

Intro to CSS

Intro to CSS

CSS = cascading style sheets

Intro to CSS

CSS = cascading style sheets

CSS was designed to solve a problem — specifically, that HTML was never designed for styling content.

Intro to CSS

CSS = cascading style sheets

CSS was designed to solve a problem — specifically, that HTML was never designed for styling content.

External style sheets will save you a LOT of work.

Intro to CSS

CSS = cascading style sheets

CSS was designed to solve a problem — specifically, that HTML was never designed for styling content.

External style sheets will save you a LOT of work.

External style sheets are saved as CSS files.

CSS syntax

CSS syntax

CSS rules have two main parts: a **selector**, and one or more **declarations**. The selector is generally written on one line, and the declarations are wrapped in angle brackets: { }

CSS syntax

CSS rules have two main parts: a **selector**, and one or more **declarations**. The selector is generally written on one line, and the declarations are wrapped in angle brackets: { }

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


CSS syntax

CSS rules have two main parts: a **selector**, and one or more **declarations**. The selector is generally written on one line, and the declarations are wrapped in angle brackets: { }

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

Selector

h1

Declaration(s)

color: blue

text-align:center

CSS declarations

CSS declarations consist of a **property** and a **value**. The property is the style attribute you want to change. Every property has a value.

CSS declarations

CSS declarations consist of a **property** and a **value**. The property is the style attribute you want to change. Every property has a value.

A CSS declaration always ends with a semicolon, and declarations are surrounded by curly brackets.

CSS declarations

CSS declarations consist of a **property** and a **value**. The property is the style attribute you want to change. Every property has a value.

A CSS declaration always ends with a semicolon, and declarations are surrounded by curly brackets.

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

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

CSS declarations

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

Selector

h1

Declaration(s)

color:blue
text-align:center

Property

color
text-align

Value

blue
center

CSS declarations

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

Selector	Declaration(s)	Property	Value
h1	color:blue text-align:center	color text-align	blue center

Declaration block: { color:blue; text-align:center; }

CSS comments

Comments serve two functions: to explain your code, and to activate/deactivate portions of your code. Comments are ignored by the browser.

CSS comments

Comments serve two functions: to explain your code, and to activate/deactivate portions of your code. Comments are ignored by the browser.

CSS comment syntax:

```
/* this is a comment */
```


CSS comments

Comments serve two functions: to explain your code, and to activate/deactivate portions of your code. Comments are ignored by the browser.

CSS comment syntax:

```
/* this is a comment */
```

```
/* p {text-align:center;} */
```

External, internal and inline styles

An **external stylesheet** is a .css file that's stored in your web folder and defines styles for a series of pages in your site. A change made to that file will change all your pages.

External, internal and inline styles

An **external stylesheet** is a .css file that's stored in your web folder and defines styles for a series of pages in your site. A change made to that file will change all your pages.

Internal styles can be used to style a single page. Internal styles are defined within the <head> tag, using the <style> tag. Normal css syntax applies inside the <style> tag.

External, internal and inline styles

An **external stylesheet** is a .css file that's stored in your web folder and defines styles for a series of pages in your site. A change made to that file will change all your pages.

Internal styles can be used to style a single page. Internal styles are defined within the <head> tag, using the <style> tag. Normal css syntax applies inside the <style> tag.

Inline styles can be used to style one occurrence of an element.

Embedding an external stylesheet

Here's the code you'd use to embed an external stylesheet:

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

Multiple styles will cascade into one style

Styles can be specified in three ways:

Multiple styles will cascade into one style

Styles can be specified in three ways:

- * inside an HTML element
- * in the head element of an HTML page
- * in an external CSS file

Multiple styles will cascade into one style

Styles can be specified in three ways:

- * inside an HTML element
- * in the head element of an HTML page
- * in an external CSS file

Multiple style sheets can be referenced inside a single HTML document (although this is confusing)

Multiple styles will cascade into one style

Styles will “cascade” into a new set of styles following the following four rules:

Multiple styles will cascade into one style

Styles will “cascade” into a new set of styles following the following four rules:

- 1) browser default
- 2) external style sheet
- 3) internal style sheet (in head element)
- 4) inline style (inside HTML element)

Multiple styles will cascade into one style

Styles will “cascade” into a new set of styles following the following four rules:

- 1) browser default
- 2) external style sheet
- 3) internal style sheet (in head element)
- 4) inline style (inside HTML element)

So, an inline style will override a style in the head element, which will override an external style sheet, which will override browser default styles.

CSS tag, class and ID selectors

You can specify which part of your document to apply CSS to using tag, class or ID selectors.

CSS tag, class and ID selectors

You can specify which part of your document to apply CSS to using tag, class or ID selectors.

