SENG1050

Internet Communications

Week 04 CSS

Lecture Plan

Weekly program (lectures)

Key concepts from last lecture

- The rules and principles of XML
  - Tag names are case sensitive
  - Tags can have as many attributes as you want
  - Root tag contains all other tags
  - Tags must be correctly nested
  - Tags must be closed
- XML Parser analyses and identifies different pieces of information
  - CDATA and PCDATA

Key concepts from last lecture

- XML Trees – the hierarchical structure of the XML document
  - Easier to understand the relationship among data pieces
- How XML Namespace can resolve name conflicts
  - Offers a standard for providing uniquely named elements and attributes
- How to define your own XML tags using DTD

Key concepts from last lecture

- The purpose of a DTD is to define the legal building blocks of an XML document.
  - Internal and External DTD
- How to declare DTD elements
  - `<!ELEMENT tag (children)>`
- How to set the attributes for DTD elements
  - `<!ATTLIST tag attribute type default>`
- What is a DTD entity: internal, external and parameter
  - `<!ENTITY name "value">`

Week 04 Lecture 01 Outline

CSS

- ❑ What are Cascading Style Sheet
- ❑ Advantages of CSS
- ❑ CSS Syntax
- ❑ CSS Levels
- ❑ CSS Cascading order
- ❑ CSS Selectors
- ❑ Common CSS properties and values
- ❑ RGB Colour Scheme
- ❑ Background, margin, boarder
- ❑ CSS Inheritance
- ❑ Image

Visual Formatting

- Tags imply suggestions to the browser about how to draw their content
  - Headings change the font size and weight
  - `<blockquote>` indents the margins
- These are only suggestions!
  - A browser can ignore them
 - It may not even support them – e.g., some older or simpler mobile phones

---
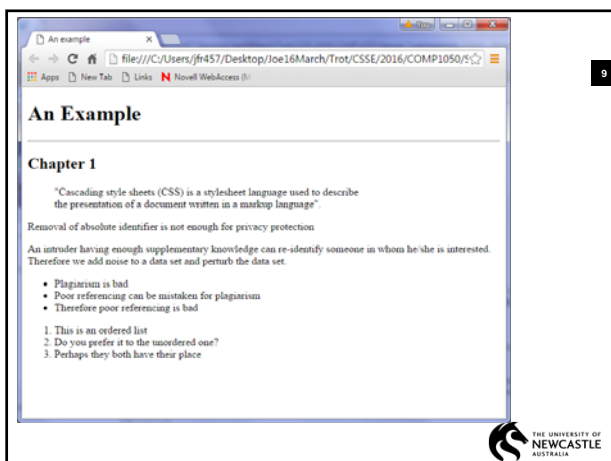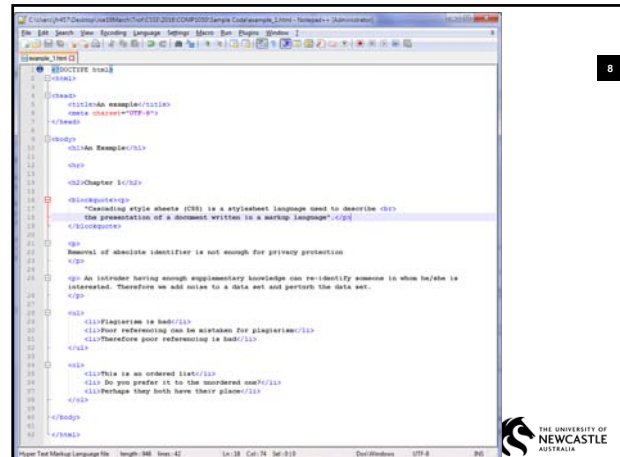
SEMANTICS!

- All tags, except a few such as `<div>` and `<span>`, have implicit semantic meanings
  - For example: <h1>, <p>, <strong>, <em>
  - Before using a tag, think about whether it is really what you "mean" in the given context
- Never use a tag just because it gives you the desired visual formatting!!
  - It won't always work – especially on every platform
  - There are much better ways to do it

Cascading Style Sheets (CSS)

