

# *Using CSS with XHTML*

Martin Holmes

## Using CSS with XHTML

- You know how to write XSLT stylesheets to transform XML into XHTML.
- But now you have XHTML, how do you style it?
- Cascading Style Sheets (CSS) are the answer.

## What is CSS?

- Simple, declarative syntax for positioning, styling, colouring and outlining things.
- Very human-readable:

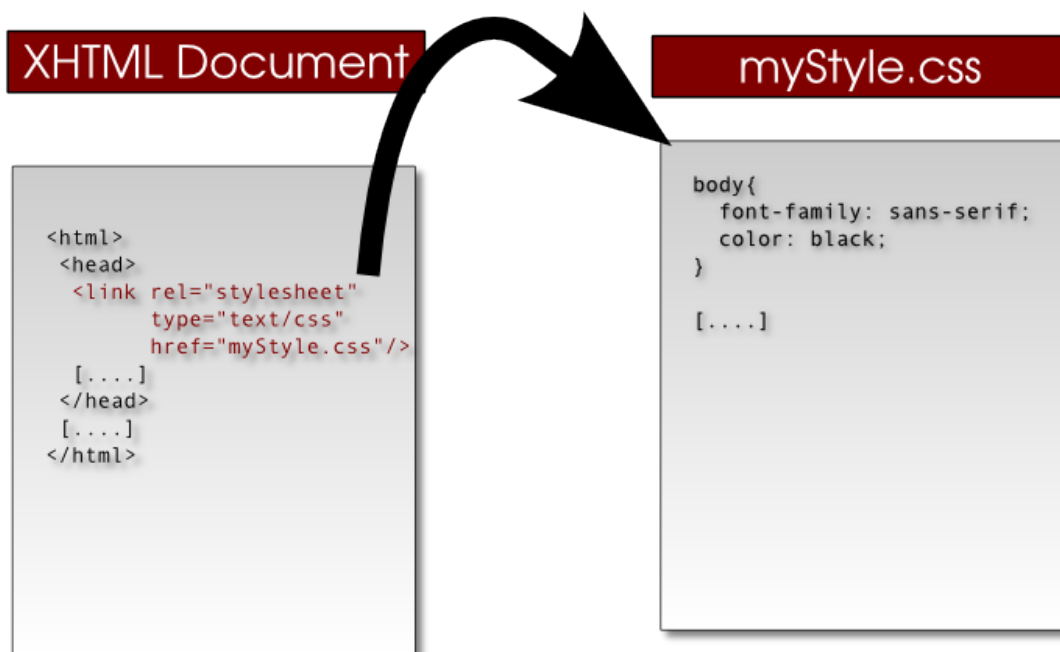
```
font-style: italic;
```

means exactly what you would expect.

- Very powerful and rich, especially recent 2.1 and 3.0 versions.

## Getting started: how to link your XHTML file to a CSS file

Linking an XHTML document to a CSS stylesheet



## The <link> element

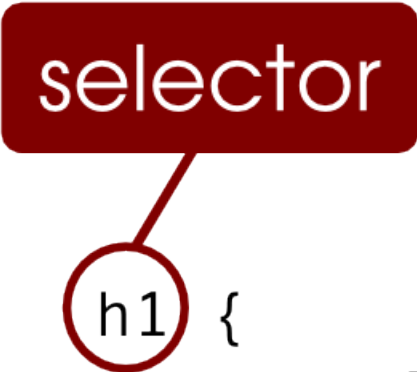
```
<link rel="stylesheet"  
      type="text/css"  
      href="myStyle.css" />
```

