

CS1115/CS5002

Web Development 1

Dr Derek Bridge

School of Computer Science & Information Technology
University College Cork

Font properties

- There are various font properties, e.g.

- font-family
- font-size
- font-style
- font-weight

E.g.

```
p {  
  font-family: "Times New Roman";  
  font-size: 12px;  
  font-style: italic;  
  font-weight: bold;  
}
```

- There is also a shorthand — but it's unintuitive so avoid it!

```
p {  
  font: italic bold 16px "Times New Roman";  
}
```

Text formatting

- We can set the color of text:

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

- There are text properties, including text-align, e.g.:

```
p {  
  letter-spacing: 1em;  
  line-height: 90%;  
  text-align: right;  
  text-indent: 1em;  
  text-transform: uppercase;  
  word-spacing: 2em;  
}
```

- There are text decoration properties, e.g.:

```
p {  
  text-decoration: underline;  
  text-shadow: 0.125em 0.0625em gray;  
}
```

Fonts

- Some fonts are **proportional**, others are **monospace**
- Some fonts have **serifs**, others are **sans-serif**
- Very dated advice is to use serif for print and sans-serif for screens

Fonts on the Web

- You do not know what fonts will be available on your user's machine
 - different platforms come with different fonts installed
 - many software packages install further fonts
- E.g. see <http://www.cssfontstack.com/>

Downloadable fonts

- CSS has downloadable fonts using the `@font-face` rule

```
@font-face {
  font-family: MuseoSans;
  src: url('fonts/museosans-500-webfont.woff2') format('woff2');
}

p {
  font-family: MuseoSans, Arial, sans-serif;
}
```

- The browser requests the `museosans-500-webfont.woff2` file from the URL
- Problem: copyright!
(Even buying a copy may not entitle you to put it on your server — since then you're effectively giving it away)
- Problem: there is a performance hit

Font families

- Use the `font-family` property but give it multiple values (a **font stack**) to cover multiple platforms, e.g.:

```
body {
  font-family: Verdana, Geneva, Tahoma, sans-serif;
}
```

- First give the ideal font, then some good but not ideal alternatives, then some common alternatives, then a **generic**
- The generics are: **serif**, **sans-serif**, **cursive**, **fantasy**, **monospace**

Google Web Fonts

- Google hosts many fonts, free for use: <https://fonts.google.com/>
- Choose a font and Google will give you code to put into your CSS, e.g.

```
@import 'https://fonts.googleapis.com/css?family=Open+Sans';

p {
  font-family: 'Open Sans', sans-serif;
}
```

(`@import` must come at the start of the whole stylesheet)

- Problem: can be an even bigger performance hit!

Font sizes

- The font-size property, e.g.:

```
section {  
  font-size: 2em;  
}
```

- Absolute font sizes:

Pixels	e.g. 16px
Points = (1/72th inch)	e.g. 12pt
Others: picas, inches, millimetres, centimetres	

- Relative font sizes:

Percentages	e.g. 80%	relative to inherited font size
em	e.g. 2em	relative to inherited font size
rem (root em)	e.g. 2rem	relative to root's font size (i.e. <body> or <html>)
Others: ex, ch, vh, vw, vmin, vmax		

Relative font sizes: class exercise

```
<body>  
...  
<aside>  
  <h1>Did you know?</h1>  
  <p>  
    <p> Mojito is pronounced Moh-HEE-toh.  
    ...  
  </aside>  
  ...  
</body>
```

```
body {  
  font-size: 16px;  
}  
aside {  
  font-size: 2em;  
}  
h1 {  
  font-size: 1.5em;  
}  
p {  
  font-size: 1.5rem;  
}
```

- How big is the text in the h1?
- How big is the text in the p?

Font sizes: Advice

- For your root font (<body>):
 - in a print stylesheet, use points
 - in a regular stylesheet, use pixels

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

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

- Then use em, % or rem for all other rules
 - Some people prefer rem — to avoid thinking about complex cases of inheritance!
 - But browser default stylesheets tend to use em (and so do I), e.g.:

```
h1 {  
  font-size: 2em;  
}  
section h1 {  
  font-size: 1.5em;  
}  
section section h1 {  
  font-size: 1.17em;  
}  
section section section h1 {  
  font-size: 1em;  
}
```