- Style sheets are a way of adding visual formatting to HTML tags
  - CSS is the most common style sheet language
  - CSS is independent of the semantics of the HTML tags – you can change the (suggested) formatting without affecting the semantics!
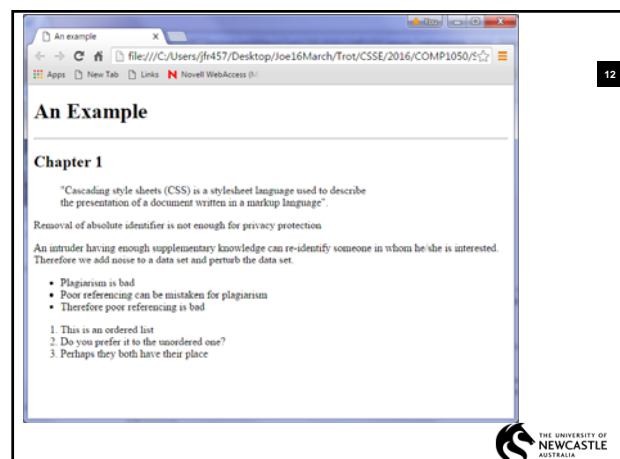
Slide 13

```
/* styletest1.css
 * Purpose: To introduce a CSS file
 * Date 16/08/16
 */

body {background-image: url('images.jpg');
         margin-left: 1cm
         }

h1
{ color:red;
text-align:center}

h2
{ color:blue;
text-align:left;
}

p
{ text-align:left;
color:black;
font-family:arial, serif;
text-indent:+2%
}
```
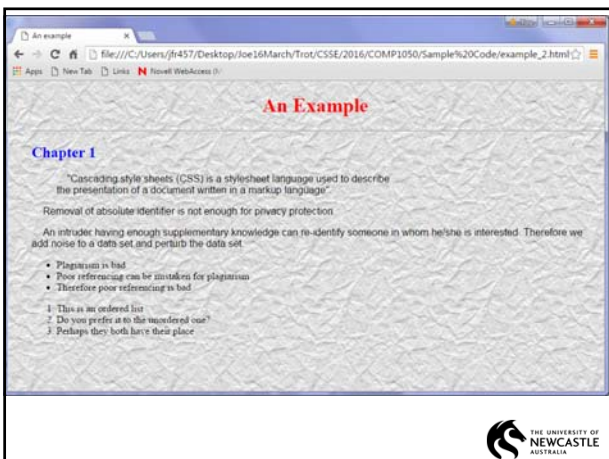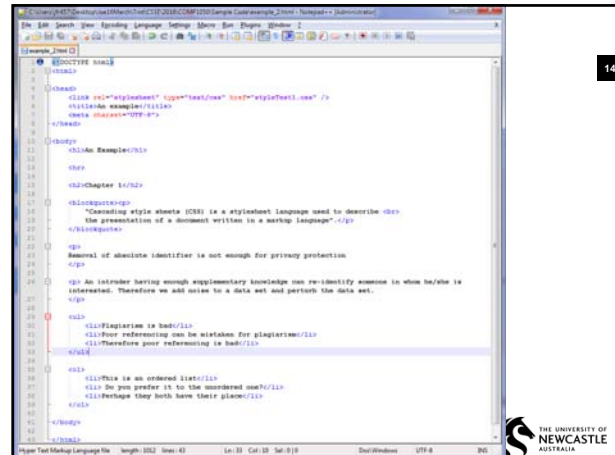
Slide 14

Slide 15

Slide 16

CSS - Advantages

- Give greater control over page layout and typography
- Style is separate from structure
- Style can be stored separately
  - Saves work if we want to change many pages
  - Potentially smaller documents
  - Site maintenance is easier

Slide 17

Cascading Style Sheets

- **CSS1** introduced styles for the following document features:
  - Fonts
  - Text
  - Colour
  - Backgrounds
  - Block-level Elements

Slide 18

Cascading Style Sheets

- **CSS2** introduced styles for the following document features:
  - Positioning
  - Visual Formatting
  - Media Types
  - Interfaces
- **CSS 2.1** did not add any new features to the language

Cascading Style Sheets

- **CSS3** introduces styles for the following document features:
  - User Interfaces
  - Accessibility
  - Columnar layout
  - International Features
  - Mobile Devices
  - Scalable Vector Graphics

CSS - Syntax

- Style sheets are composed of rules that describe the styling to be applied
- Rules have the following syntax:

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

- The following CSS rule would set the colour of the text to blue

```
body {color: blue}
```

```
/* styletest1.css
 * Purpose: To introduce a CSS file
 * Date 16/08/16
 */

body {background-image: url('images.jpg'),
        margin-left: 1cm
        }

h1
{ color:red,
text-align:center)

h2
{ color:blue;
text-align:left;
}

p
{ text-align:left;
color:black;
font-family:arial, serif;
text-indent:+2%
}
```

