



warwick
C<>ding



Session 4

Getting Fancy with CSS



Agenda/Objectives

- Continue using koding.com text editor
- Understand the fundamentals of CSS
- Selectors and properties
- Why and what is Cascading?
- Exercises



Last week recap

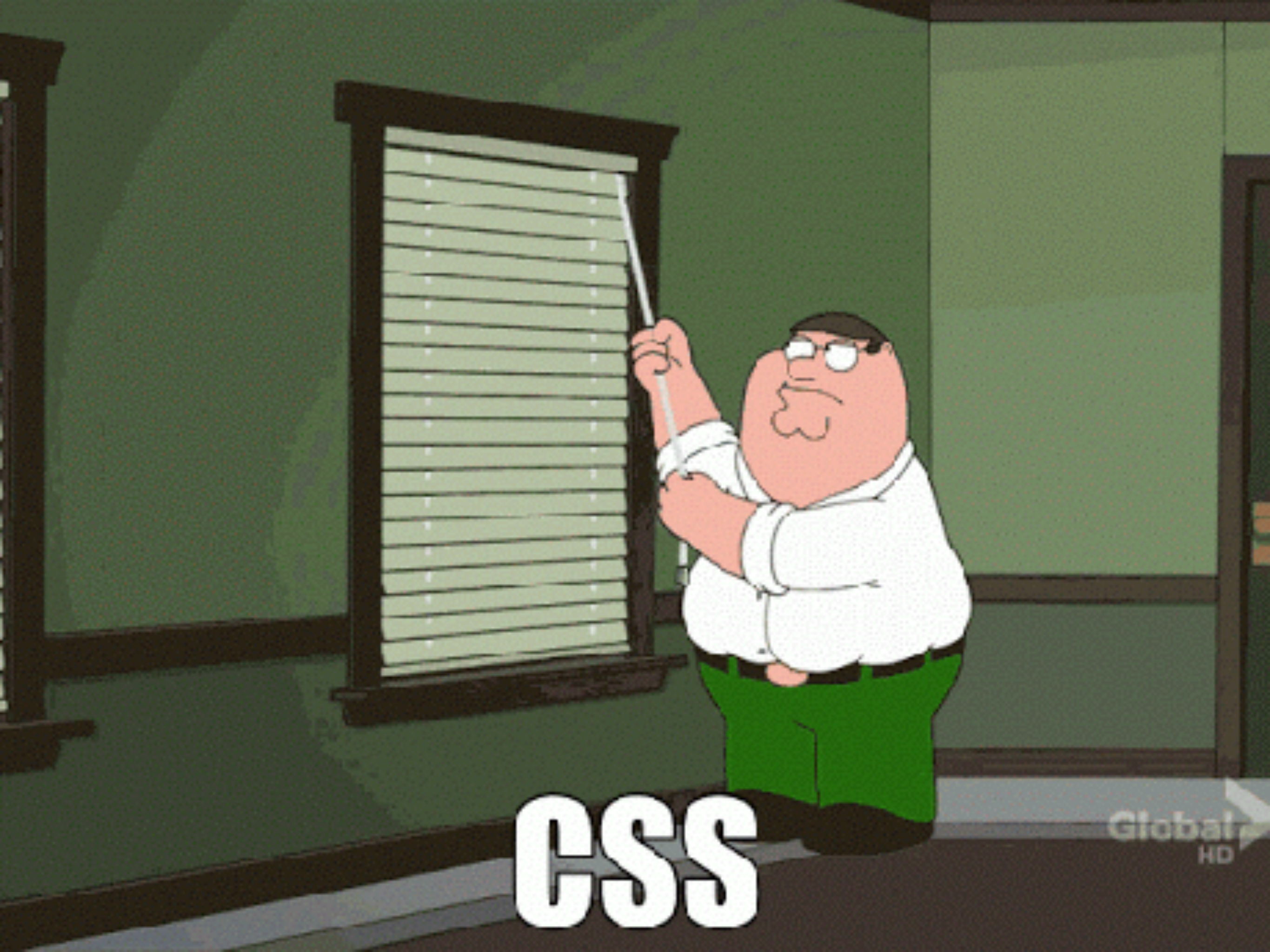
- HTML - HyperTextMarkupLanguage (Content)
- Forms, Tables, etc...
- Differentiate between `<tag>`, `element` & `attribute`
- And?

