Week 4

COMP 1000, 17s1

Aarthi Natarajan

Outline

- Introduction to the Internet
- Structure of an HMTL page
- Web page construction tools
- Basic HTML elements
- CSS

Web Page Construction

Material adapted from the COMP1000 reference book
"Web 101" by Wendy Lehnert & Richard Kopec,
New Perspectives on HTML, XHTML and XML - Comprehensive, 3rd Edition by Carey
and slides by Bill Wilson

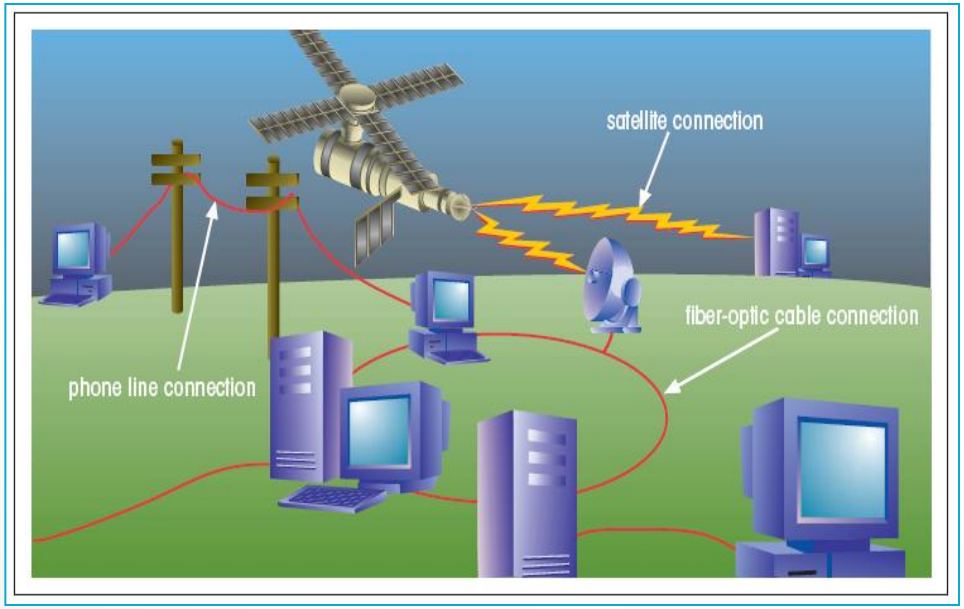
Resources

Textbooks:

- "Web 101" by Wendy Lehnert & Richard Kopec
- “New Perspectives in HTML, XHTML and XML”, 3rd ed., Cengage Learning by Carey, Patrick (2010).
- Web Sites:
  - World Wide Web Consortium (“W3C”):
    - http://www.w3.org
  - W3Schools:
    - http://www.w3schools.com
      - HTML tutorial: http://www.w3schools.com/html/
      - CSS tutorial:http://www.w3schools.com/css/

Internet

- a global network of connected computers, governed by a system of standards and rules.
- purpose of connecting computers is to share information – through a variety of standardised rules and methods known as **protocols** e.g., email, FTP
- uses fiber-optic cables, satellites, phone lines wireless access points and other telecommunication media.



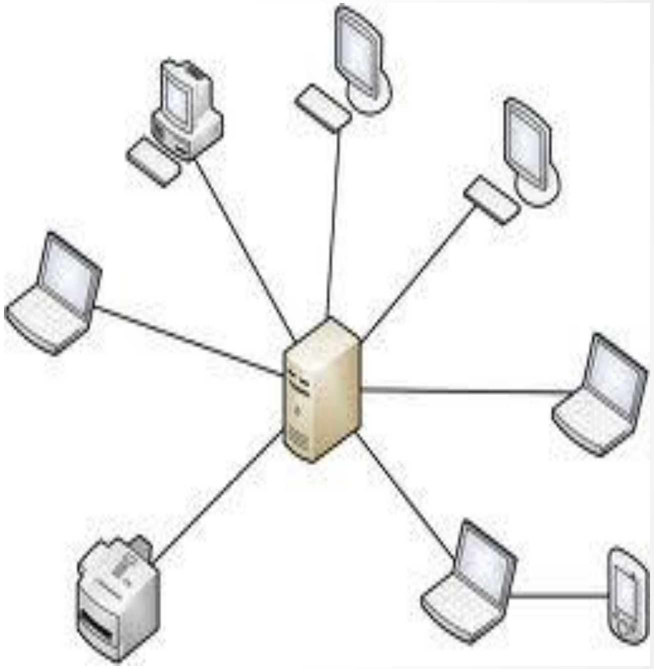
Text: New Perspectives on HTML, XHTML and XML - Comprehensive, 3rd Edition by Carey

World Wide Web

- The World Wide Web, or simply Web, is one way of accessing information over the medium of the Internet.
- It is an information-sharing model that is built on top of the Internet, where information is not accessed in a linear fashion, but allows documents to be connected using **hypertext links** forming a huge “web” of connected information.
- It is based on a protocol called HTTP protocol, only one of the languages spoken over the Internet, to transmit data.

Basic Terms

- A network is a structure linking computers together for the purpose of sharing information and services.
- Users typically access a network through a computer called a host or node.
- A node that provides information or a service is called a server.
- A computer or other device that requests services from a server is called a client.
- Networks can be designed in many ways - one of the most commonly used designs is the client-server network.



