

COMP4021
Internet Computing

Basic CSS

Gibson Lam and David Rossiter

CSS

- HTML elements all have their own predefined visual style, e.g. `...` changes text to bold and `<h1>...</h1>` changes text to big heading
- It will be fun if we can customise our own HTML elements
- CSS (**C**ascading **S**tyle **S**heets) is a language that allows you to do that



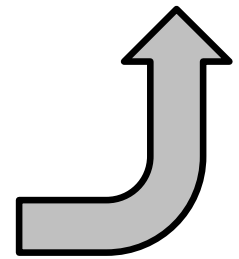
A Simple HTML Page

- The HTML below is a simple web page with a header, a paragraph and some bold text
- It looks boring so let's use CSS to change the *style* of the page!

Beautiful

I look **absolutely great!**

```
<body>  
  <h1>Beautiful</h1>  
  <p>  
    I look <b>absolutely great</b>!  
  </p>  
</body>
```



*Default visual style
shown in the browser*

Adding Style to the Page

- You can easily change the appearance of any HTML element by giving a style attribute to it
- For example, the header can be changed to red using the following style attribute:

```
<body>
```

```
  <h1 style="color: red">Beautiful</h1>
```

The style attribute

```
  ...
```

```
</body>
```

Beautiful

I look **absolutely great!**

CSS Properties

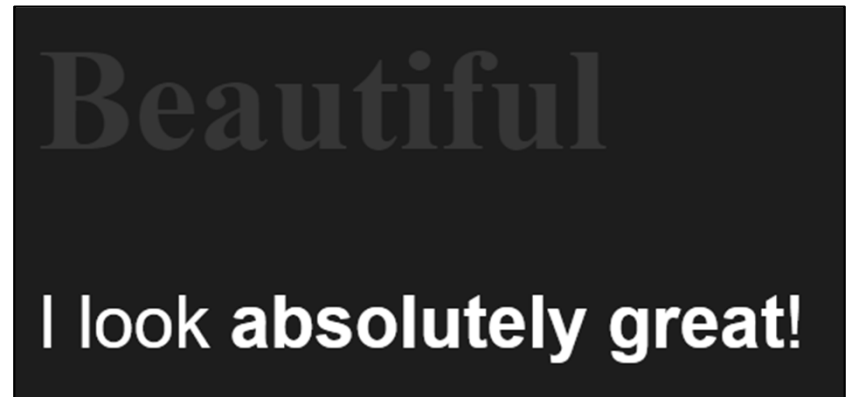
- The attribute content in the previous slide is called a *CSS property*
- In the example, the property name is 'color' and its value is 'red'

<u>color</u>	:	<u>red</u>
<i>Property name</i>		<i>Property value</i>

- You can also put multiple properties in a style attribute by separating them with semi-colons, see next slide

Using Many Style Attributes

- Instead of just the header, we make changes to almost every element on the page
- As you can see, there can be many different CSS properties that control different things



```
<body style="background: darkred">  
  <h1 style="color: red">Beautiful</h1>  
  <p style="font-family: Arial; color: white">  
    I look <b>absolutely great</b>!  
  </p>  
</body>
```

*Two properties in
one attribute!*

List of All CSS Properties

- There are lots of CSS properties
- You can find **ALL** properties here:

`https://www.w3.org/Style/CSS/all-properties.en.html`

- The page contains way too many properties that you will never use!
- The following slides give some examples of more commonly used CSS properties

Font Properties

- `font-family: Helvetica`
 - Other examples: "Times New Roman", Arial
- `font-size: 18px`
 - The font size can be expressed in points (pt), pixels (px), relative size to the element (em), document (rem) and so on
 - Other examples: 12pt, 1em
- `font-weight: bold`
 - Other examples: bolder, lighter, normal

Font Properties Example

```
<p style="font-family: Helvetica;  
        font-size: 18px;  
        font-weight: bold">
```