## CSS Rulesets (1)

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

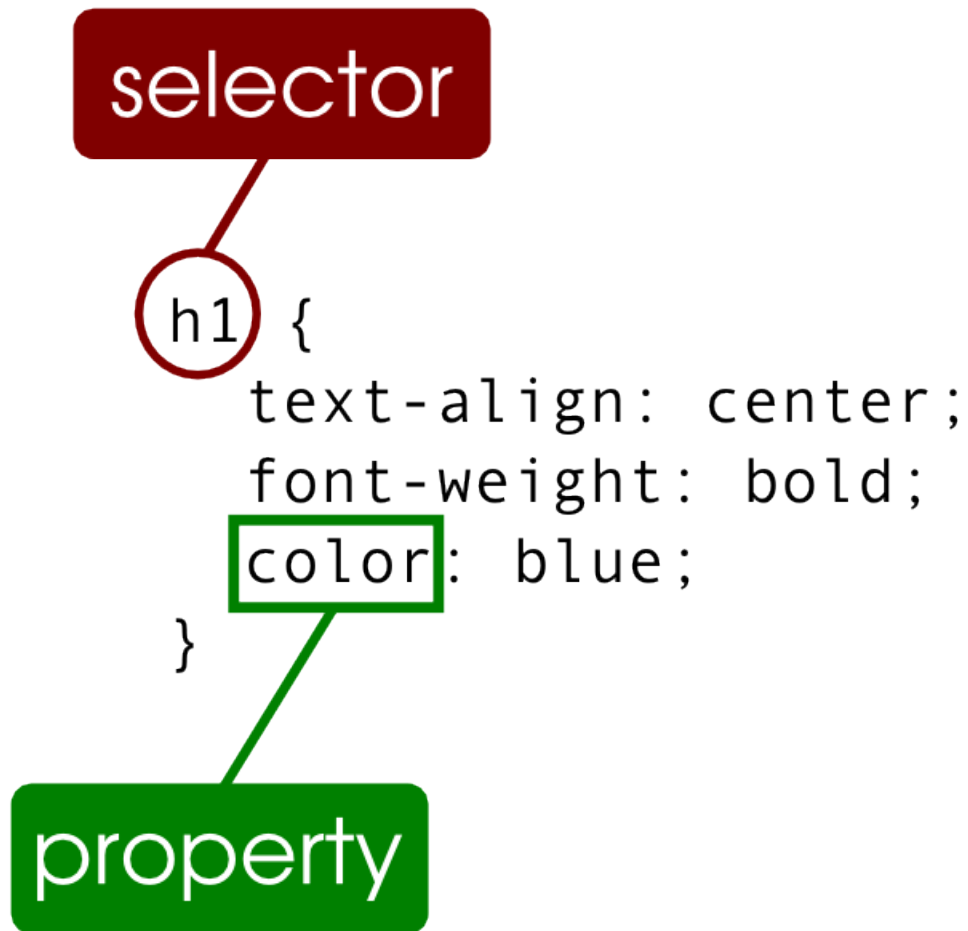
## CSS Rulesets (2): Selector

selector

