

# Instant CSS

John M. Morrison

August 12, 2022

## Contents

<b>0</b>	<b>Introduction</b>	<b>1</b>
<b>1</b>	<b>Style Rules</b>	<b>2</b>
<b>2</b>	<b>Applying Styles <i>Ad Hoc</i></b>	<b>2</b>
<b>3</b>	<b>Selectors</b>	<b>3</b>
<b>4</b>	<b>Page Style Sheets</b>	<b>4</b>
<b>5</b>	<b>External Style Sheets</b>	<b>5</b>
<b>6</b>	<b>An Annoying Problem and a Solution</b>	<b>6</b>
<b>7</b>	<b>The Grouping Tags <code>div</code> and <code>span</code></b>	<b>7</b>
<b>8</b>	<b>Programming Activity</b>	<b>8</b>

## 0 Introduction

In this very brief article, you will learn some of the basics of CSS. These will provide visible and interesting examples when we start using JavaScript to manipulate and alter the document tree.

CSS can select elements from the document tree and apply style rules to them that alter their basic appearance.

# 1 Style Rules

A *style declaration* consists of two parts. The first part is a *property* and the second is a *value*. A typical CSS declaration looks like this.

```
color:blue;
```

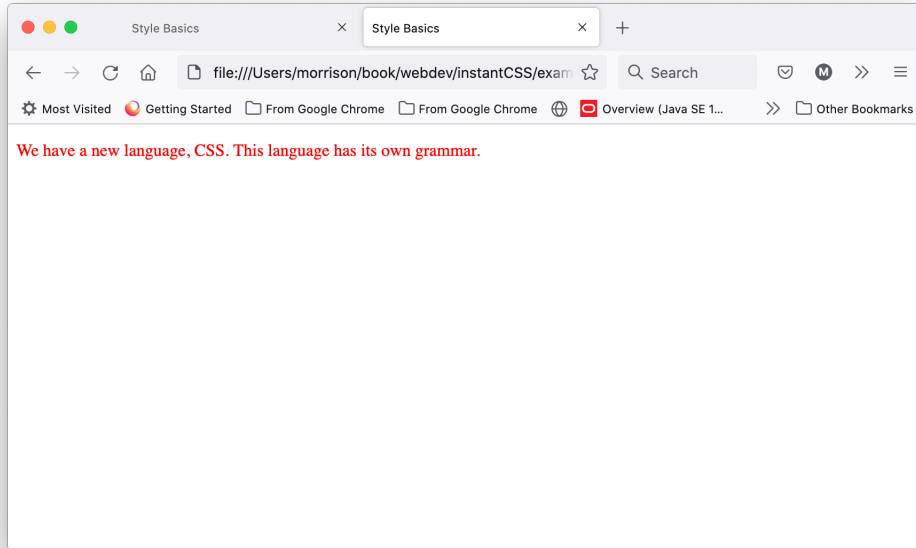
The property `color` controls the color of text in an HTML element. In an element using this style declaration, text is colored blue. Now for some syntax. The property and value are separated by a colon `:` and the statement is terminated by a `;` (semicolon). If these are missing, your browser will punt and ignore the rest of your style file.

# 2 Applying Styles *Ad Hoc*

You can impose a style declarations in any element. Notice the paragraph in this example.

```
<!doctype HTML>
<html lang="en">
  <head>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="main.css">
    <title>Style Basics</title>
  </head>
  <body>
    <p style="color:red;">We have a new language, CSS. This language
      has its own grammar.
    </p>
  </body>
</html>
```

Here is the rendered page.



All text within the paragraph turns red.

### Do-Now Programming Exercises

1. Add a paragraph before and after the paragraph on this page. What color are they?
2. Add this style declaration to the red paragraph  
`text-align:center;`  
What happens?

This method of applying style declarations should be used very sparingly. Typically it is used in situations where you are doing something that is a one-time exception to your general page appearance.

## 3 Selectors

In the intro, we said CSS selects page elements. There are several ways to do this. The most basic is by tag type. For example

```
h1  
{
```

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

selects all `h1` elements and centers the text within them. You can select all `h1` and `h2` elements this way.

```
h1, h2
{
    text-align:center;
}
```

The comma acts as an “or” operator; this selects all `h1` and `h2` elements. You can put as many tag types on the list as you wish.