Big Big Text</p>

```
<p>normal text</p>
```

Big Big Text

normal text

*The use of
spacing is not
important in CSS*

Colour Properties

- `background-color: lightgray`
 - Other examples: `yellow`, `lightblue`
 - Alternatively, the name can be `background`, i.e. `background: lightgray`
- `color: darkgray`
 - This changes the text colour of the element
- Some colour names can be found here:

<https://htmlcolorcodes.com/color-names/>

The HTML Span Tag

- We have not discussed the span tag before
- `...` is useful because it doesn't have any default visual style
- E.g. `<p>In a paragraph In a span</p>` looks like this:

In a paragraph In a span

*No extra style
has been used
by the ``*

- So we can apply styles to `...` without worrying whether it already has a style which might mess up the result in some way

Colour Properties Example

```
<body style="background-color: skyblue">  
  <p>  
    <span style="color: red">R</span>  
    <span style="color: orange">A</span>  
    <span style="color: yellow">I</span>  
    <span style="color: green">N</span>  
    <span style="color: lime">B</span>  
    <span style="color: blue">O</span>  
    <span style="color: purple">W</span>  
  </p>  
</body>
```



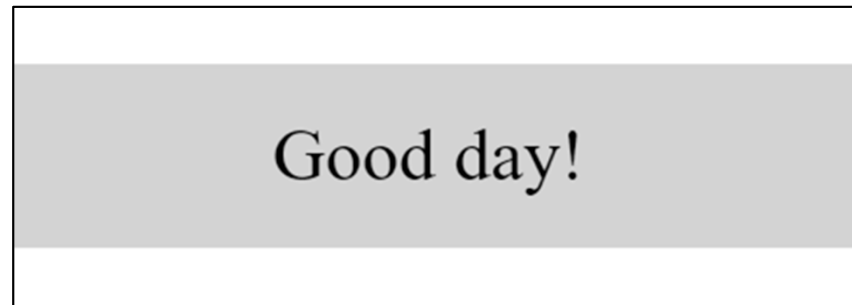
Paragraph Properties and Example

- `text-align: center`
 - Other examples: `left`, `right`, `justify`
- `line-height: 30px`
 - This is sometimes very useful in making your text occupying fixed vertical space

```
<p style="background: lightgray;  
        text-align: center;  
        line-height: 40px">
```