CSS - Levels

- CSS can be applied to HTML in three main ways…

  1. Inline styles
  2. Embedded (Document-level) styles
  3. External style files

CSS - Levels

1. Inline styles
   - Almost every tag in HTML has an optional **style** attribute

   - **style="**$property_1$: $value_1$; … $property_n$: $value_n$**"**

   - The style applies to the content of the tag … including nested tags

---

CSS - Levels

Examples:
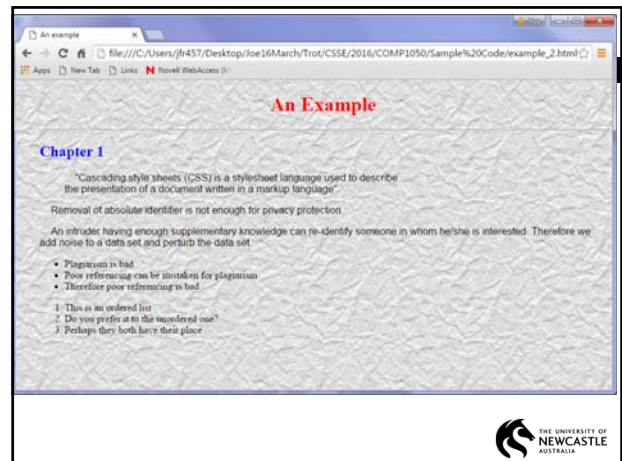
- `<h1 style="color:red">Little Red Riding Hood</h1>`

- `<h1 style="text-align:right">Teach Me Chinese</h1>`

- `<body style="background-color:green">`

  `<h1 style="text-align:center">It Ain't Easy…</h1>`

  `<img src="kermit.png" alt="Kermit the Frog style="width:150px; height:200px; float:right">`

CSS - Levels

28

2. Embedded styles
- `<head>`
  `<style type="text/css">`
  $selector$ { $property_1$: $value_1$; … }
  …
  `</style>`
  `</head>`
- *selector* determines which tags the style applies to
- Applies to all such tags in the current document…

CSS - Levels

29

Example:
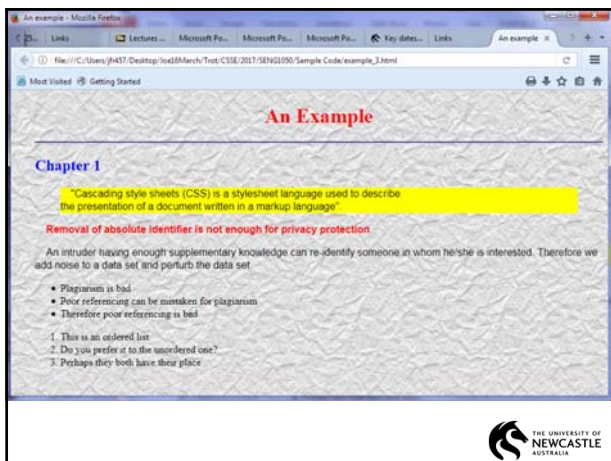
```
<style type="text/css">
  body { color: white; background-color: green;
         text-align: left }
  h1 { color: yellow; text-align: center }
  h2 { font-style: italic }
  strong { font-size: larger; color: red }
  img { float: right }
  img#special { width: 150px; height: 200px }
  #green {color:green}
</style>
```

CSS - Levels

3. External style files
- `<head>`
  `<link rel="stylesheet"`
    `type="text/css" href="uri" />`
  `</head>`
- Applies the CSS rules found in the external style file *uri* to this document
- The style file should contain only rules of the form *selector* { *property₁*: *value₁*; … } – no `<style>` or `<head>` or any other XHTML tags
- Comments in the style file have the form `/* … */`

CSS – Cascading Order

- Generally speaking we can say that all the styles will "cascade" into a new "virtual" style sheet by the following rules, where the top has the highest priority:

CSS – Cascading Order

- Browser default style rules are overridden by rules in an external style file
- Style rules in an external style file are overridden by rules in a document-level style tag
- Rules in a document-level style tag are overridden by rules in an in-line style attribute
- So inline style rules have the highest priority
- Except for user defined instructions to the browser

CSS - Drawbacks

- CSS is only a suggestion to the browser – it might be ignored
- Not every property applies to every tag
- Not every browser supports every property, or every value for a property
- Users can override any or all CSS rules at the browser level

CSS - Selectors

$h1$ { $color$: $blue$; $font-size$: $12px$; }

Selector   Property   Value   Property   Value

selector { *property₁*: *value₁*; *property₂*: *value₂*; … }

Slide 37

CSS - Selectors　`37`

- *selector* can be –
  - A tag name **[Element Selector]**
    ```
    p { text-align: justify }
    ```
  - A tag descended from another tag
    ```
    div p { font-weight: bold }
    ```
  - A tag that is an immediate child of another tag
    ```
    div > p { background-color: yellow }
    ```
  - A tag that immediately follows another tag
    ```
    div + p { background-color: yellow }
    ```
  - A tag that follows another tag
    ```
    div ~ p { background-color: yellow }
    ```

THE UNIVERSITY OF NEWCASTLE AUSTRALIA

Slide 38

CSS - Selectors　`38`

- *selector* can be…
  - A hyperlink that has not been visited yet
    ```
    a:link { color: blue }
    ```
  - A hyperlink that has already been visited
    ```
    a:visited { color: black }
    ```
  - A tag during a user action
    ```
    a:active { color: red }
    em:hover { font-size: 120% }
    ```