A *style rule* is a selector followed by a list of style declaration enclosed in matching curly braces. You can place several style rules in a style declaration. This previous example we saw is a style declaration.

```
h1
{
    text-align:center;
}
```

## 4 Page Style Sheets

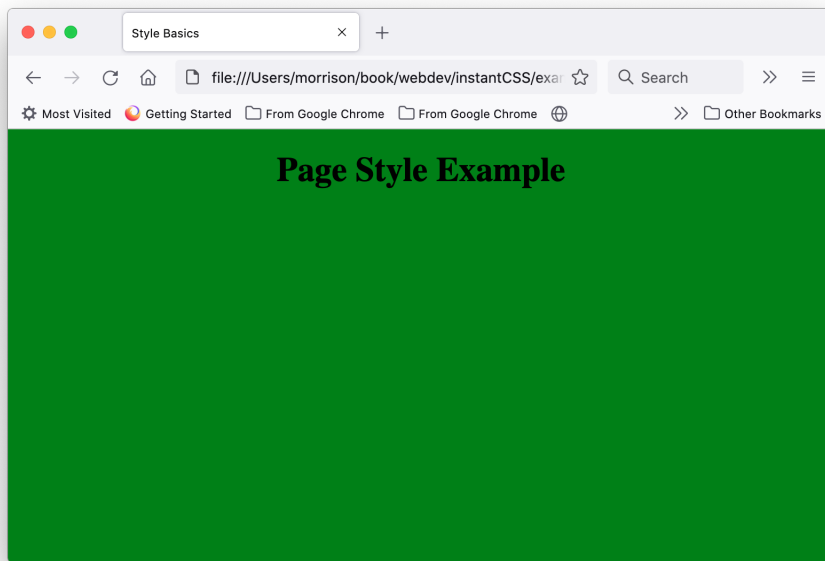
These style sheets work on just one page. They consist of style declarations places inside of a `style` element in the head of a page.

```
<!doctype HTML>
<html lang="en">
  <head>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="main.css">
    <title>Style Basics</title>
    <style>
      h1, h2
      {
        text-align:center;
      }
      body
      {
        background-color: green;
      }
    </style>
```

```
</head>
<body>
<h1>Page Style Example</h1>

</body>
</html>
```

Here is your rendered page.



You will see that the `h1` element is centered and that the page has a green background. These rules only apply on this page. Again, this kind of style sheet should only be used for exceptions to your general rules on a single page.

## 5 External Style Sheets

An external style sheet consists of style rules in a `.css` file. This is the by-far preferred means of imposing style rules on a page. One big advantage: a single CSS file can control the appearance of many pages. You make a change in a single file and the appearance of many pages is affected. This can give a website a consistent appearance.

So, how do we do this? Let us convert our previous example to this system. Note that the comment tokens in CSS are `/*` and `*/`.

**Step 1** Save the page CSS to an external file.

```
h1, h2  /* this is a selector */
{
    text-align:center;
}
body
{
    background-color: green;
}
```

Let's save this as `main.css` in the same directory as your HTML file, which is named `main.html`.

**Step 2** Delete the `style` element in the head of your page.

**Step 3** Link in your style sheet. Add this line to the head of your document.

```
<link rel="stylesheet" href="main.css">
```

This stylesheet can be linked into many documents. Additionally, you can link several stylesheets into a single document. Care must be taken to avoid precedence problems.

## 6 An Annoying Problem and a Solution

Suppose we create a web page and use inline styles to color important sections red. All of a sudden ... Marketing calls! That red is dreadful! It reminds our customers of Miss Wormwood's infamous grading pen! Make 'em green instead. Maybe we have hundreds of these throughout our website. Uh oh, an whole day of ugly search-and-replace excruciation is before us. It is unnecessary.

Another type of selector is a `class`. To make a class, just start your selector's name with a period (`.`). So, we might do this in the external stylesheets for our site.

```
.important
{
    color:red;
}
```

If we want a paragraph to be red, we do this.

```
<p class="important">This information is absolutely critical.</p>
```

Do not use the period in `class="important"`; it is there to say “this selector is a class selector.” Now, when those knuckleheads in Marketing call, you have ONE change to make and you are done.

Here is a common thing to do.

```
h1, h2, .display
{
    text-align:center;
}
```

All `h1` and `h2` elements, and any elements marked with `class="display"` have their text centered. The comma is a general “or” for all types of selectors.