Good day!

```
</p>
```

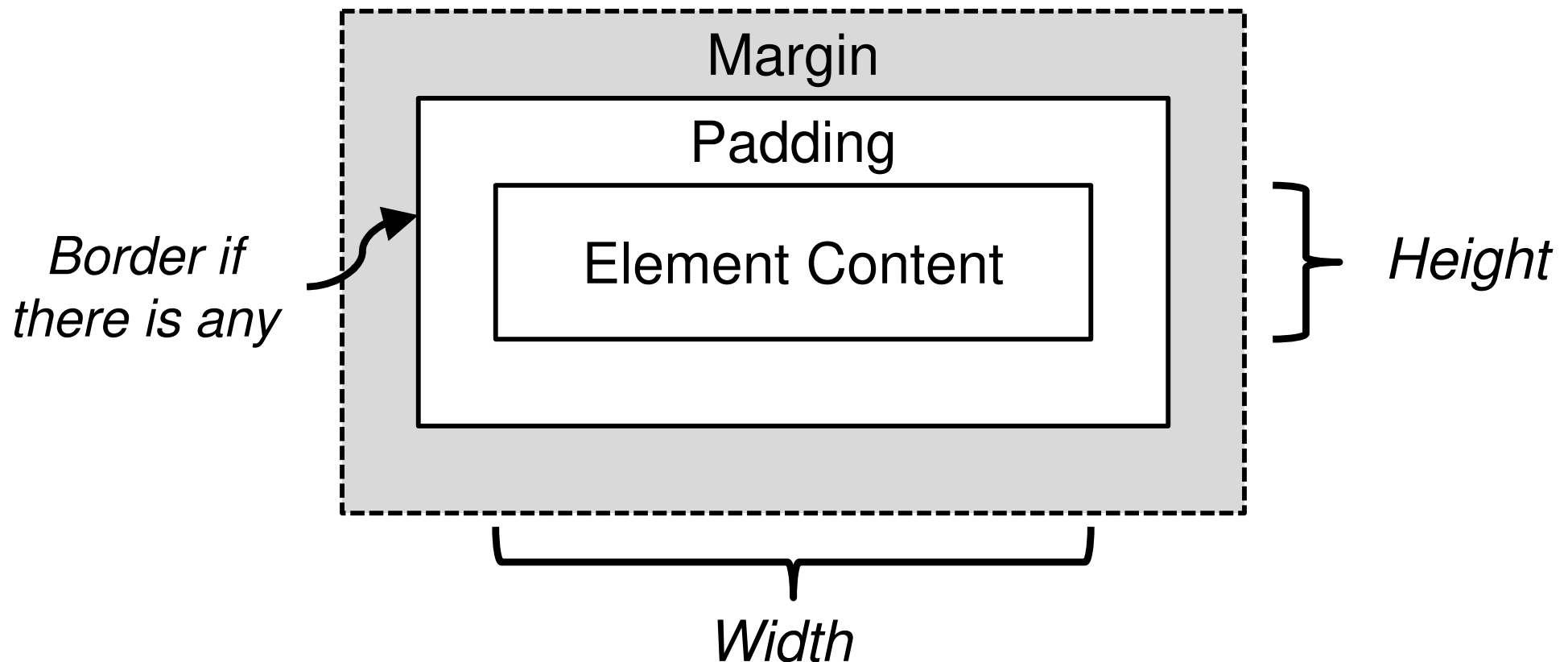


Inline and Block Elements

- There are generally two types of layout
- Inline elements
 - Elements that ‘flow’ with the page such as `` and ``
- Block elements
 - Elements that occupy a rectangular area such as `<h1>` and `<div>`
 - They start on a ‘new line’

Spacing of Block Elements

- CSS box properties control the width, height, padding and margin of block elements



Box Properties

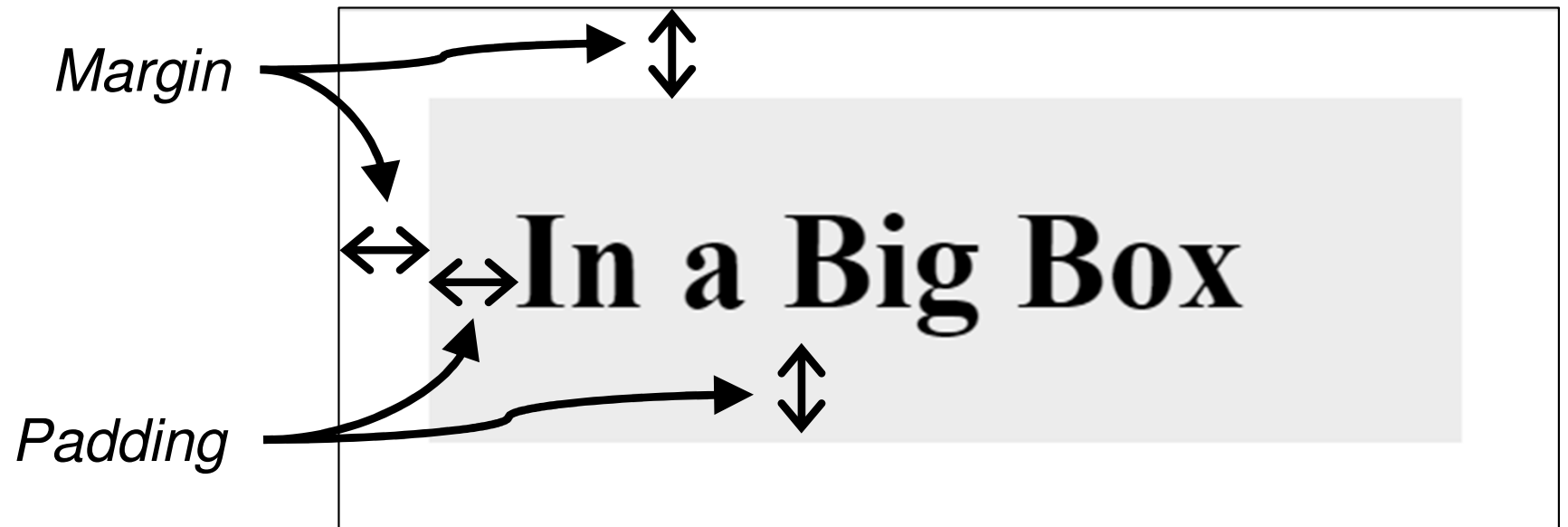
- `width: 300px`
`height: 200px`
 - You change the size of the element, excluding the margin and padding
- `margin: 20px`
 - Margin is the spacing outside the border (see later) of the element
- `padding: 10px`
 - Padding is the spacing between the content and the border of the element

