)
 - **px** specifies number of pixels on the screen
 - **em** relative to the font-size of the element (2em means 2 times the size of the current font)
 - **pt** specifies number of points, where a point is 1/72 of an inch on screen
- vw and vh represent 1% of viewport width and height
 - https://www.w3schools.com/cssref/tryit.asp?filename=trycss_unit_vw

CSS properties for text

Property	Description	Values
text-align	alignment of text within its element	left, center, right, justify
text-decoration	decorations such as underlining	underline, overline, line-through, blink, none
text-indent	Indent first line	a size(px, pt, %, em)
line-height	vertical size of each line	a size(px, pt, %, em)
letter-spacing	Horizontal gap between letters	a size(px, pt, %, em)



CSS properties for text

```
p {  
    /* Alignment of text within its element,  
       can be left, right, center, or justify */  
    text-align: center;  
    line-height: 30px;  
    word-spacing: 30px;  
    /* Indent the first line */  
    text-indent: 50px;  
}
```

CSS properties for background

Property	Description	Values
background-color	color to fill background	A color
background-image	image to place in background	url(image URL)
background-repeat	whether/how bg image should be repeated	repeat(default), repeat-x, repeat-y or no-repeat
background	shorthand to set all background properties	



Background

```
body {  
    /* image to place in background */  
    background-image: url("../images/draft.jpg");  
    /* How bg image should be repeated */  
    /* can be repeat (default), repeat-x, repeat-y, or no-repeat */  
    background-repeat: repeat-x;  
    /* placement of bg image within element */  
    /* value consists of two tokens, can be top, left, right, bottom,  
       center, a percentage, or a length value in px, pt, etc */  
    background-position: 370px 20px;  
}  
p {  
    /* shorthand to set all background properties */  
    background: #ffffff url("image.png") no-repeat right top;  
}
```



[CSS Background Shorthand \(w3schools.com\)](https://www.w3schools.com/css/css_background_shorthand.asp)

CSS properties for borders

`border: border-width border-style border-color;`



```
p { border: 5px solid red; }
```

border-color, border-width, border-style, border-bottom, border-left, border-right, border-top,
border-bottom-color, border-bottom-style, border-bottom-width, border-left-color, border-left-style,
border-left-width, border-right-color, border-right-style, border-right-width, border-top-color,
border-top-style, border-top-width



```
h2 {  
    border-left: thick dotted #CC0088;  
    border-bottom-color: rgb(0, 128, 128);  
    border-bottom-style: double;  
}
```


Main Point

We discussed the CSS Properties for color, font, text, background, borders which are the basic properties used on almost every page.

There are a lot of details here, but few concepts. As long as you know the general concepts the details will follow with practice. Highest first—capture the fort to control the entire territory.

Exercise

- Add a background color to the body of the html demo page from yesterday

Lesson outline

- setting CSS properties
 - color, font, text, background, borders
- how rule sets are combined through cascading and inheritance
- selecting elements with id and class attributes

Cascading style sheets

- There are multiple sources of potentially conflicting CSS rules that might apply to any HTML element
- It's called Cascading Style Sheets because there is a set sequence (or cascade) for how rule sets are applied, and which sets have highest precedence
 1. Browser's default styles
 2. External style sheet files (in a `<link>` tag)
 3. Internal style sheets (in a `<style>` tag in the page header)
 4. Inline style (the style attribute of an HTML element)

Style inheritance

- Some styles, like font family, text-alignment etc., are automatically inherited by child elements from their parent element.



```
body { font-family: sans-serif; background-color: blue; }  
p { color: red; }  
a { text-decoration: underline; }  
h2 { font-weight: bold; text-align: center; }
```

Style inheritance and Body styles

- Some styles, like font family, text-alignment etc., are automatically inherited by child elements from their parent element.
- To apply a style to the entire body of your page, write a selector for the body element
- Saves you from manually applying a style to each element



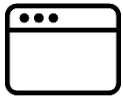
```
body {  
    font-size: 16px;  
}
```

Multiple selectors, Conflict & Overriding



```
/* select multiple elements separated by commas */
p, h1, h2 {
    color: green;
    background-color: grey;
}
/* when two styles set conflicting values for the same
property, the latter style takes precedence */

h2 {
    background-color: blue;
}
```



This paragraph uses the above style.

This h2 uses the above styles.

Lesson outline

- setting CSS properties
 - color, font, text, background, borders
- how rule sets are combined through cascading and inheritance
- selecting elements with id and class attributes

The HTML **class and id attribute**

- **id attribute** allows you to give a unique ID to any element on a page
 - Each ID must be unique; can only be used once in the page
- **class attribute** is used to group some elements and give a style to only that group
 - unlike an id, a class can be reused as much as you like on the page



class vs id examples



```
<p id="mission">Our mission is to provide the most</p>
<p class="special">See our spectacular spatula specials</p>
<p class="special shout">Today only, satisfaction guaranteed</p>
```

```
#mission {
  font-style: italic;
  color: #000000;
}
```



```
.special { /* any element with class="special" */
  background-color: yellow; font-weight: bold;
}
p.shout { /* only p elements with class="shout" */
  color: red;
  font-family: cursive;
}
```

Our mission is to provide the most

See our spectacular spatula specials

Today only, satisfaction guaranteed

Class naming

- focus on the semantics and meaning of the content vs appearance
- Bad example: redtext, bigfont
 - if change style later, it doesn't make sense to be called redtext.
- Good example:
 - warningMsg
 - errorMsg

CSS pseudo-classes

- A **pseudo-class** is used to define a special state of an element
 - Style an element when a user mouse's over it
 - Style visited and unvisited links differently
 - Style an element when it gets focus

class	description
:active	an activated or selected element
:focus	an element that has the keyboard focus
:hover	an element that has the mouse over it
:link	a link that has not been visited
:visited	a link that has already been visited



```
selector:pseudo-class { property:value; }
```

Examples pseudo-classes



```
/* unvisited link */
a:link {
    color: #FF0000;
}
/* visited link */
a:visited {
    color: #00FF00;
}
/* mouse over link */
a:hover {
    color: #FF00FF;
}
/* click on a link */
a:active {
    color: #0000FF;
}
```

More info and examples: [Pseudo-classes](#) and [Pseudo-elements](#)

Exercise

- Use a class selector like .special to change the style of several pieces of text in the html demo
 - use the tag to select and style text inside a <p>
 - <p>Sed ut perspiciatis, unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam eaque ipsa, quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt, explicabo. Nemo enim ipsam voluptatem, <p>

Main Point

The Cascading in CSS indicates that there are multiple levels of style sheets. More specific styles overwrite more general styles. We can be more specific by using Class selectors (can apply to multiple elements) and even more so with id selectors (individual elements) and context selectors.

Life is found in layers.

W3C CSS Validator

- Check your CSS to make sure it meets the official CSS specifications
- Pickier than the web browser, which may render malformed CSS correctly



```
<p>  
  <a href="https://jigsaw.w3.org/css-validator/check/referer">  
    </a>  
</p>
```


Explore more

- [CSS Tutorial \(w3schools.com\)](https://www.w3schools.com/css/)