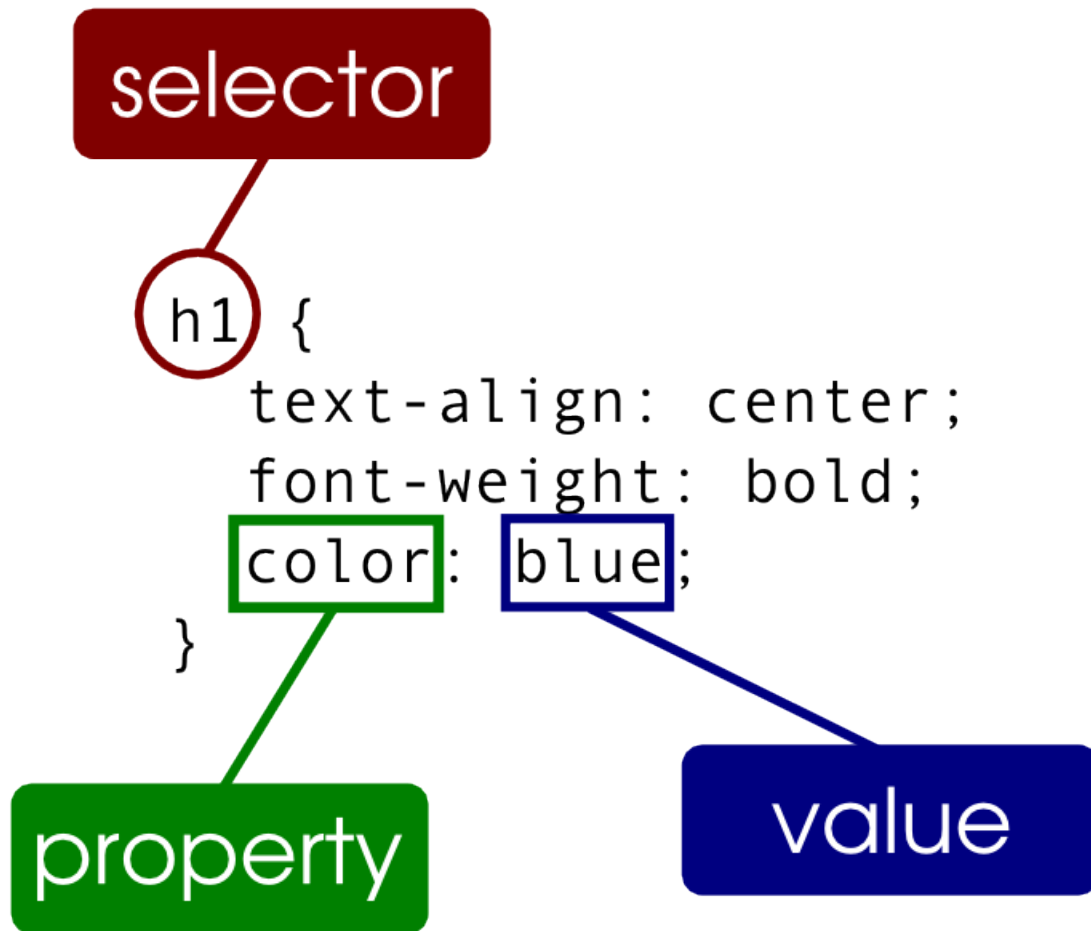


```
h1 {  
  text-align: center;  
  font-weight: bold;  
  color: blue;  
}
```