Box Properties Example

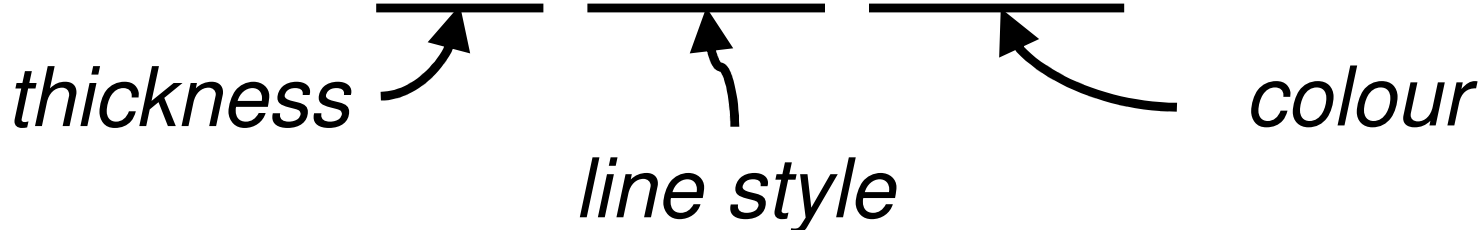
```
<h1 style="background: yellow;  
margin: 20px;  
padding: 20px;  
width: 200px; height: 40px">
```

In a Big Box

```
</h1>
```



Border Properties

- `border: 2px solid black`


thickness *line style* *colour*

 - Line style can be dotted, dashed, double and so on
- `border-radius: 10px`
 - You can make rounded corners around the element with a border or background

Border Properties Example

```
<h1 style="background: red;  
width: 180px;  
text-align: center;  
border: 2px solid black;  
border-radius: 10px">
```

Warning!
</h1>



Examining CSS

- It is useful to look at websites (your web site or another) and study their style using Chrome Inspector (*Right click > Inspect*)

The screenshot shows a web browser with a page titled "Undergraduate Programs". The page content includes a heading "Major Programs" and a paragraph describing the department's offerings. The Chrome DevTools Inspector is open, showing the HTML structure of the page. The element selected is an

tag with the text "Undergraduate Programs". The Inspector shows the HTML structure, including the tag and its parent container. The CSS panel on the right shows the styles applied to the selected element, including font-size, font-weight, and color. A bracket on the right side of the CSS panel is labeled "CSS Info".

h1 278.67 x 57.19

Undergraduate Programs

Major Programs

The department offers two major programs. COMP is our general undergraduate degree program. It provides a broad education in all core areas of Computer Science, while allowing students the flexibility to pursue individual interests in higher-level areas. Students can pursue double majors with COMP as one of the two majors; however, the different requirements of the two majors, including single-counted credits for each major, may be complicated to satisfy. COSC is a special program catered only to

```
<!DOCTYPE html>
<html lang="en">
  <head>...</head>
  <body>
    <main role="main">
      <section id="header" class="navbar navbar-d
        default navbar-static-top xrm-margin-bottom">
        ...</section>
      <section id="page-section" style="padding-t
        op:3px">
        <div class="header-img" style="background
          -image: url("/images/banners/004.jpg"); h
            eight: 180px;"></div>
        <div class="container">
          ::before
          <h1>Undergraduate Programs</h1> == $
          <div class="row">...</div>
          ::after
        </div>
        <div class="nav-menu-holder closed activ
          e">...</div>
        <div class="nav-menu-holder open">...</div>
      </section>
      <section id="footer">...</section>
      <section id="hkust-footer">...</section>
    </main>
    <script src="/js/vendor.js"></script>
    <script src="/is/app.is?&t=1628318278">
  </script>
</html>
```