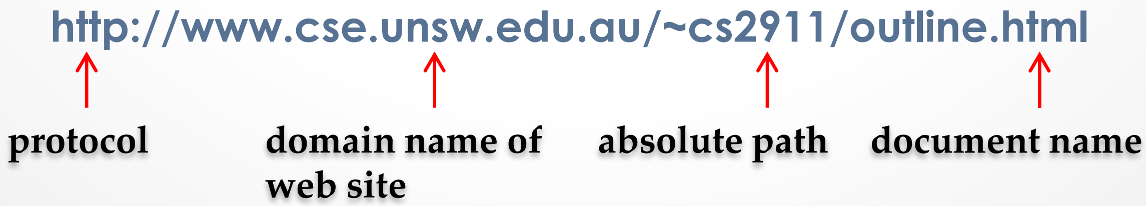
Text: New Perspectives on HTML, XHTML and XML - Comprehensive, 3rd Edition by Carey

Web Pages and Web Servers

- Each document on the World Wide Web is referred to as a Web page and it is basically a text file written in HTML or Hypertext Markup Language

- Web pages are stored on Web servers (HTTP servers), which are computers that run a special server software that enables communication between computers through the HTTP protocols and these servers make available any web page to any device connected to the Internet e.g., Apache HTTP server, Microsoft IIS server

- Each web page has a special address, URL (Uniform Resource Locator)



http://www.cse.unsw.edu.au/~cs2911/outline.html

protocol

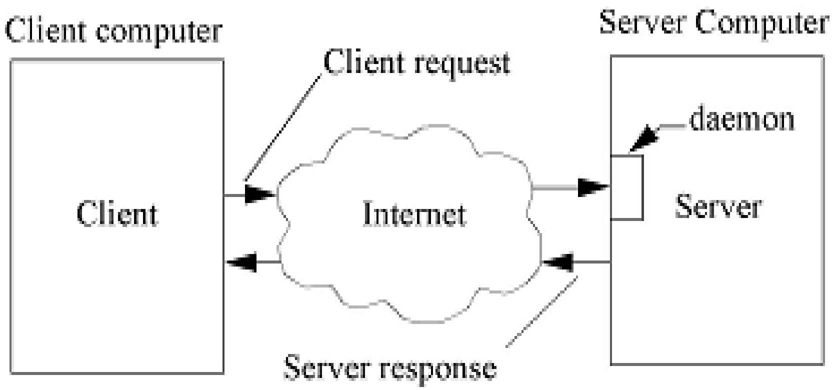
domain name of web site

absolute path

document name

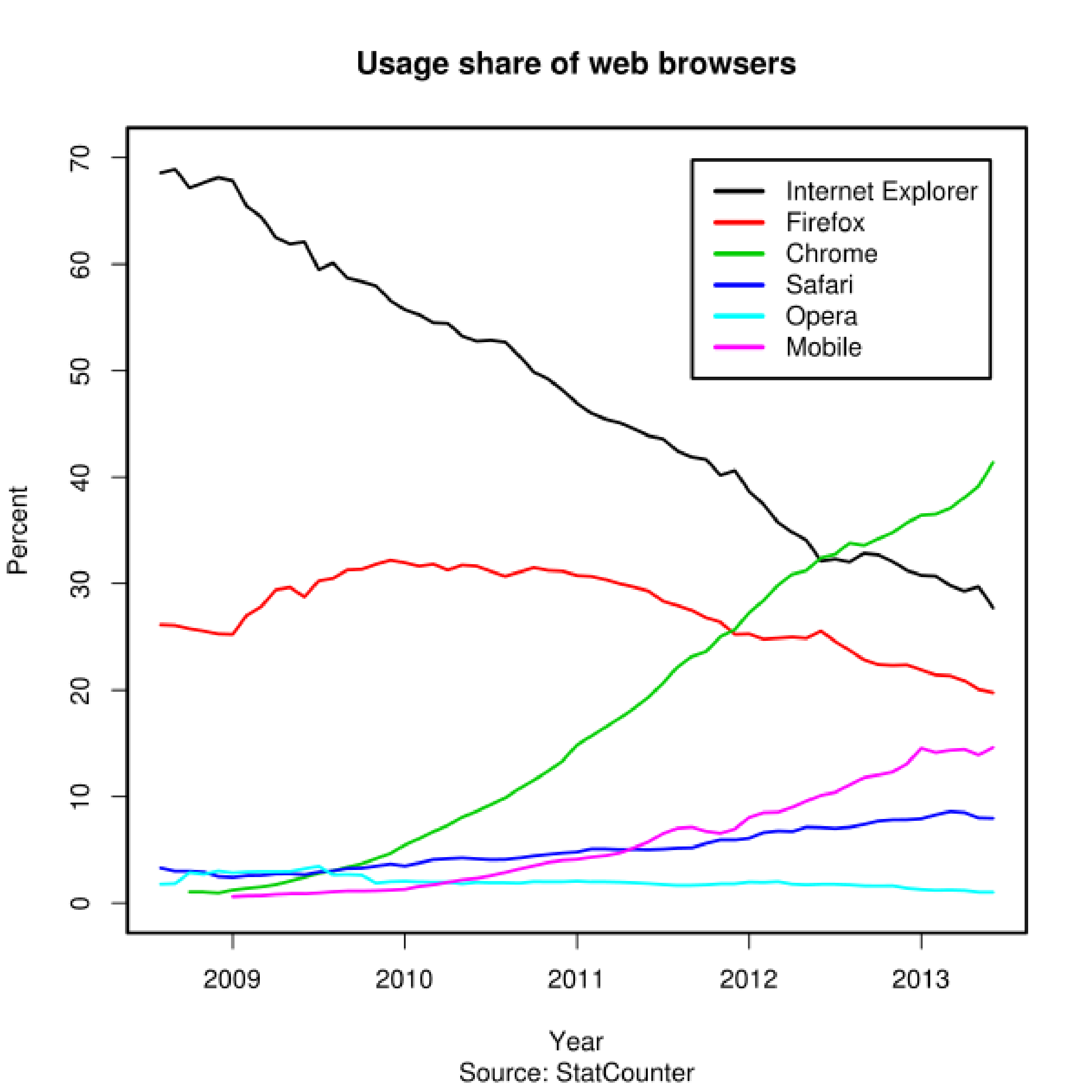
Web Browsers

A Web browser is a software running on the end-user's device that retrieves the page from the Web server and renders it on the user's device



Text: New Perspectives on HTML, XHTML and XML - Comprehensive, 3rd Edition by Carey

Web Browsers Today



From the Wikipedia article on Chrome.

