

CF:G INTRODUCTION TO WEB DEVELOPMENT

Week 2: Cascading style sheets (CSS)

WHAT WE'LL COVER THIS WEEK

- What is CSS
- How you can link your HTML to your CSS
- How to write CSS
- More about the HTML `<link>` Tag
- Selectors and Attributes
- The Universal Selector

A lot of this lesson is a 'code-along' session, so make sure you keep up and ask us to pause when needed!

RECAP

- A website is a collection of files in a folder
- The folder can contain HTML, CSS and JavaScript files
- We can create and edit the files locally
- We can view the files locally in a browser

website-folder

-- index.html

-- page.html

-- **images**

-- picture.jpg

-- **css**

-- style.css

-- **js**

-- script.js

CHAPTER 4: WHAT IS CSS?

Cascading Style Sheets

- Adds styles to the HTML document
- One HTML file can be styled in an infinite number of ways with CSS

CHAPTER 5: LINKING CSS TO HTML

In a separate CSS file

```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" href="css/styles.css">
  </head>
  ...
```

- Separates content from design, avoiding cluttered HTML
- Reduce repetition of code
- One CSS file for a whole website

LINKING CSS AND HTML CODE (OLD SCHOOL WAYS)

Inline in HTML

```
<p style="color: red">
```

Can be useful very rarely, should be avoided

Inside <head> element

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      p {
        color: red;
      }
    </style>
  </head>
  ...
```

Better, but should still be avoided

THE ANATOMY OF THE LINK ELEMENT

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

relation

What the relation
is of the linked file
to this document

type

The file
type

file

Where the
CSS file is
located

TYPES OF LINKS

- Absolute links
- Root-relative links
- Document-relative links

first_site

– index.html

– images

–

background.jpg

– css

– main.css

ABSOLUTE LINKS

In HTML

```
<link rel="stylesheet" type="text/css"
href="http://www.first_site.com/css/main.css">
```

In CSS

```
body {
    background-image:
url("http://www.first_site.com/images/backgroun
d.jpg");
}
```

first_site

– index.html

– images

–

background.jpg

– css

– main.css

ROOT-RELATIVE LINKS

In HTML

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

Root-relative links
always start with /

first_site

– index.html

– images

–

background.jpg

– css

– main.css

DOCUMENT-RELATIVE LINKS

In HTML

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

In CSS

```
body {
    background-image:
url("../images/background.jpg");
}
```

first_site

– index.html

– images

–

background.jpg

– CSS

– main.css

TASKS

1. Open the exercise you completed in the last session in atom and Chrome. (if you haven't completed it, you can copy the solution [from here](#)). Make sure you open the whole folder in atom, not just the 'index.html' file.
2. In the exercise folder, create a new folder called 'css'
3. Inside this, create a new file called 'styles.css'
4. Follow along with the subsequent slides, adding the css and **lines in red** as you go!

Chapter 7: Splitting the page up... [index.html](#)

First, we're going to split the page up into sections using dividers or 'div's

```
<div>
  <h1>Hello World!</h1>
  <h2>Welcome to my site</h2>
  
</div>
<div>
  <p>
    I am currently learning to code...
  </p>
  <ol>
    ...
  </ol>
</div>
```

Splitting the page up...

index.html

First, we're going to split the page up into sections using dividers or 'div's

```
<div>
  <h2><em>What do you need to create a website?</em></h2>
  <ul>
    ...
  </ul>
</div>
<br>
<p>
  Follow my progress
</p>
```

writing CSS & some basic Definitions.

index.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>First site</title>
    <link rel="stylesheet" type="text/css" href="css/styles.css">
  </head>
  <body>
    <h1>Hello world</h1>
  ...
```

body {

color: rgb(65,75,86);

}

styles.css

THE ANATOMY OF A CSS RULE SET

The element you
want to style

selector {

property: value;

}

What you are
styling

Declaration block

The effect value of
the style

THE ANATOMY OF A CSS RULE SET

The element you
want to style

selector {

property: value;

}

What you are
styling

Declaration block

**Don't forget these
semicolons!**

The effect value of
the style

writing CSS & some basic Definitions.

```
body {
```

```
    color: rgb(65,75,86);
```

```
    font-family: arial;
```

```
    margin: 0px;
```

```
}
```

styles.css

writing CSS & some basic Definitions.

```
body {
```

```
    color: rgb(49,58,69);
```

```
    font-family: arial;
```

```
}
```

```
div {
```

```
    min-height: 100vh;
```

```
    width: 100vw;
```

```
}
```

styles.css

writing CSS & some basic Definitions.

```
div {  
    min-height: 100vh;  
    Width: 100vw;  
}  
h1, h2 {  
    text-align: center;  
}  
h1 {  
    font-size: 5em;  
}
```

styles.css

writing CSS & some basic Definitions.