THE UNIVERSITY OF NEWCASTLE AUSTRALIA

Slide 39

`39`

```
/* styletest1.css
 * Purpose: To introduce a css file
 * Date 16/08/16
 */

body {background-image: url('images.jpg');
            margin-left: 1cm
         }

h1
{    color:red;
text-align:center}

h2
{    color:blue;
text-align:left;
}

p
{    text-align:left;
color:black;
font-family:arial, serif;
text-indent:+2%
}

address {text-align:right}

a:link {color:blue}
a:visited {color:green}
a:active {color:red}
a:hover {font-size: 120% color:purple}
a[href="http://www.newcastle.edu.au/profile/joe-ryan"]
```

THE UNIVERSITY OF NEWCASTLE AUSTRALIA

Slide 40

THE UNIVERSITY OF NEWCASTLE AUSTRALIA

Slide 41

THE UNIVERSITY OF NEWCASTLE AUSTRALIA

Slide 42

THE UNIVERSITY OF NEWCASTLE AUSTRALIA

CSS - Selectors

`43`

- *selector* can be…
  - A tag with a given attribute
    `input[checked] { color: red }`
  - A tag with a given attribute with a given value
    `a[href="home.html"] { color: green }`
  - A tag with a given **id** attribute value **[ID Selector]**
    `#copyright { font-style: italic }`
  - A comma-separated list of selectors
    `h1, h2, h3 { color: white }`

The `class` Attribute

`44`

- There is a special attribute for use when formatting HTML using CSS – **class**
  - It contains a comma-separated list of "class names"
  - Classes can be used in CSS selectors…
    `<div class="copyright">…</div>` with
    `div.copyright { font-style: italic }`
  - The class attribute is also very useful for associating extra semantic meaning with XHTML tags
    `p.intro, p.body, p.conclusion { … }`

`45`

```
/* styletest1.css
 * Purpose: To introduce a CSS file
 * Date 16/08/16
 */

body {background-image: url('images.jpg');
      margin-left: 1cm
}

h1
{   color:red;
text-align:center;
}
h2  color:blue;
text-align:left;
}

p.right
text-transform:uppercase;
text-align:right}

p
{   text-align:left;
color:black;
font-family:arial, serif;
text-indent:+2%
}

address {text-align:right}

a:link {color:blue}
a:visited {color:green}
a:active {color:red}
a:hover {font-size: 120%; color:purple}
a[href="http://www.newcastle.edu.au/profile/joe-ryan"]
```

`45`

`46`

`46`

`47`

`<span>` and `<div>`

`48`

- Most HTML tags have implicit visual formatting
  - CSS lets you override these, but you can get caught if a particular browser uses an unusual property – e.g., Lynx underlines `<em>` text; if you don't set `text-decoration` in the CSS style, the underline will not be overridden
  - Only `<div>` and `<span>` are guaranteed not to carry implicit visual formatting, so you should use these if another tag is not required for its semantics
  - You can add semantics to `<div>` and `<span>` using classes

CSS - Properties and Values `49`

- Here are some of the most common CSS properties and their values
  - There are many, many more
- For the gory details, see http://www.w3.org/TR/CSS2
- Tutorials
  - http://www.w3schools.com/css/css_intro.asp
  - https://developer.mozilla.org/en-US/docs/Web/Guide/CSS/Getting_started

---

CSS - Properties and Values `50`

- **font-style:**
  - **normal | italic | oblique**

  ```
  p.normal {font-style:normal}
  p.italic {font-style:italic}
  p.oblique {font-style:oblique}
  ```

- **font-weight:**
  - **normal | bold | 100 | … | 900**
  - **lighter | bolder**

  ```
  p.normal {font-weight:normal;}
  p.thick {font-weight:bold;}
  p.thicker {font-weight:900;}
  ```

- **font-variant:**
  - **normal | small-caps**

  ```
  p.small
  {
  font-variant:small-caps;
  }
  ```

CSS - Properties and Values `51`

- **font-family:**
  - A comma-separated list of fonts, in order of preference
  - Names with spaces must be in double- or single-quotes
  - You should include at least one of these: **serif**, **sans-serif**, **cursive**, **fantasy**, **monospace**
  - http://en.wikipedia.org/wiki/Font_family_%28HTML%29

  ```
  p
  {
  font-family:"Times New Roman",Georgia,Serif;
  }
  ```

CSS - Properties and Values `52`

- **font-size:**
  - Absolute: **xx-small | x-small | small | medium | large | x-large | xx-large**
  - Relative: **smaller | larger**
  - Length: a number followed immediately by units **cm | mm | in | pt | pc | em | ex | px**
  - Percentage: relative to the enclosing tag

  ```
  h1 {font-size:250%}
  h2 {font-size:200%}
  p {font-size:100%}
  ```