How is a web page assembled ?

- A client requests a web page by specifying the URL or clicking on a hyper link

- Browser sends a HTTP request to the server named in the URL and requests for the specific document

- The web server locates the requested file and sends a response.
    - If document is not found, an error message "404, Not found" is returned
    - If the document is found, the server returns the requested file to the browser

- The browser parses the HTML document and assembles the page
    - If the page contains images, the browser requests the server for the image, inserts the image into the document in the position indicated and displays the assembled page

HTML (HyperText Markup Language)

- Web pages are text files written in HTML, a platform-independent, markup language that describes the content and structure of the document

- The textual content is represented by plain text and the structure of the document is made up elements, which are distinct objects in the document like a paragraph, a tiltle and are indicated in the document by HTML tags

  <element> textual content </element>

  e.g., <p>, </p>

- HTML is not a programming language

- It is also NOT a formatting language i.e., it does not specify how the content should be rendered. The browser on the user's devices parses the HTML tags and renders the content. Use styles for formatting

Structure of an HTML File

An HTML document is divided into two main sections: the **head** and the **body.**

- The root element <html> ….. </html>
- The head element <head> ….. </head>
  - It contains information about the document, for example the document title or the keywords.
  - The title element contains the page's title. A document's title is usually displayed in the browser's title bar (or as a title of the tab)
  - The content of the head element is not displayed within the Web page.

- The body element <body> ….. </body>
  - It contains all of the content to appear on the Web page.
  - It contain code that tells the browser how to render the content.

Structure of an HTML Page

```
A basic HTML page has the following structure
<!DOCTYPE html>
<html>
       <head>
              <title> My Web Page </title>
              ... head content
       </head>
       <body>
              ... body content
       </body>
</html>
```

- <!DOCTYPE html> means we are using HTML5

Most "tags" have matching closing tags

Example HTML Page

```
<!DOCTYPE html>
<html>
<head>
 <title>Example Web Page</title>
</head>
<body>
<p>The cat sat on the mat.</p>
</body>
</html>
```

p for paragraph

124 characters

HTML Layout and Indentation

- It is easier to get HTML right if you lay it out neatly – with parts that are "inside" other parts *indented* to show the structure.

```
<!DOCTYPE html >
<html>
<head>
         <title> Example Web Page </title>
</head>
<body>
        <p>
                The cat sat on the mat.
        </p>
</body>
</html>
```