What is this CSS that you
speak of?



CSS





css

Global
HD



Can we live without CSS though?

www.facebook.com



CSS powers

Style Text

Color

Size

Position



CSS Anatomy

CSS syntax typically follows the “Selector” and “Declaration” of property-value pairs

Example

```
selector {  
  property: value;  
  property: value;  
}
```

```
body {  
  color: red;  
  background-color: green;  
}
```



CSS Embedding

1. We can embed our CSS inside the `<head>` tag
2. We can embed the rules as an attribute inside a specific tag (inline)
3. We can put all our CSS rules in a separate file.



1. Internal Stylesheet

```
<head>
  <style>
    body {
      background-color: gray;
    }
    h1 {
      color: pink;
      margin-left: 12px;
    }
  </style>
</head>
```



2. Inline Styles

```
<h1 style="color:blue;margin-left:30px;">This is a heading.</h1>
```



3. Separate Stylesheet

HTML

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



Selectors

```
selector {  
    property: value;  
    property: value;  
}
```




How can we style things?





Types of Selectors

1. element
2. id
3. class
4. position in document

Types of Selectors



element

CSS

```
p {  
  color: blue;  
}
```

HTML

```
<p>Hello!!!</p>
```

Types of Selectors



id



CSS

```
#facebook-button {  
  color: pink;  
  margin-left: 12px;  
}
```

HTML

```
<button id="facebook-button"></button>
```

Types of Selectors



class

Instagram

Search

zlahham

CSS

```
.nav-bar {  
  border-top: 1px;  
  border-bottom: 1px;  
}
```

HTML

```
<div class="nav-bar"></div>
```



Types of Selectors

position in document

CSS

```
ul li {  
  list-style: none;  
}
```

HTML

```
<body>  
  <ul>  
    <p>A list of items</p>  
    <li>item1</li>  
    <li>item2</li>  
    <li>item3</li>  
  </ul>  
</body>
```




More Positioning Examples

Types of selectors



pseudo classes

```
a:link { /* unvisited link */
  color: red;
}
a:visited { /* visited link */
  color: blue;
}
a:hover { /* moused over link */
  color: green;
}
a:active { /* current link */
  color: purple;
}
a:focus { /* focused link */
  color: purple;
}
```



Cascading Rules

- id is more specific than a class
- class is more specific than an element
- the longer the selector, the more specific it is
- If style rules are equally specific, the last one wins



Style Conventions

- Space between selector and {
- 2 spaces before the property-value pair
- semi-colon after value
- line between each rule
- When naming classes and ids, describe content not the style.
- Use all lowercase and use hyphens instead of spaces



Properties



property-value pairs

- Each property can have one or more comma separated values

```
color: #333;  
background-color: red;  
font-family: Arial, sans-serif;
```

Value 1 Comma Value 2



Properties: color

- The “color” property changes the text color. Colours can be specified either by name, for the most common colours, or by hexadecimal value.

```
color: red;  
color: #ff0000;  
color: rgb(255,0,0);
```

- This property is inherited, which means it'll also be applied to all descendant elements but can be overridden by more specific rules. this rule makes all the body text red unless specified otherwise:

```
body {  
    color: red;  
}
```



Properties: background-color

- The “background-color” property changes the background color. Besides the “Body” and “Table” element, all elements default to a transparent background.

```
background-color: black;  
background-color: #000000;  
background-color: rgb(0,0,0);
```



```
body {  
    background-color: yellow;  
}  
  
table {  
    background-color: #ffcc00;  
}
```