A tag selector applies to an html tag: `<p>`

A class selector applies to an item that's been given a class: `<p class="first">`

An ID selector applies to an item that's been given an ID: `<div id="nav">`

CSS tag, class and ID selectors

Selector syntax:

	HTML	CSS
Tag	<code><p></code>	<code>p {declaration: property;}</code>
Class	<code><p class = "first"></code>	<code>.first {declaration: property;}</code>
ID	<code><p ID= "nav"></code>	<code>#nav {declaration; property;}</code>

CSS tag, class and ID selectors

Additional notes about tags, classes and IDs:

- * Tag selectors in the syntax above apply to all tags in the document, unless further specifications are given.
- * Class selectors can be used multiple times (the point is to create a “class” of similar objects)
- * ID selectors may only be used once.

Tags, classes, IDs, divs and spans

If you recall the xhtml lecture, you'll also remember that we touched briefly on two generic html tags: **div** and **span**.

Tags, classes, IDs, divs and spans

If you recall the xhtml lecture, you'll also remember that we touched briefly on two generic html tags: **div** and **span**.

These tags do not have any semantic meaning, and do not by themselves affect document style. They are used to organize content and, often, to write rules to target specific parts of your document.

Tags, classes, IDs, divs and spans

For example, I might write the following html:

Tags, classes, IDs, divs and spans

For example, I might write the following html:

```
<body>
  <div id="main">
    <h1>Header</h1>
    <p><span class="first">Content goes here</span> on several
    paragraphs.</p>
  </div>
  <div id="sidebar">
    <p>Sidebar content goes here</p>
  </div>
</body>
```

Tags, classes, IDs, divs and spans

This css selector:

#main {some-css-rule:here}

would target this part of the html:

```
<body>
  <div id="main">
    <h1>Header</h1>
    <p><span class="first">Content goes here</span> on several
    paragraphs.</p>
  </div>
  <div id="sidebar">
    <p>Sidebar content goes here</p>
  </div>
</body>
```

Tags, classes, IDs, divs and spans

This css selector:

#sidebar {some-css-rule:here}

would target this part of the html:

```
<body>
  <div id="main">
    <h1>Header</h1>
    <p><span class="first">Content goes here</span> on several
    paragraphs.</p>
  </div>
  <div id="sidebar">
    <p>Sidebar content goes here</p>
  </div>
</body>
```

Tags, classes, IDs, divs and spans

and this css selector:

.first {some-css-rule:here}

would target this part of the html:

```
<body>
  <div id="main">
    <h1>Header</h1>
    <p><b>span class="first">Content goes here</span> on several
    paragraphs.</p>
  </div>
  <div id="sidebar">
    <p>Sidebar content goes here</p>
  </div>
</body>
```

Defining colors in CSS

Keyword: black, red, green

Defining colors in CSS

Keyword: black, red, green

Hex: #CC3300 (white: #FFFFFF; black: #000000)

Defining colors in CSS

Keyword: black, red, green

Hex: #CC3300 (white: #FFFFFF; black: #000000)

Short hex: #C30

Defining colors in CSS

Keyword: black, red, green

Hex: #CC3300 (white: #FFFFFF; black: #000000)

Short hex: #C30

RGB decimal: rgb(204, 51, 0) (white: 255, 255, 255; black 0, 0, 0)

Defining colors in CSS

Keyword: black, red, green

Hex: #CC3300 (white: #FFFFFF; black: #000000)

Short hex: #C30

RGB decimal: rgb(204, 51, 0) (white: 255, 255, 255; black 0, 0, 0)

RGB percent: rgb(80%, 20%, 0%) (white: 100% 100% 100%; black 0%, 0%, 0%)

Defining colors in CSS

Keyword: black, red, green

Hex: #CC3300 (white: #FFFFFF; black: #000000)

Short hex: #C30

RGB decimal: rgb(204, 51, 0) (white: 255, 255, 255; black 0, 0, 0)

RGB percent: rgb(80%, 20%, 0%) (white: 100% 100% 100%; black 0%, 0%, 0%)

Full list of CSS color names

Styling links in CSS

A common use of CSS is to style links for different states. Links can be styled with any CSS property, but there are four states links are styled in:

Styling links in CSS

A common use of CSS is to style links for different states. Links can be styled with any CSS property, but there are four states links are styled in:

- * a: link
- * a: visited
- * a: hover
- * a: active

Styling links in CSS

A common use of CSS is to style links for different states. Links can be styled with any CSS property, but there are four states links are styled in:

- * `a: link` — a standard unvisited link
- * `a: visited` — a link the user has visited
- * `a: hover` — a link being hovered over
- * `a: active` — the link when the user clicks it

Styling links in CSS

A common use of CSS is to style links for different states. Links can be styled with any CSS property, but there are four states links are styled in:

- * a: link — a standard unvisited link
- * a: visited — a link the user has visited
- * a: hover — a link being hovered over
- * a: active — the link when the user clicks it

Syntax notes:

a: hover **must** come after a: link and a: visited
if using a: active, a: active **must** come after a: hover
so, basically write what you see up top there.

The CSS box model

http://www.w3schools.com/css/css_boxmodel.asp