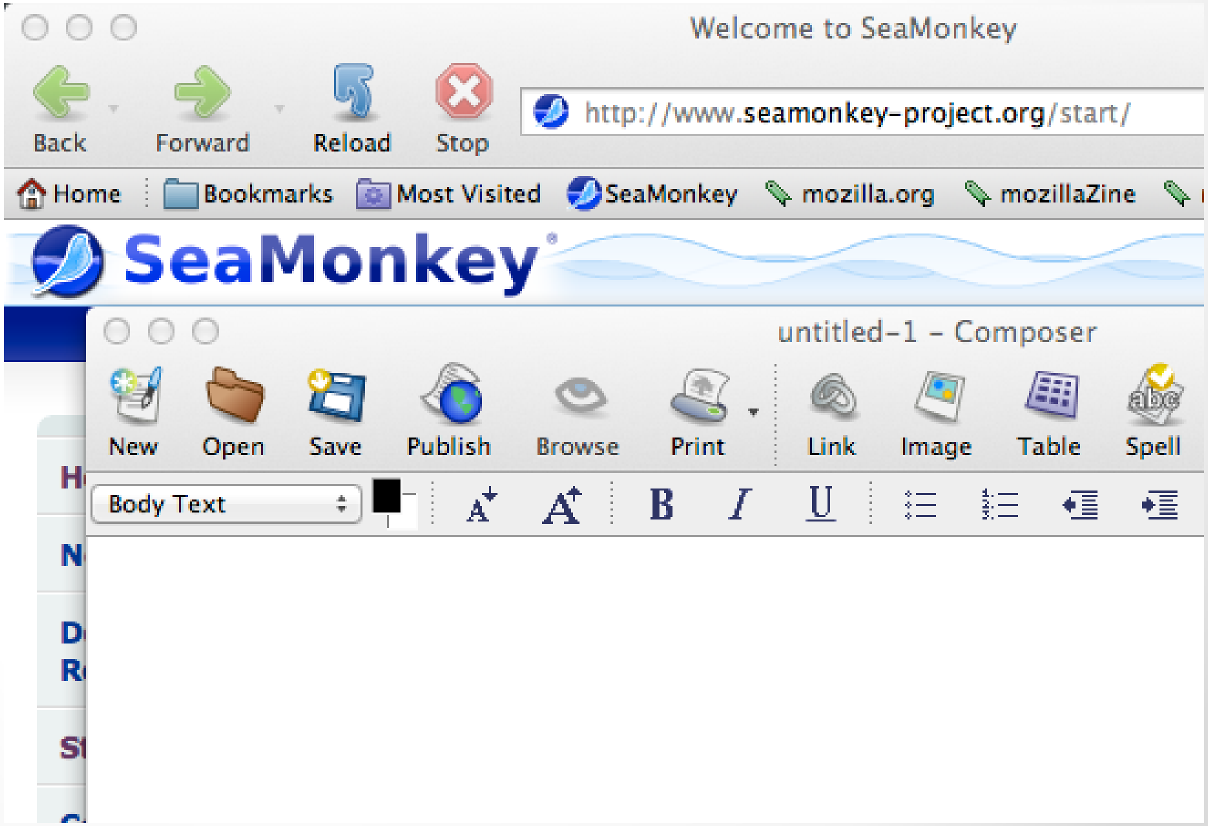
Web Page Construction Tools

- One way to create a web page is to construct it in Microsoft Word and then Save as…/Format arrow/Web page (.htm)
- This works, but is not ideal: typically you will want to adjust the content to do something Word cannot do.
- And …
  - If you make a Word document with “The cat sat on the mat.” in it and save as .html, the resulting html file contains, not 124 characters, (9 lines) but about 21,000 characters (400+ lines). (.htm *filtered* format gives about 900 characters and 46 lines).
  - .

Web Page Construction Tools (Cont'd)

- A purpose-made WYSIWYG web page tool is a better option.
- You will still want to adjust the content at times.
- SeaMonkey is one such (free download).

After starting SeaMonkey, click **File > New > Composer Page**



Cat example from SeaMonkey

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
<html>
 <head>
  <meta http-equiv="Content-Type“ content="text/html; charset=ISO-8859-1“>
  <title>Example Web Page</title>
</head>
<body>
      The cat sat on the mat.<br>
</body>
</html>
```

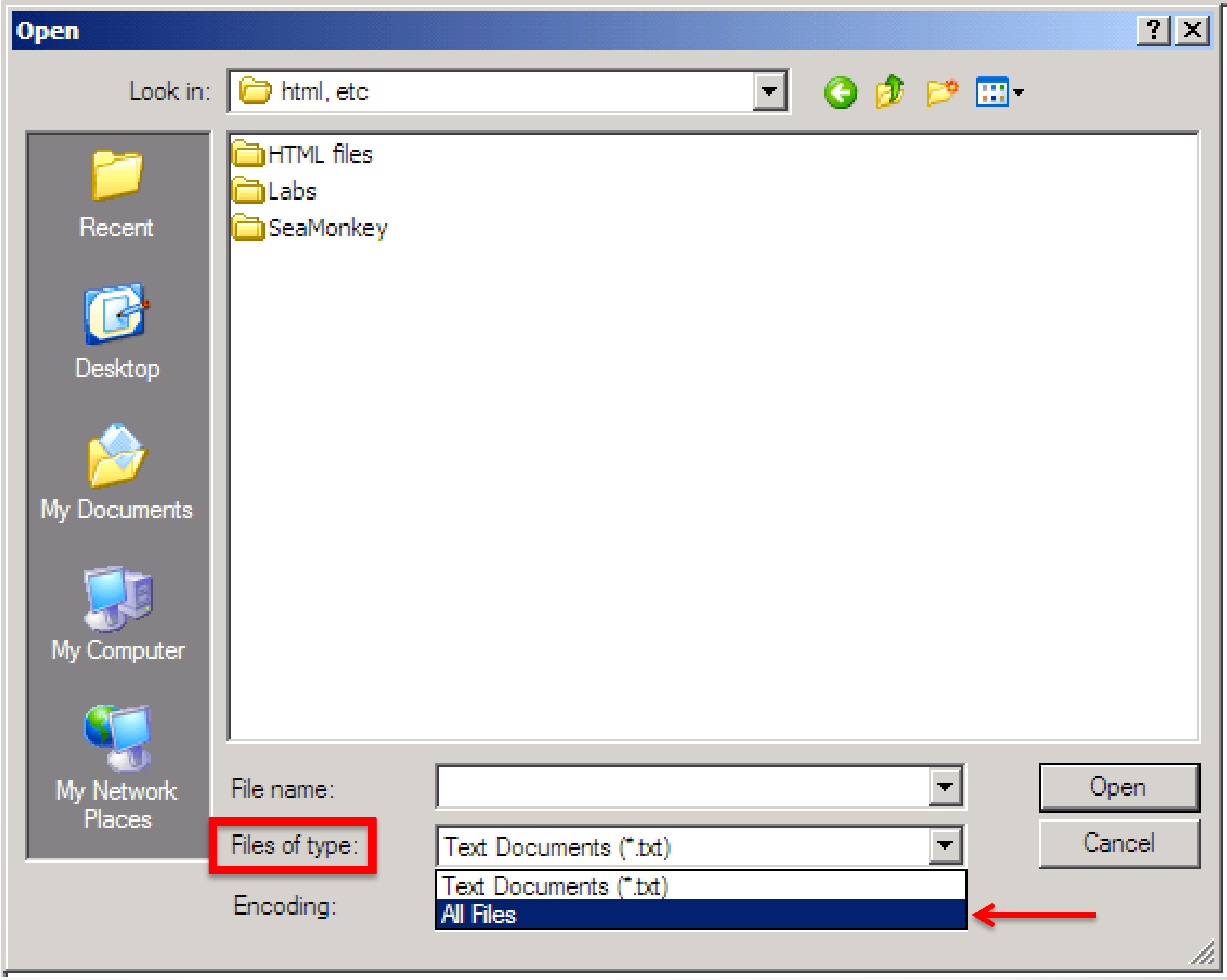
- Slightly different but quite compact (247 characters / 11 lines).
- The first line specifying the DOCTYPE shows that it is using a different version of html

Writing HTML by hand

- In this course we will be learning to write HTML using nothing but our brain and notepad (or a similar text editor) ☺

Windows

Use Notepad (Start/All Programs/Accessories/Notepad). To open .css and .html files, see right.→ Ensure you have the right extension when saving.



Open

Look in: html, etc

HTML files

Labs

SeaMonkey

File name:

Files of type: Text Documents (*.txt)

Text Documents (*.txt)

All Files

Encoding:

An Example

- It is easier to get HTML right if you lay it out neatly – with parts that are "inside" other parts *indented* to show the structure.