## CSS Rulesets (3): Property



## CSS Rulesets (4): Value





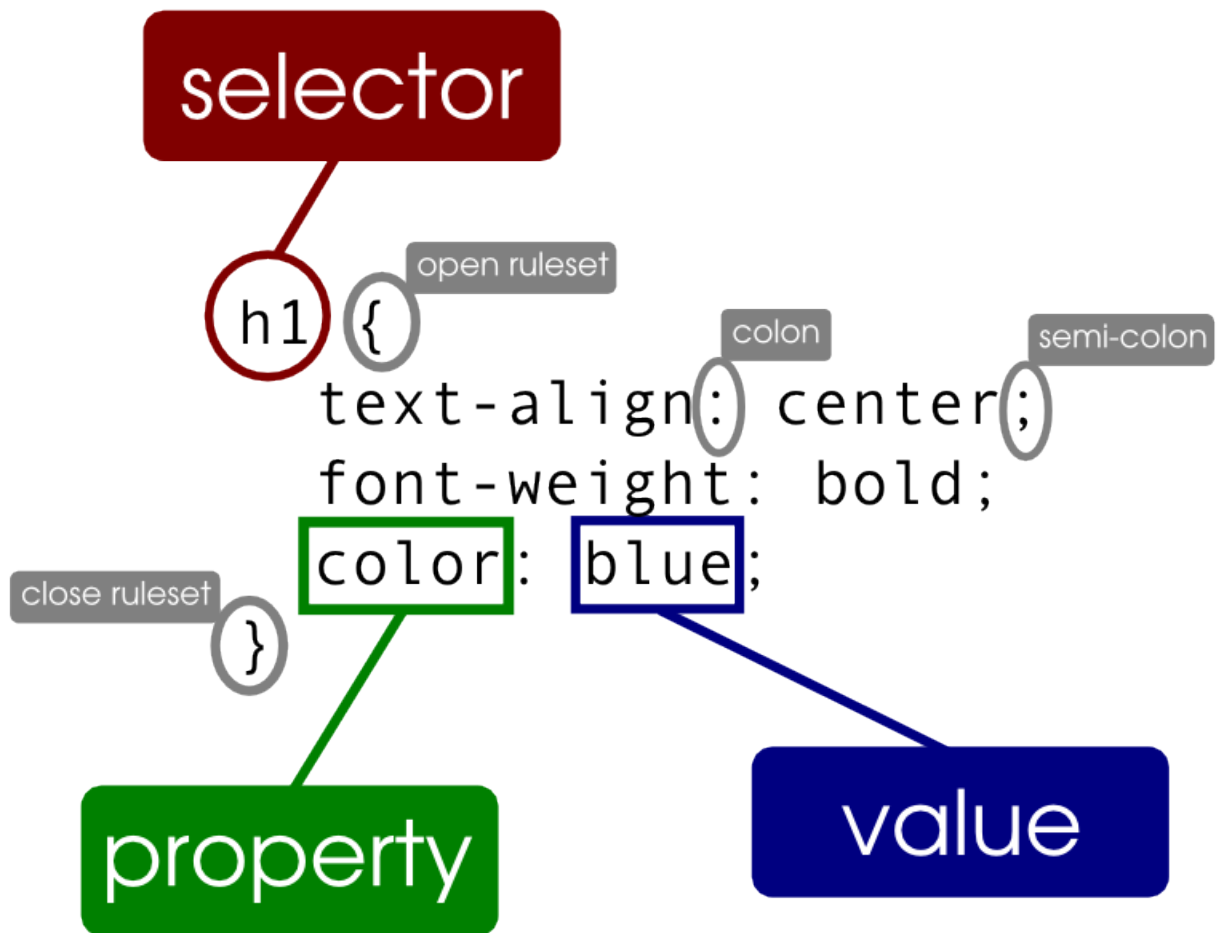
## CSS Rulesets (5): Punctuation

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

Diagram illustrating the punctuation in a CSS rule set:

- open ruleset**: Points to the opening curly brace `{`.
- colon**: Points to the colon `:` in the first property declaration.
- semi-colon**: Points to the semicolon `;` at the end of the first property declaration.
- close ruleset**: Points to the closing curly brace `}`.

## CSS Rulesets (6): Summary



## CSS Selectors (1)

- `h1`  
Select by element name (`<h1>`)

- `span.placeName`  
Select by element and class, using a dot (`<span class="placeName">`)
- `p span.placeName`  
Select by descendant: all `<span class="placeName">` elements inside a `<p>` element
- `h2 > span.bookTitle`  
Select by child using ">": all `<span class="bookTitle">` elements which are direct children of an `<h2>` element
- `h2#mainTitle`  
Select by id (`<h2 id="mainTitle">`)

## CSS Selectors (2)

- `.placeName`  
Select by class, using a dot (any element with `class="placeName"`)
- `#subTitle`  
Select by ID (without needing to specify the element name)
- `h1, h2, h3.chapterTitle`  
Group several selectors together, comma-separated (`<h1>`, `<h2>`, `<h3 class="chapterTitle">`)

## CSS Selectors: Task

How would you write a selector for:

- **both `<p>` and `<blockquote>` elements?**
- **a `<div>` element with `@id "notes"` ?**

- a `<p>` with a `@class` of `"intro"` which is inside a `<div>` with a `@class` of `"front"` ?