Styles

Filter :hov .cls +

element.style { }

.header-img + .container > h1:first-of-type { margin-top: 0; }

h1, h2, h3, h4, h5, h6, .h1, .h2, .h3, .h4, .h5, .h6 { font-weight: 600; }

h1 (max-width: 767px) { font-size: 26px; }

h1 { font-size: 28px; }

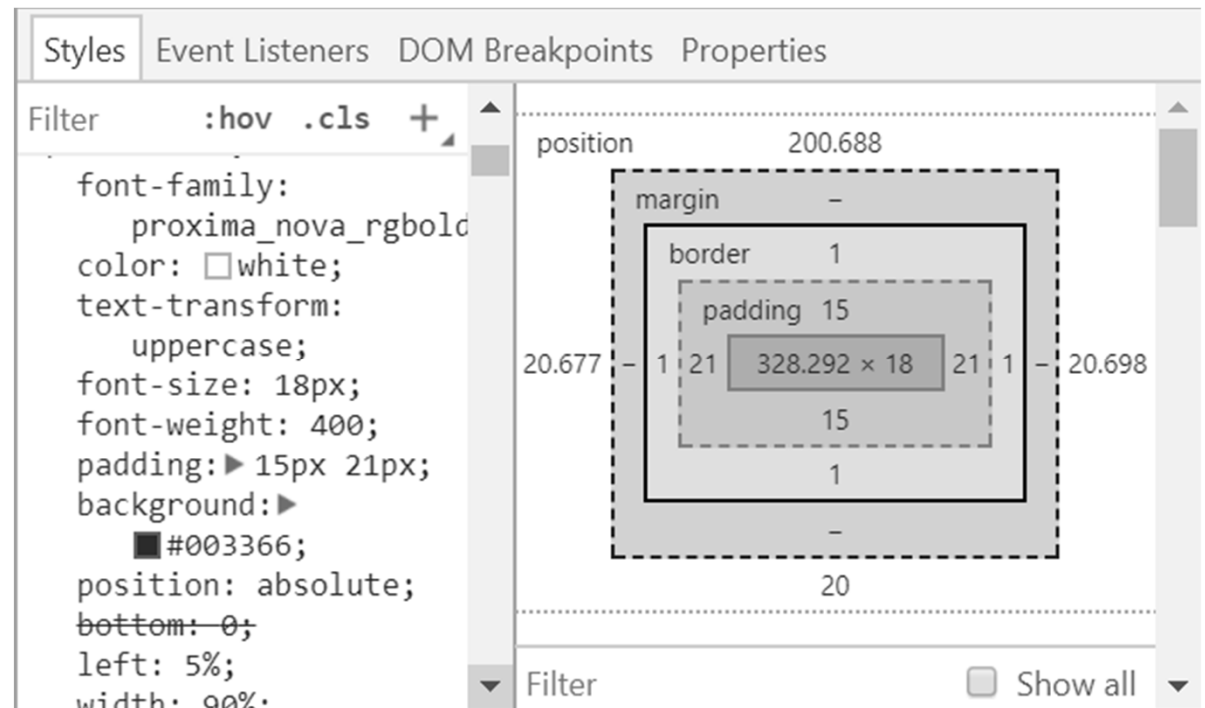
.h1, .h2, .h3, .h4, h1, h2, h3, h4 { font-weight: 500; }

.h1, .h2, .h3, .h4, .h5, .h6, h1, h2, h3, h4, h5, h6 { color: #164f71; }

CSS Info

The Chrome Inspector

- The Chrome inspector also helps you understand the box layout of each element
- This is very helpful for you to understand the box properties