```
<!DOCTYPE html >
<html>
<head>
     <title> Example Web Page </title>
</head>
<body>
    <p>
        Welcome to Comp1000 17s1
    </p>
</body>
</html>
```

Adding Comments

- The comment tag adds notes to your HTML code

<!– “comment” -->

- Comments are not displayed by the browser
- Comments can be spread over several lines.
- Comments are useful in documenting your HTML code for yourself and others.

Marking Elements with Tags

- The core building block of HTML is the tag, which marks the presence of an element.
- A two-sided tag is a tag that contains some document content. General syntax for a two-sided tag:

  <element>content</element>

- A two-sided tag's opening tag (<p>) and closing tag (</p>) should completely enclose its content.
- Elements can contain other elements.
  - Tags cannot overlap

    <p>Welcome to <em>COMP1000 17s1</em></p>

Marking Elements with Tags

- Many tags can have attributes, which modify the behavior of the tag
  - Eg. alignment, line justification, block size
- Each attribute has a value enclosed in double quotes
- Attributes are specified within the opening tag to which they belong:
  - <element attribute="value">content</element>
  - Ex: <p align = "center"> content</p>

- The closing tag cannot contain any attributes

Block-Level vs Inline Elements

- **Block-level** elements are elements/tags that contain content that is viewed as a distinct block within the Web page.
  - Eg. Headings, paragraphs, tables
  - The browser (usually) inserts a new line between each block-level element
- An inline element marks a section of text within a block-level element
  - They can contain other inline elements but can't contain block-level elements
  - <em> </em> is an inline ele

Important Block-Level Elements

- Headings: <hn>*content*</hn>
  - where *n* is an integer from 1 to 6 and *content* is the text of heading.
  - <h1>This is a heading</h1>
  - h1 is the most important heading, h2 is a subheading etc.
- Paragraphs <p>*content*</p>
  - Paragraphs tag <p> - creates a line space ahead of text. The attribute align has values of left, center or right.
    - <p align="center">

Block-Level List Elements

- HTML supports three kinds of lists: **ordered**, **unordered**, and **definition**
  - You use an **ordered list** for items that must appear in a numerical order
  - You use an **unordered list** for items that do not need to occur in any special order
  - The **definition list** contains a list of terms, each followed by the term's description.
- One **list** can contain another list
  - This is called a **nested list**

Text: New Perspectives on HTML, XHTML and XML - Comprehensive, 3rd Edition by Carey

Lists

- **Unordered lists:**
  - Created using the `<ul>` tag
  - The items of the list are created using the `<li>` tag
- **Ordered lists:**
  - Created using the `<ol>` tag and has a type attribute
  - The items of the list are created using the `<li>` tag
- **Definition lists:**
  - Created using the `<dl>` tag
  - It consists of term/definition pairs created using `<dt>`, `<dd>` tags

Some other Block-Level Elements

- <pre>…</pre>
  - Preformatted text – retaining all whitespace and special characters
- <address> … </address>
  - Contact information
- <blockquote>…</blockquote>
  - an extended quotation
- <div>…</div>
  - Used to group elements – important when we use CSS

Text Formatting Inline Elements

<b>…</b>	**Defines bold text**
<i>…</i>	*Defines italicized text*
<strong>…</strong>	**Defines important text**
<em>…</em>	*Defines emphasized text*
<small>…</small>	Defines smaller text
<sub>…</sub>	Defines subscripted text
<sup>…</sup>	Defines superscripted text
<ins>…</ins>	Defines inserted text
<del>…</del>	~~Defines deleted text~~
<mark>…</mark>	Defines marked/highlighted text
<code>…</code>	`Defines computer code text`
<q>…</q>	“Defines inline quotes”

- http://www.w3schools.com/html/html_formatting.asp

Working with Empty Elements

- An empty element contains no content
- Empty elements appear in code as one-sided tags

  <element />
- The one-sided tag to mark a line break is

  <br /> or <br>
- The horizontal rule element places a horizontal line across the Web page

  <hr /> or <hr>

Text: New Perspectives on HTML, XHTML and XML - Comprehensive, 3rd Edition by Carey

Hyperlinks

- Hyperlinks create hypertext, which is the driving force of the web
- Hyperlinks can be used as:
  - Links (interdocument links)
  - Anchors (intradocument links)
- Links start at the source and point to a destination
- Link source may be text or an image, and the destination may be a file, a web page, program, image, video/audio file, etc.
- The source of the link may be shown as visited or unvisited

Hyperlinks

- A hyperlink is created by using the <a> tag.
  - <a href="*insert URL here*">link name here</a>
- The most important attribute of the <a> tag is href
  - href defines where the link's destination can be found
- Example:

```
<a href="http://www.unsw.edu.au">UNSW</a>
```

Absolute vs Relative URLs

- The URL (Universal Resource Locator) can be *absolute* or *relative*.
- Absolute URLs start with http (or sometimes ftp, mailto, https) and spell out the path to the document being linked, in full. e.g.

  "http://www.cse.unsw.edu.au/~cs1000/17s1/courseOutline.html"

- Relative URLs often point to a file in the same folder/directory as the page currently being displayed.