`53`

```
/* styletest1.css
 * Purpose: To introduce a CSS file
 * Date 18/08/16
 */

body {background-image: url('images.jpg');
        margin-left: 1cm
        }

h1
{   color:red;
text-align:center}

h2
{   color:blue;
text-align:left;
font-variant:small-caps}

p.right
{text-transform:uppercase;
 text-align:right}

p.large
{font-size:120%
font-family:courier}

p
{   text-align:left;
color:black;
font-family:arial, serif;
text-indent:+2%
}

address {text-align:right}

a:link {color:blue}
a:visited {color:green}
a:active {color:red}
a:hover {font-size: 120%; color:purple}
a[href="http://www.newcastle.edu.au/profile/joe-ryan"]
```
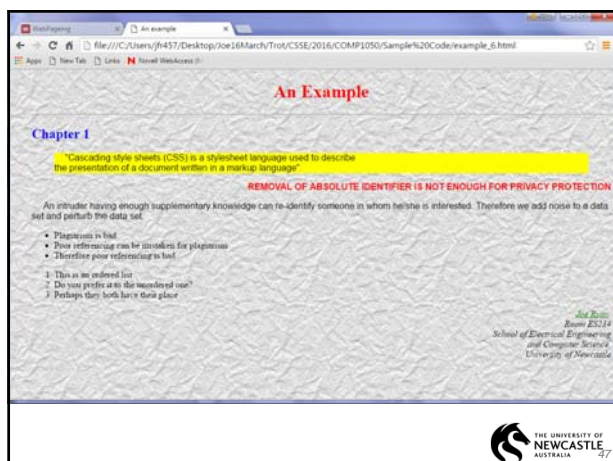

Slide 55
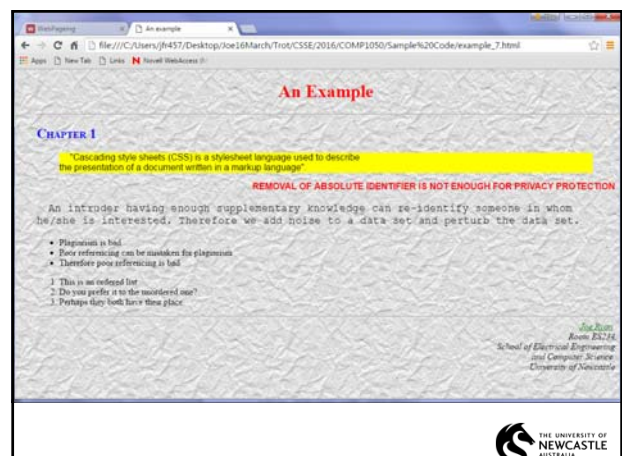
CSS - Properties and Values `55`

- **text-decoration:**
  - **none | overline | underline | line-through**
- **text-align:**
  - Horizontal alignment of text
  - **left | right | center | justify**
- **text-indent:**
  - Indent (**+** or **−**) for the first line
  - A length or percentage
- **text-shadow (CSS3):**   **Text-shadow effect!**
  - Specify the horizontal shadow, the blur distance, and color of the shadow
  - *h-shadow v-shadow* blur-color | none | initial | inherit;
  - h1 {text-shadow:*2px 2px* #FF0000;}

Slide 56

CSS - Properties and Values `56`

- **text-transform:**
  - **capitalize | uppercase | lowercase | none | inherit**
- **white-space:**
  - **normal | pre | nowrap**

white-space: nowrap

```
<p>I have      multiple
spaces and a line break</p>
```

WILL LOOK LIKE
↓

I have multiple spaces and a line break

(multiple spaces ignored, and line breaks will never occur, even if space is limited)

Slide 57

CSS - Properties and Values `57`

- **list-style-type:**
  - change the behaviour of list markers
  - **disc | square | circle** – unordered markers
  - **decimal | lower-roman | upper-roman | lower-alpha | upper-alpha** | …
- **list-style-image: url(*uri*)**

Slide 58

`58`
```
/* styletest1.css
 * Purpose: To introduce a CSS file
 * Date 16/08/16
 */
body {background-image: url('images.jpg');
        margin-left: 1cm
        }
h1
{   color:red;
text-align:center}
h2
{   color:blue;
text-align:left;
font-variant:small-caps}

p.right
{text-transform:uppercase;
 text-align:right}

p.large
{font-size:120%;
 font-family:courier}

p
{   text-align:left;
color:black;
font-family:arial, serif;
text-indent:+2%
}
address {text-align:right}
ul {list-style-type: square}
ol {list-style-type: lower-roman}

a:link {color:blue}
a:visited {color:green}
a:active {color:red}
a:hover {font-size: 120%; color:purple}
a[href="http://www.newcastle.edu.au/profile/joe-ryan"]
```
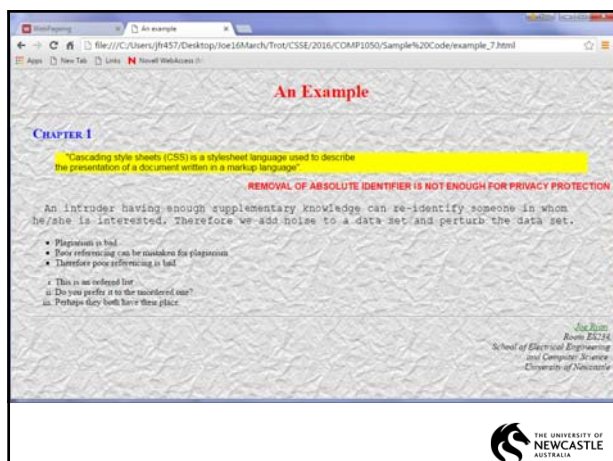
Slide 59