Repeating the Same Properties

- So far we use the CSS properties separately on each individual element
- If you need to use the same properties on several elements, it will be very clumsy, e.g.:

```
<h1 style="color: darkblue">Name</h1>
```

```
<p>...</p>
```

```
<h1 style="color: darkblue">Job</h1>
```

```
<p>...</p>
```

```
<h1 style="color: darkblue">Hobbies</h1>
```

```
<p>...</p>
```

Using Style Sheets

- Luckily you don't need to repeat the same properties multiple times using a *style sheet*
- A style sheet is a collection of CSS properties like this:

```
<style>  
h1 { color: darkblue; }  
</style>
```

- It is written inside the `<style>` tag, consisting of style rules, see next slide

Style Rules

- Here is a style rule:

h1 { color: darkblue; }
Selector *CSS properties*

- The selector is the HTML elements that you want the rule to be applied to
- In this example, the rule is applied to **every** <h1>...</h1> on the page

A Style Sheet Example

*A style sheet
is put inside
the head
section*

```
<html>
<head>
  <title>Using Style Sheet</title>
  <style>
    h1 { color: darkblue; }
  </style>
</head>
<body>
  <h1>Name</h1>
  <p>...</p>
  <h1>Job</h1>
  <p>...</p>
  <h1>Hobbies</h1>
  <p>...</p>
</body>
</html>
```

Name

...

Job

...

Hobbies

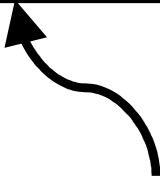
...

*All headers
are affected*

Linking to External Style Sheets

- If you want to, you can put the style sheet in a separate file and then link it back using the `<link>` tag, like this:

```
<html>
<head>
  <title>Using Style Sheet</title>
  <link href="mystyle.css" rel="stylesheet">
</head>
<body>
  ...
</body>
</html>
```



*The file containing the style sheet
in the same folder of the HTML file*

Using an External Style Sheet

```
<html>
<head>
  <title>Using Style Sheet</title>
  <link href="mystyle.css" rel="stylesheet">
</head>
<body>
  <h1>Name</h1>
  <p>...</p>
  <h1>Job</h1>
  <p>...</p>
  <h1>Hobbies</h1>
  <p>...</p>
</body>
</html>
```



`h1 { color: darkblue; }`

mystyle.css

We get the same visual styles as before

Name

...

Job

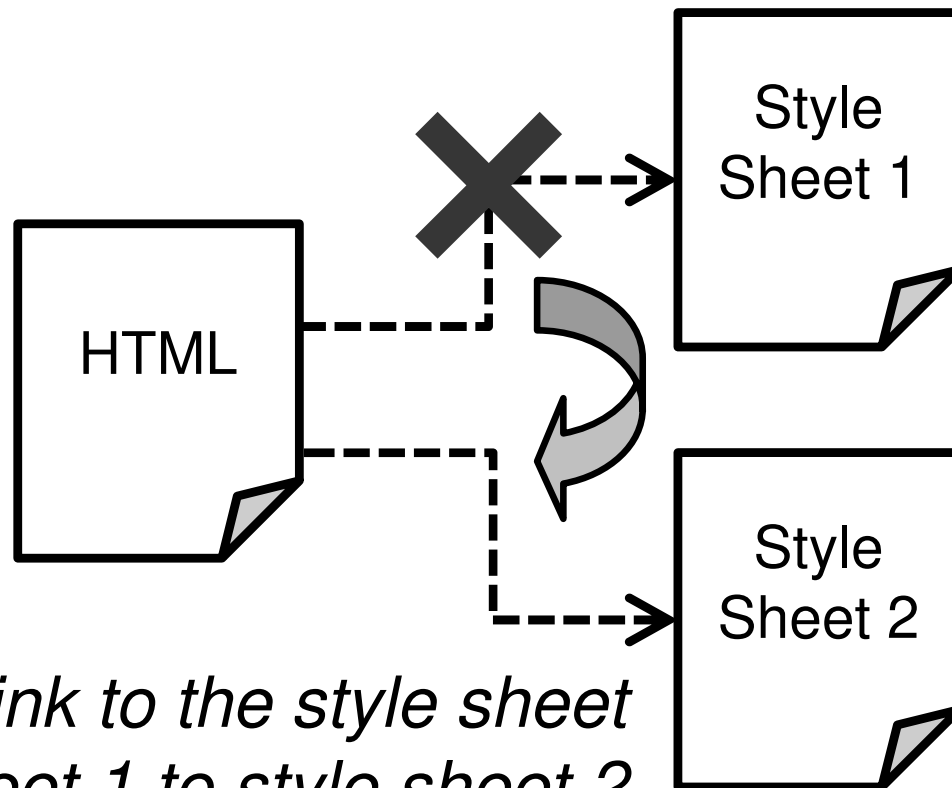
...