**Do-Now Exercise** Create a page with an image and link in a stylesheet with the declaration we just created. Put an image inside of a paragraph marked with `class="display"`. What happens? Now combine this with an attribute of `style="width:50%"` for the image tag and what do you see?

## 7 The Grouping Tags `div` and `span`

Recall that there are two major types of elements, block-level and inline. Block-level elements bound a vertical chunk of a page. Inline elements bound a textographical chunk of a page.

The block-level `div` tag appears to do nothing. However you can select this tag, usually by a class, and apply style rule to a portion of the page. You would see something like this

```
<div class="foo">
    . (stuff)
    .
    .
</div>
```

Your CSS would include this.

```
.foo
{
```

```
}  
  .  
  . (style rules)  
}
```

You can do a similar thing to a span and your HTML would something like include this.

```
<p>We will apply <span class="foo">foo style rules inside  
of this span</span> and not outside.</p>
```

## 8 Programming Activity

Suppose you are given this HTML code.

```
<!doctype html>  
<!--Author: Morrison-->  
<!--Date: 2021-09-07-->  
  
<html lang="en">  
<head>  
<title>puzzler</title>  
<meta charset="utf-8"/>  
<link rel="stylesheet" href="puzzler.css">  
</head>  
<body>  
  <h1>Awful CSS Puzzler</h1>  
  
  <p>You are not to touch the HTML on this page. All of your work must  
  appear the file <code>puzzler.css</code>. <span class="nerd">Your  
  page must match the screenshot provided in this puzzler  
  project. </span></p>  
  
  <ul>  
    <li class="lucky">This list item is green. and its font is bold.  
    Check W3 schools for that.</li> <li class="nerd">This list  
    item and its bullet are cyan. That's color  
    <code class="nerd">#00FFFF</code>. </li>  
    <li>This list item is  
    normal-looking</li>  
  </ul>  
  
  <ol>  
    <li>I insist</li>  
    <li>that all ordered lists</li>
```



```

        <li>be enumerated with</li>
        <li>lower-case Roman Numerals</li>
    </ol>

<p>Yeah this ordered list too.</p>

<ol>
    <li>Now we</li>
    <li>see this miracle</li>
    <li>pulled off</li>
    <li>a second time</li>
</ol>

<p class="middle"> This paragraph's text is centered. Even if it goes
on for many lines and it begins to look like the <span
class="title">King James Bible</span> or <span
class="title">War and Peace</span>.</p>

<div class="squawk">
    <p>Here is a paragraph in the div.</p>

    <p>
    </p>

    <p>Notice the pretty colors and the border. You get a
    freebie: to center any block-level element use these
    two style rules.</p>

    <pre>
margin-left:auto;
margin-right:auto;
    </pre>
</div>
</body>
</html>

```

Here is a partial CSS file to go along with it.

```

*Author: Morrison*/
/*Date: 2021-09-07*/
.squawk
{
    width:80%;
    margin-left:auto;

```

```

margin-right:auto;
background-color:orange;
border:solid 1px black;
padding:1em;/* change this; tell me what it does in a comment.*/
}
img
{
  /*This is a freebie. What does it do?*/
  display:block;
  margin-left:auto;
  margin-right:auto;
  /* Put a black border of width 20 pixels around the image
   * that's your job.
   */
}

```

Without making any changes to the HTML, replicate this page. If you open Firefox, you can use its Eyedropper tool to determine colors. Download and scale the image so that you can use Eyedropper on any colored type.

### Awful CSS Puzzler

You are not to touch the HTML on this page. All of your work must appear the file `puzzler.css`. *Your page must match the screenshot provided in this puzzler project.*

- This list item is green, and its font is bold. Check W3 schools for that.
- This list item and its bullet are cyan. That's color `#00FFFF`.
- This list item is normal-looking

- i. I insist
- ii. that all ordered lists
- iii. be enumerated with
- iv. lower-case Roman Numerals

Yeah this ordered list too.

- i. Now we
- ii. see this miracle
- iii. pulled off
- iv. a second time

This paragraph's text is centered. Even if it goes on for many lines and it begins to look like the *King James Bible* or *War and Peace*.

Here is a paragraph in the div.



Notice the pretty colors and the border. You get a freebie: to center any block-level element use these two style rules.

```
margin-left:auto;
margin-right:auto;
```