## Slide 60

CSS - Properties and Values – colour `60`

- Any color can be thought of as a combination of three primary colors: **red**, **green**, and **blue**
- By varying the intensity of each primary color, you can create almost any color and any shade of color
- This principle allows a computer monitor to combine pixels of red, green, and blue to create the array of colors you see on your screen

CSS - Properties and Values – colour
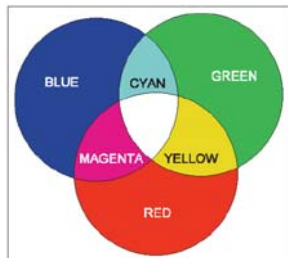
CSS - Properties and Values – colour

- Software programs, such as your Web browser, define color mathematically
- The intensity of each of three colors (RGB) is assigned a number from 0 (absence of color) to 255 (highest intensity)
- In this way, 256^3, or more than 16.7 million, distinct colors can be defined
- Each color is represented by a triplet of numbers, called an **RGB triplet**, based on the strength of its **R**ed, **G**reen, and **B**lue components

---

CSS - Properties and Values – colour

- HTML color values be entered as hexadecimals
- A **hexadecimal** is a number based on a base-16 numbering system rather than a base-10 numbering system that we use every day
  - Base 10 counting uses a combination of 10 characters (0 through 9) to represent numerical values
  - Hexadecimals include six extra characters: A (for 10), B (for 11), C (for 12), D (for 13), E (for 14), and F (for 15)

CSS - Properties and Values – colour

- To represent a number in hexadecimal terms, you convert the value to multiples of 16 plus a remainder.
- Once you know the RGB triplet of a color, the color needs to be converted to the hexadecimal format

---

CSS - Properties and Values – colour

- Using the basic color names allows you to accurately display them across different browsers and operating systems
- The list of only 17 colors is limiting to Web designers

CSS - Properties and Values – colour

Partial list of extended color names

Sample	Name	RGB	Hexadecimal
	aliceblue	(240,248,255)	#F0F8FF
	antiquewhite	(250,235,215)	#FAEBD7
	aqua	(0,255,255)	#00FFFF
	aquamarine	(127,255,212)	#7FFFD4
	azure	(240,255,255)	#F0FFFF
	beige	(245,245,220)	#F5F5DC
	bisque	(255,228,196)	#FFE4C4
	black	(0,0,0)	#000000
	blanchedalmond	(255,235,205)	#FFEBCD
	blue	(0,0,255)	#0000FF
	blueviolet	(138,43,226)	#8A2BE2
	brown	(165,42,42)	#A52A2A
	burlywood	(222,184,135)	#DEB887

11

CSS - Properties and Values

- **background-image:**
  - **url(*uri*)** | **none**
- **background-repeat:**
  - **repeat** | **repeat-x** | **repeat-y** | **no-repeat**
- **background-attachment:**
  - **scroll** | **fixed**
- **background-position:**
  - Has two values…
  - **left** | **center** | **right** | *length* | *percent*
  - **top** | **center** | **bottom** | *length* | *percent*
- **background-size (CSS3):**
  - **x y**
- **background-origin (CSS3):**
  - **Padding-box** | **border-box** | **content-box**

THE UNIVERSITY OF NEWCASTLE AUSTRALIA

```
/* styletest1.css
 * Purpose: To introduce a CSS file
 * Date 16/08/16
 */

body {background-image: url('images.jpg');
      margin-left: 1cm;
      background-repeat:repeat-y}

h1
{   color:red;
text-align:center}

h2
{   color:blue;
text-align:left;
font-variant:small-caps}

blockquote {background-color:yellow}
hr {color:navy}
hr.light {color:yellow}

p.right
{text-transform:uppercase;
 text-align:right}

p.large
{font-size:120%;
 font-family:courier}
```

THE UNIVERSITY OF NEWCASTLE AUSTRALIA

CSS - Properties and Values

- **margin-left** | **margin-right** | **margin-top** | **margin-bottom:**
  - The amount of space **outside** the "border" of the tag
  - Always transparent
  - Shorthand – **margin:** *top right bottom left*
  - Shorthand – **margin:** *top right bottom*
  - Shorthand – **margin:** *top right*
- **padding-left** | **padding-right** | **padding-top** | **padding-bottom** | **padding:**
  - The amount of space around the content of the tag, **inside** the "border"
  - Has the same background as the content