  ```
  <a href="solution.html">Solution is here</a>
  ```

Hyperlinks - Anchors

- Anchors use the <a> tag to link different sections of the same web page.
- The creation of an anchor requires
  - an element with an id to be defined

    ```
    <p id="aims">
    ```

  - and a link to that location in the file

    ```
    <a href="#aims">
    ```

    Note the "#" in the href attribute with the same value "aims" which can refer to the top of the same document

Images

To include an image in the page, use the <img> tag and the src attribute that specifies where the image is saved

Example:

```
 <img src="images/night.gif" />
```

or

```
<img src="images/night.gif" > </img>
```

Other attributes include alt border width height align hspace vspace etc.

Images

- `src` Attribute
  - File name of the graphic
- `height` Attribute
  - Height of the graphic in pixels
- `width` Attribute
  - Width of the graphic in pixels
  - Use height and width attributes to cause the browser to load efficiently otherwise the browser may shift the image position after loading.
- `alt` Attribute
  - Configures alternate text content for browsers to display
  - Useful for spider programs and vision-impaired people

Image Links

To create an image link use an anchor element to contain an image element

```
<a href="http://www.unsw.edu.au">
     <img border=“5“
     src="Images/something.gif"
     alt=“A nice logo“
     height="62"
     width="300">
</a>
```

Graphics

- Graphics can add a ‘wow’ factor to web pages BUT they can make web pages very slow to load.
- Web browsers load documents, line-by-line, starting at the top of the document. The top of an image begins to display after 50% of it has been read by the browser.
- Deciding which type of image to use depends on the balance between image type, its size and other factors like compression of the image.
- Graphic types commonly used on Web pages:
  - GIF (Graphics Interchange Format)
  - JPEG, JPG (Joint Photographic Expert Group)
  - PNG (Portable Network Graphic)

Working with GIF Images

- **GIF (Graphics Interchange Format)** is the most commonly used image format on the Web
- Compatible with virtually all browsers
- GIF files are limited to displaying 256 colours
- Often used for graphics requiring fewer colours, such as clip art images, line art, logos, and icons
- Images that require more colour depth, such as photographs, can appear grainy when saved as GIF files.

Text: New Perspectives on HTML, XHTML and XML - Comprehensive, 3rd Edition by Carey

GIFs

- Best used for line art and logos
- Maximum of 256 colors
- One color can be configured as transparent
- Uses lossless compression
- Can be animated
- Can be interlaced, progressively displayed 13% to 25% to 100%

Text: New Perspectives on HTML, XHTML and XML - Comprehensive, 3rd Edition by Carey

JPEG Images

- JPEG stands for Joint Photographic Experts Group
- Supports up to 16.7 million colours
- Most often used for photographs and other images that cover a wide spectrum of colour
- Usually smaller than their GIF counterparts

Text: New Perspectives on HTML, XHTML and XML -
Comprehensive, 3rd Edition by Carey

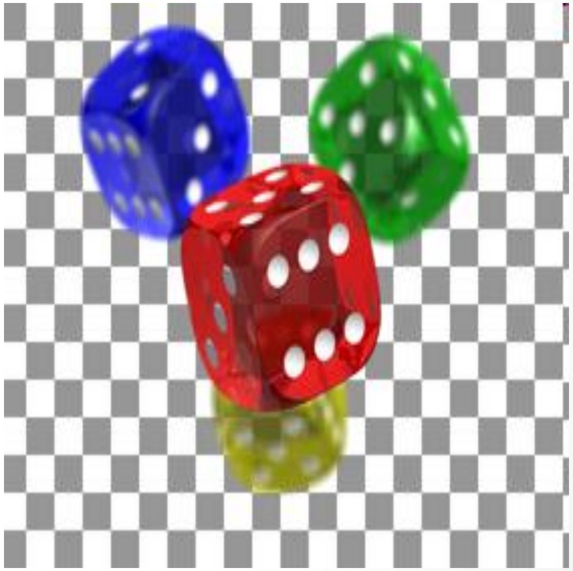
JPEG

- Best used for photographs
- Up to 16.7 million colors
- Uses lossy compression.
- Cannot be animated
- Cannot be made transparent
- Interlaced progressive JPEG

Text: New Perspectives on HTML, XHTML and XML - Comprehensive, 3rd Edition by Carey

PNG

- A file format called PNG (Portable Network Graphics) has been gaining wider acceptance
- PNG files use a free and open file format and can display more colours than GIFs



- Support millions of colours
- Support interlacing
- Use lossless compression
- Browser support is growing

Text: New Perspectives on HTML, XHTML and XML - Comprehensive, 3rd Edition by Carey

Tables

Tables can be used to organise and structure content on a web page.

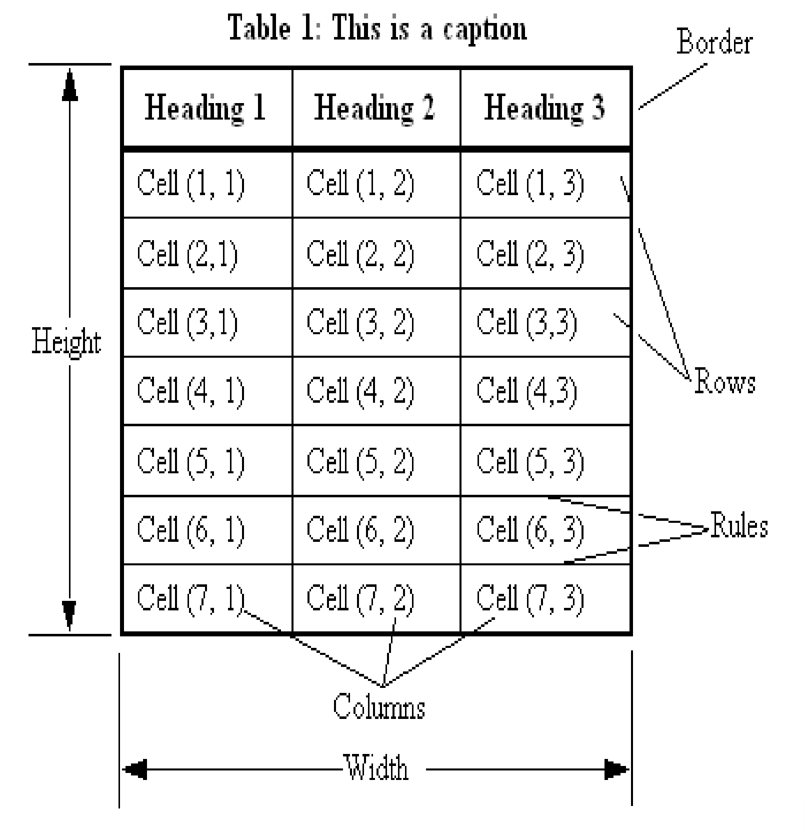


Table 1: This is a caption

Heading 1	Heading 2	Heading 3
Cell (1, 1)	Cell (1, 2)	Cell (1, 3)
Cell (2,1)	Cell (2, 2)	Cell (2, 3)
Cell (3,1)	Cell (3, 2)	Cell (3,3)
Cell (4, 1)	Cell (4, 2)	Cell (4,3)
Cell (5, 1)	Cell (5, 2)	Cell (5, 3)
Cell (6, 1)	Cell (6, 2)	Cell (6, 3)
Cell (7, 1)	Cell (7, 2)	Cell (7, 3)

Text: New Perspectives on HTML, XHTML and XML - Comprehensive, 3rd Edition by Carey

Introducing Web Tables

- Each table in a Web page follows a basic structure consisting of the table element and a collection of table rows nested in the table element and table data nested in each row