## CSS Selectors: Task answer 1

How would you write a selector for:

- both `<p>` and `<blockquote>` elements?

- ```
p, blockquote { ... }
```

## CSS Selectors: Task answer 2

How would you write a selector for:

- a `<div>` element with `@id "notes"` ?

- ```
div#notes { ... }
```

## CSS Selectors: Task answer 3

How would you write a selector for:

- a `<p>` with a `@class` of `"intro"` which is inside a `<div>` with a `@class` of `"front"` ?

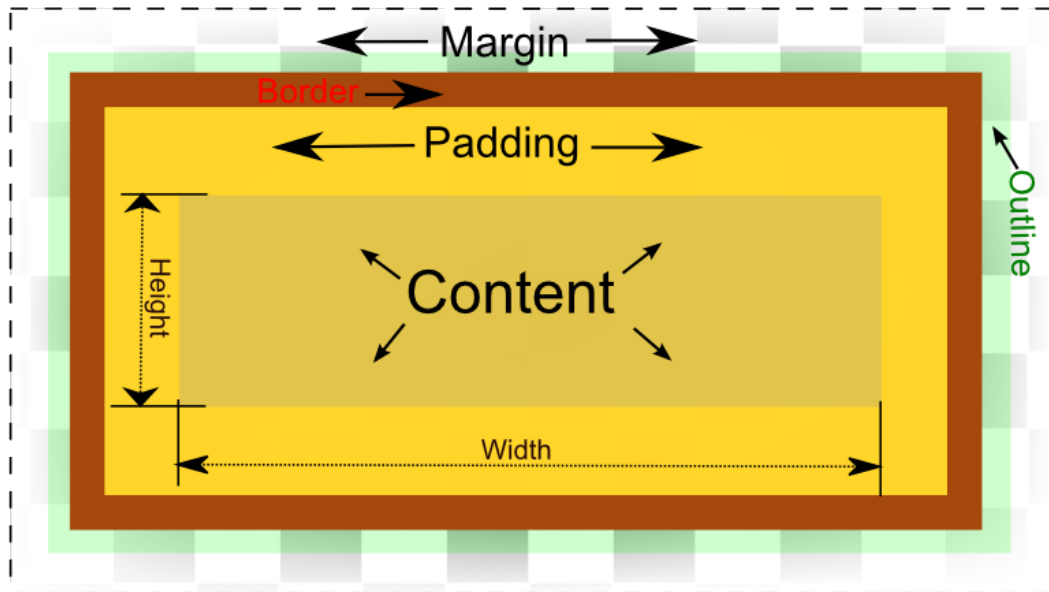
- ```
div.front p.intro { ... }
```

## CSS Units

- CSS property values which involve measurement can be expressed in several different units:
- Physical measurements: **in**, **cm**, **mm**, **pt**, **pc** (caution)
- Screen-related units: **px** (don't use)
- Relative units: **em**, **ex**, **%** (the best options)

## CSS Box model

This is how padding, margins, borders etc, work for block elements in CSS:



(Image courtesy of Felix.leg/Wikimedia)

## CSS Properties

- Too many to list exhaustively
- Lots of useful references on the web (e.g. <http://www.w3schools.com/css/>)
- Browse the references for ideas
- Figure out what you want to do and then Google, ask, read specs, etc.

## CSS: A detailed example

```
div.warningBox{
  border: solid 2px black;
  background-color: white;
  color: red;
  margin: 1em;
  text-align: center;
  font-weight: bold;
  line-height: 150%;
}
```

## CSS: Task

Write a ruleset that:

- applies to all paragraphs which are children of a <div> with an @id of "afterword"
- indents the first line of the paragraph by 1 em
- justifies the paragraph
- sets a font size which is twice as big as the default
- draws a dotted border around the paragraph
- leaves a space of 2 ems between the border and the text

## CSS: Task answer

```
div#afterword p{
  text-indent: 1em;
  text-align: justify;
  font-size: 200%;
  border-width: 1px;
  border-style: dotted;
  padding: 2em;
}
```