THE UNIVERSITY OF NEWCASTLE AUSTRALIA

CSS - Properties and Values

- `border-left-width` | `border-right-width` | `border-top-width` | `border-bottom-width` | `border-width:`
  - The border itself
  - `thin` | `medium` | `thick` | *length*
- `border-color:`
  - If different from the text `color`
- `border-style:`
  - `none` | `dotted` | `dashed` | `solid` | `double` | `groove` | `ridge` | `inset` | `outset`

CSS - Properties and Values

- `border-radius`
  - Setting all the four border
  - `1-4 length` | `%` | `initial` | `inherit`
- `Box-shadow:`
  - Attaches one or more drop-shadows to the box
  - `None` | `h-shadow v-shadow blur spread color` | `inset` | `initial` | `inherit;`
- `border-image:`
  - `Source slice width outset repeat` | `initial` | `inherit`

CSS - Properties and Values

13
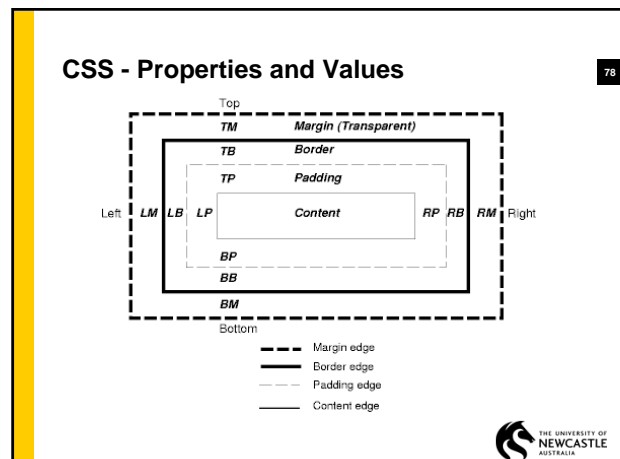
CSS - Properties and Values

CSS - Images

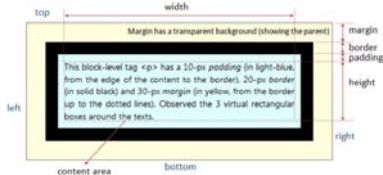- The default width and height of an image can be overridden
  - **width:** *length* **| auto**
  - **height:** *length* **| auto**
  - The image is "scaled" to the new size
  - If only one is specified (the other is auto) then the aspect ratio is maintained

```
<body>
    <div style="text-align: center">
        <img src="salogo.gif" alt="Sunny Acres" style="width:599;  height:223">
        <a href="home.htm"><img src="scarecrow.gif" alt="animated GIF"
                        style="width:500; height:300; border-width:0"></a>
    </div>
</body>
```

CSS - Images

- Loading a large image then scaling it to a small icon is very inefficient
  - Generate the small image offline
  - Embed the small image in the document with **<img>**
  - Hyperlink the small image to the large image file by wrapping **<a>** around **<img>**
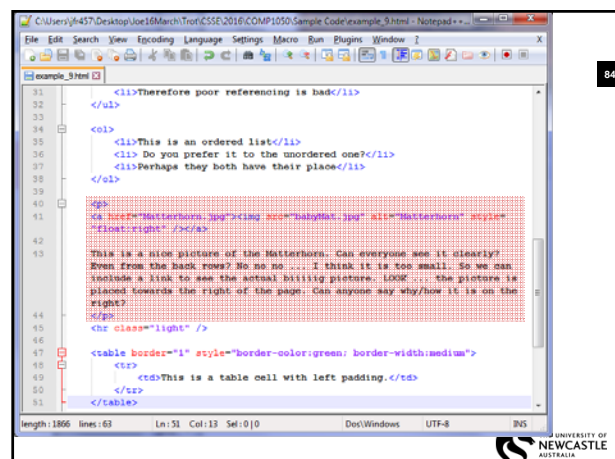 - The large image is then only loaded if needed

CSS - Images

- **float:**
  - Where to position the image relative to text
  - **left** – align with the left edge of the enclosing box and text flows around to the right
  - **right** – align with the right edge of the enclosing box and text flows around to the left
  - **none** – place inline, or in a new block with text above and below
  - Works for other content as well