Finding Colors

To find colors or color schemes for your webpage, you can use:

- Desktop graphics programs: Adobe Photoshop, Mac Colors
- Online tools: Kuler
- Browser extensions: Eye Dropper for Chrome



Properties: font-family

- The “font-family” property specifies the font family (or “font face”) of the text. You can specify either a specific font name or a generic family name (serif, sans-serif, monospace, cursive).

```
font-family: "Times New Roman";  
font-family: sans-serif;
```

- A comma separated list of font families can be specified if you want the browser to prefer one but use the others as backup options.

```
font-family: "Times New Roman", serif;  
font-family: "Arial", sans-serif;  
font-family: Courier, monospace;
```



Finding fonts

Several websites exist that help you pick fonts for your website and to customise it with non-default fonts:

- TypeTester
- TypeKit
- Google Fonts



Property: font-size

- The “font-size” property specifies the size of a font. it can be specifies as a fixed size in various units, a percentage, or as a predefined keyword.

```
font-size: 1.5em;
```

```
font-size: 12px;
```

```
font-size: 100%;
```

```
font-size: larger;
```




Property: font-size(em)

- The “em” unit lets you set the size of the text relative to the text around it. this makes the page resize nicely in proportion if the user changes their default font-size.
- the default size is “1em”

```
p {  
  font-size: 0.9em;  
}
```



Hello**World!**

```
strong {  
  font-size: 1.5em;  
}
```



Property: font-size(px)

- The “px” unit lets you size font in terms of pixels, which is the unit also used to size images and other elements.
- It is easier to understand than em, but doesn’t work as well when printing or resizing.

```
h2 {  
  font-size: 17px;  
}
```



Normal Headline

Resized Headline



Properties: font-size(keywords)

- There are various keywords that can be used if you're not as worried about the precise sizing: xx-small, x-small, small, medium, large, x-large, xx-large.

```
p.footnote {  
  font-size: small;  
}
```

- There are also two relative keywords, that act similar to em's in setting size relative: smaller, larger.

```
p.intro {  
  font-size: larger;  
}
```



Property: font-size(%)

- The size can also be specified as a percentage, which works similar to “ems”, and can be used in conjunction with other units.

```
body {  
  font-size: 12px;  
}
```

```
h1 {  
  font-size: 200%;  
}
```

```
h1 a {  
  font-size: 75%;  
}
```




Header link

text



Property: font-style

- The “font-style” property specifies the font style of the text, either “normal” by default or “italic”.

`font-style: italic;`  *italic text!*



Property: font-weight

- The “font-weight” property specifies the thickness of the font. the default is “normal” and the “typical” override is “bold”. You can also specify “bolder”, “lighter”, or a number from 100-900.

`font-weight: bold;`  **BOLD text!**



“Shorthand” properties

- A “shorthand” property in CSS lets you specify multiple properties in one property, for conciseness purposes.
- Instead of specifying each “font-” property separately, you can bundle them up in one “font” property.

```
table.geeky {  
    font-weight: bold;  
    font-style: italic;  
    font-size: 10px;  
    font-family: sans-serif;  
}
```



Instead...

```
table {  
  font: italic bold 10px sans-serif;  
}
```




Other text properties

```
line-height: 10px;
```

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

```
text-decoration: underline;
```

```
text-indent: 20px;
```



Exercises

github.com/warwickcoding/frontend



Important Resources

- [20 CSS tips for beginners](#)
- [CSS Tricks \(Recommended\)](#)
- MozillaDeveloperNetwork MDN
 - <https://developer.mozilla.org/en-US/docs/Web/CSS>
 - <https://developer.mozilla.org/en-US/docs/Web/HTML>
- [Quick HTML CSS Prototyping with Codepen](#)



```
.ninja {  
visibility: hidden;  
color: black;  
}
```