```
<table>
    <tr>
        <td> ... </td> <td> ... </td> ...
    </tr>
    <tr>
        <td> ... </td> <td> ... </td> ...
    </tr>
    ...
</table>
```

Text: New Perspectives on HTML, XHTML and XML - Comprehensive, 3rd Edition by Carey

Introducing Web Tables

- You can optionally specify table headings

```
<table>
    <tr>
        <th> ... </th> <th> ... </th> ...
    </tr>
    <tr>
        <td> ... </td> <td> ... </td> ...
    </tr>
    <tr>
        <td> ... </td> <td> ... </td> ...
    </tr>
    ...
</table>
```

Tables

<table> - Encloses all other table tags

- Attributes: summary, width, height, border, align, cell spacing and padding

<caption> - Assigns a title to the table

- Attributes: align

<tr> - Creates a table row

- Attributes: align, valign

<th> - Creates a header cell

- Attributes: abbr, headers, rowspan, colspan, align, valign

<td> - Creates data cells

- Attributes: abbr, headers, rowspan, colspan, align, valign

Table Structure and Attributes

- Table attributes can be broadly classified as:

  - Non-cell attributes – control properties of the whole table

    e.g., caption, border

  - Cell attributes – control properties of an individual cell

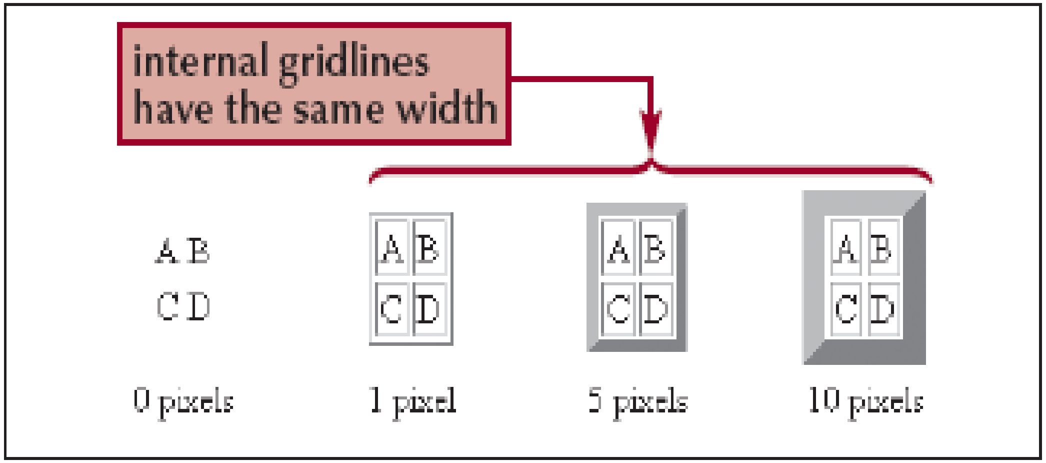
    e.g., column span, rowspan, cell padding, cell spacing

Creating a Table Border

- To add a border to a Web table using HTML, use the border attribute