Floating an Element

original layout

element is floated on the right margin and the subsequent page content wraps around it
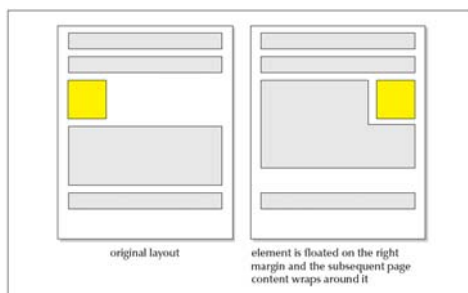
14
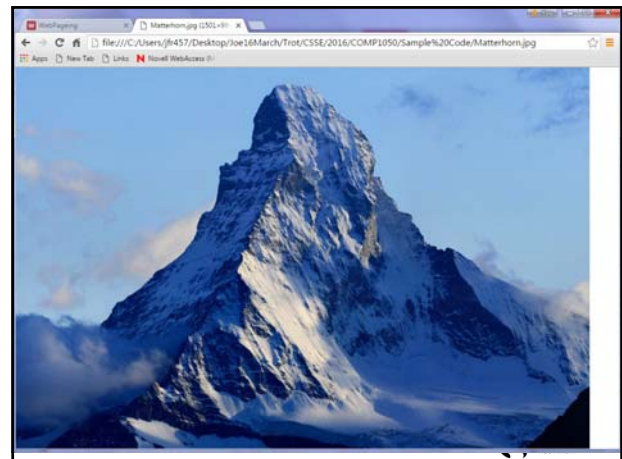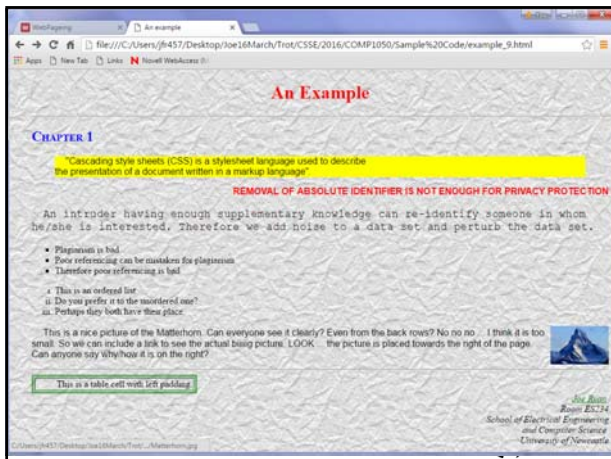
Adding Thumbnails

87
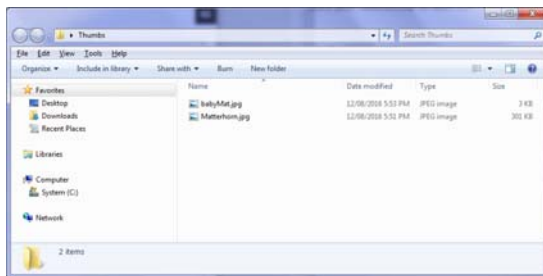

babyMat.jpg thumbnail courtesy of http://www.makeathumbnail.com/

Using Cascading Style Sheets

88

- Place style rules that apply to your **entire Web site** in an external style file and link all documents to it

- Place styles rules that apply only to **a group of documents** in an external style file and link only those documents to it (you can have many `<link>` tags)

- Place styles particular to **a given document** in a document-level style tag

Using Cascading Style Sheets

89

- Only use in-line styles for **one-off visual formatting** if you are sure it is not going to be re-used – even then, a document-level rules is probably better

- Use class attributes to distinguish tags with different semantic meanings – even if the visual formatting is the same

CSS Validation

90

- Mistakes in CSS code can be hard to pickup
  - Many browsers will stop processing a CSS rule if an error is discovered
  - Some will stop processing an entire style (document-level tag or external link)
- Or your CSS might be correct, but the browser might not support the property or value
- Use a CSS validator
  - http://jigsaw.w3.org/css-validator/
  - May fail if the HTML is invalid

Authoring Tools

91

- Positives
  - Good for beginners
  - Good for quick prototyping
- Negatives
  - Rarely generate valid HTML
  - Sometimes alter files as they open them
  - Often have trouble opening files not generated by the same tool
  - Almost never consider the semantics of the tags

Authoring Tools

92

- Your knowledge of HTML and CSS will last a lot longer and be more useful than knowledge of any authoring tool

- **Authoring tools for Assignments are NOT allowed!**

Summary

93

- How different CSS selectors can be used
  - Element Selector
  - ID Selector
  - Attribute Selector
  - Class Selector
- Common CSS properties and values
  - Font
  - Text
  - Background
  - Margin
  - Border
- RGB Color scheme
  - RGB

Summary

94

- How CSS can separate semantics and visual formatting
- CSS facilitate smart, effective and efficient management, of sites
- Syntax of CSS
- Three levels of CSS: inline, embedded and external
- Cascading order of CSS

COMP1050/6050 - Semester 2 - 2016 www.newcastle.edu.au

References

95

- **Web Development and Design Foundations with HTML5 (6e)**
  - *By Terry Felke-Morris*
    - Chapter 3

- http://www.w3schools.com/css/
- http://www.w3.org/TR/CSS2/selector.html