styles.css

```
h1 {  
    font-size: 5em  
}  
h2 {  
    font-size: 2.5em;  
}  
img {  
    display: flex;  
    padding: 20px;  
    margin: 0 auto;  
    border-radius: 130px;  
}
```

writing CSS & some basic Definitions.

```
img {
```

```
    ...
```

```
}
```

```
p {
```

```
    width: 40%;
```

```
    margin: 10vh 10vw;
```

```
    font-size: 2em;
```

```
}
```

```
a {
```

```
    color: rgb(0,0,0);
```

```
}
```

styles.css

writing CSS & some basic Definitions.

```
a {  
    color: rgb(0,0,0);  
}  
ol {  
    list-style: none;  
    padding: 0;  
}
```

styles.css

writing CSS & some basic Definitions.

```
ol {  
    list-style: none;  
    padding: 0;  
    margin-bottom: 10%;  
    font-size: 3em;  
}
```

styles.css

writing CSS & some basic Definitions.

```
ol {  
    list-style: none;  
    padding: 0;  
    margin-bottom: 10%;  
    font-size: 3em;  
    display: flex;  
    justify-content: space-evenly;  
}
```

styles.css

CH. 8: SELECTORS AND ATTRIBUTES

What if you want to style the `` elements differently?

```
<ul>  
  <li>HTML is cool</li>  
  <li>CSS is cooler</li>  
  <li>JS is the best</li>  
</ul>
```

RECAP ON ATTRIBUTES

`<tag attribute="value">`

No spaces on
either side of
the = sign

Quote marks
surrounding the
value of the
attribute

```
<div class="info-section">
```

```

```

```
<a href="http://google.com">
```

CH. 9: USING ID AND CLASS SELECTORS

ID

Unique: an ID can only be used on an HTML page

```
<h2 id="subtitle">Puddings</h2>
```

Class

It's not unique: the same class can be used on multiple items on an HTML page

```
<ul>
  <li class="item">A computer</li>
  <li class="item">A text editor</li>
  <li class="item">A web browser</li>
</ul>
```

ID and class selectors.

index.html

```
<ul>
  <li>HTML is cool</li>
  <li id="lower">CSS is cooler</li>
  <li id="lowest">JS is the best</li>
</ul>
```

styles.css

```
li#lower {
  margin-top: 7%;
}
li#lowest {
  margin-bottom: 14%;
}
```

USING ID SELECTORS IN CSS

```
<li id="lowest">HTML IS COOL</li>
```

```
#lowest { .. }
```

```
li#lowest { .. }
```

Both of these are valid

class selectors.

```
<div class="contrast">  
  <h2><em>What do you need to create a website?</em></h2>  
  <ul> ... </ul>  
</div>
```

index.html

```
div.contrast {  
  background-color: rgb(0,0,0);  
  color: rgb(255,255,255);  
}
```

styles.css

USING CLASS SELECTORS IN CSS

```
.item { .. }
```

```
li.item { .. }
```

```
ul .item { .. }
```

```
ul li.item { .. }
```

```
<ul>
```

```
  <li class="item">Gateau</li>
```

```
  <li class="item">Cake</li>
```

```
  <li class="item">Pie</li>
```

```
</ul>
```

All of these are valid

FINISHING UP

styles.css

```
div.contrast {  
    background-color: rgb(0,0,0);  
    color: rgb(255,255,255);  
    display: flex;  
    justify-content: center;  
    align-items: center;  
    flex-direction: column;  
}
```

Finishing up

```
div.contrast {
```

```
    ...
```

```
}
```

```
ul {
```

```
    list-style: none;
```

```
    display: flex
```

```
}
```

```
li {
```

```
    margin: 0 20px
```

```
}
```

styles.css

Getting snazzy...

```
li {  
    margin: 0 20px;  
    transition: 0.6s ease;  
}  
li:hover {  
    font-size: 1.2em;  
    transition: 0.6s ease;  
}
```

styles.css

HOMEWORK FOR WEEK 2

Finishing off

Task:

1. Finish off the code-along from this week, you can check the [solution here](#) (image of how the page should [look here](#))
2. Get to know your flexbox with [flexbox froggy!](#)
3. Read [this guide](#) and [this guide](#) on how to use GitHub and version control - **This is vital.**
4. Watch this more [in-depth video](#) about how the internet works, for Front-End Devs

HANDY HINTS - CSS SELECTORS

```
h1 { .. }
```

```
img { .. }
```

```
div { .. }
```

```
div h1 { .. }
```

```
ul { .. }
```

```
ul li { .. }
```

```
li { .. }
```

```
a { .. }
```

```
p a { .. }
```

HANDY HINTS - CSS PROPERTIES

```
h1 { font-family: .. }
```

```
img { width: .. }
```

```
div { height: .. }
```

```
div h1 { color: .. }
```

```
ul { list-style-type: .. }
```

```
ul li { padding-left: .. }
```

```
li { margin-bottom: .. }
```

```
a { text-decoration: .. }
```

```
p a { border: .. }
```

HANDY HINTS - CSS VALUES

```
h1 { font-family: 'Helvetica', sans-serif; }
```

```
img { width: 300px; }
```

```
div { height: 595px; }
```

```
div h1 { color: blue; }
```

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

```
ul li { padding-left: 30px; }
```

```
li { margin-bottom: 50px; }
```

```
a { text-decoration: underline; }
```

```
p a { border: 1px solid #000; }
```