Hobbies

...

Using Different Style Sheets

- If you link a style sheet using an external file, you can switch the look and feel of your page ‘instantly’ by changing the style sheet file
- See the example on the next slide

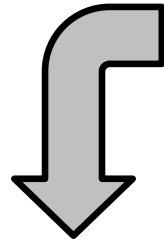


*Change the link to the style sheet
from style sheet 1 to style sheet 2*

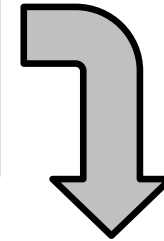
HTML Page

```
<h1>Beautiful</h1>
<p>
I look
<b>absolutely great</b>!
</p>
```

Use Style Sheet 1



Use Style Sheet 2



Style Sheet 1

```
body { font-family: Arial;
        background: deeppink; }
h1    { color: darkred; }
p     { color: white; }
```

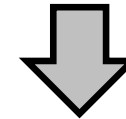


Beautiful

I look **absolutely great!**

Style Sheet 2

```
body { font-family: Broadway;
        background: blue; }
h1    { color: white; }
p     { color: black; }
```



Beautiful

I look **absolutely great!**

Class Rules and Attributes


- It will be clumsy if you want to apply the same CSS properties to different HTML tags, i.e.:

h1 { color: darkblue; }	}	<i>You want both h1 and h2 to be dark blue</i>
h2 { color: darkblue; }		

- Also, you may want to apply visual styles to only certain selected elements, e.g. not all <h1>
- One way to make the above simpler is to use class rules and attributes

Style Rules for Classes

- You make a class rule by putting a dot (.) in front of the name of the selector, like this:

You add a dot in front of the selector  `.important {
 color: red;
 font-weight: bold;
}`

- Then, your rule is only effective for the HTML tags having the selector's name (without the dot) in their class attribute, see next slide

Using the Class Attributes

```
<style>
.important {
  color: red;
  font-weight: bold;
}
</style>
...
```

HURRAY!

You have won \$1 millon !

*Elements with the class
'important' are affected by the rule*

```
<h3 class="important">HURRAY!</h3>
```

```
<p>
```

You have won

```
<span class="important">$1 millon</span>
!</p>
```


The id Attribute

- You can give a 'name' to any HTML using the id attribute

- Here is an example:

Both spans have their 'names'



```
<p>
```

```
  <span id="first-name">Gibson</span>
```

```
  <span id="last-name">Lam</span>
```

```
</p>
```

- Like the class rule, you can use a CSS rule to apply visual styles to an HTML element with a certain id, see next slide

Using the Id Selector

You add a # in front of the selector for the id attribute

```
<style>
```

```
#last-name {
```

```
  text-transform: uppercase;
```

```
  font-weight: bold;
```

```
}
```

```
</style>
```

```
...
```

```
<p>
```

```
  <span id="first-name">Gibson</span>
```

```
  <span id="last-name">Lam</span>
```

```
</p>
```

Change the text to uppercase

Gibson **LAM**

More Selectors Later

- There are plenty more selectors that we have not discussed here
- We will revisit selectors when we talk about the DOM and the use of jQuery, simply because using selectors is an essential part of jQuery programming