```
<table border="value"> ... </table>
```

where *value* is the size of the border in pixels.



Text: New Perspectives on HTML, XHTML and XML - Comprehensive, 3rd Edition by Carey

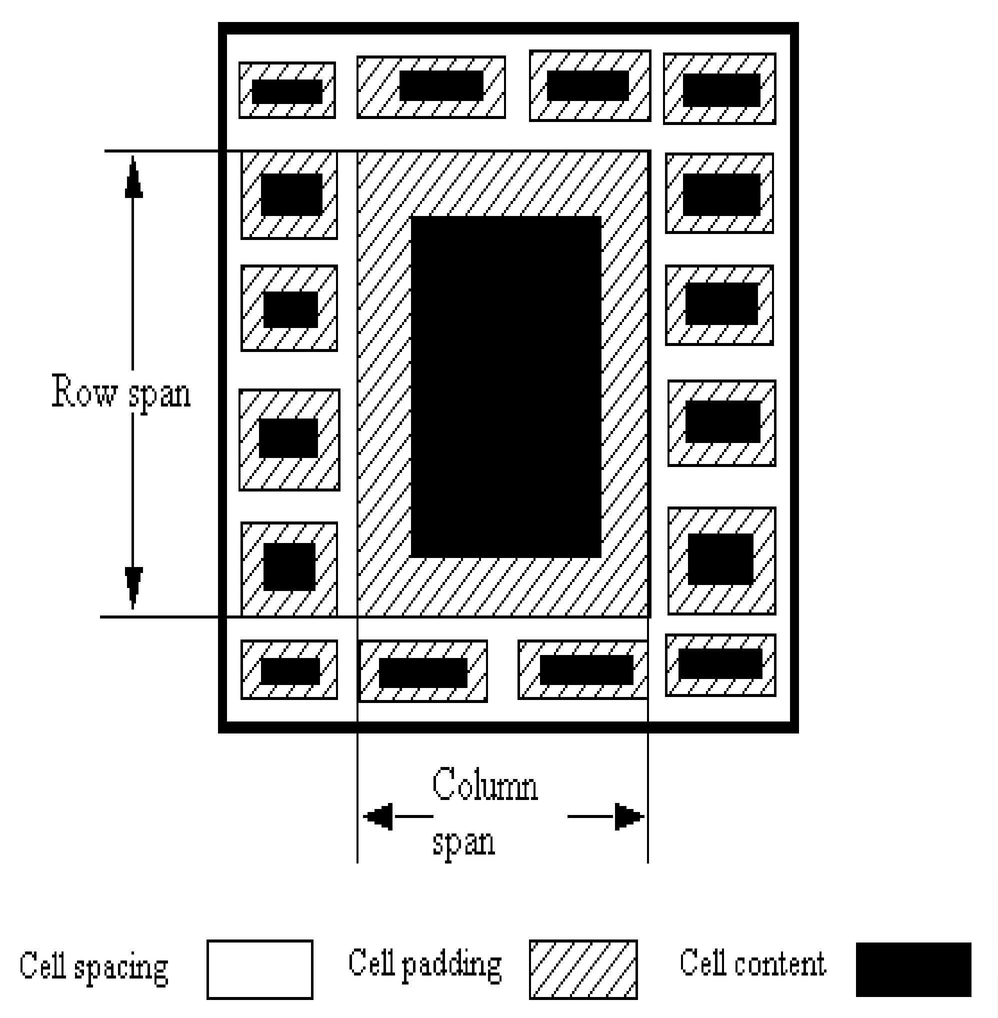
Creating a Table Caption

- To create a table caption, add the caption element directly below the opening `<table>` tag with the syntax

  ```
  <caption>content</caption>
  ```

  where *content* is the content of the table caption

Cell Structure and Attributes



Text: New Perspectives on HTML, XHTML and XML - Comprehensive, 3rd Edition by Carey

Spanning Rows and Columns

- A spanning cell is a single cell that occupies more than one row or one column in the table
- To create a table cell that spans several columns, add the attribute

  ```
  colspan="value"
  ```

  to the cell, where *value* is the number of columns covered by the cell
- To create a table cell that spans several rows, add the attribute

  ```
  rowspan="value"
  ```

  to the cell, where *value* is the number of rows covered by the cell

Text: New Perspectives on HTML, XHTML and XML - Comprehensive, 3rd Edition by Carey

Formatting Tables with HTML Attributes

- To define the padding within table cells, add the attribute

  ```
  <table cellpadding="value"> ... </table>
  ```

  to the table element, where *value* is the size of the padding space in pixels
- To define the space between table cells, add the attribute

  ```
  <table cellspacing="value"> ... </table>
  ```

  to the table element, where *value* is the space between table cells in pixels

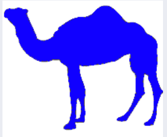
Text: New Perspectives on HTML, XHTML and XML - Comprehensive, 3rd Edition by Carey

```
<head>  <!- -  A table example - ->
<style>
    table, th, td {
       border: 1px solid black;
       border-collapse: collapse;  }
   th, td {
       padding: 5px;
       text-align: left;    }
</style>
</head>
<body>
<table style="width:100%">
  <caption style="font-weight:bold;font-size:18pt">  Food Nutrition Information
</caption>
  <tr>
     <th rowspan="2" style="text-align:center">Food</th>
     <th colspan="2" style="text-align:center">Nutrional Value</th>
  </tr>
  <tr>
     <th>Calories</th><th>Fat</th>
  </tr>
  <tr>
     <td>avacado</td>
     <td>160 calories</td>
     <td>15g</td>
  </tr>
</table> …
```

HTML Character Codes

- HTML character codes are of the form &*name*; or &*number*; here are some more:
  - &lt; is <
  - &gt; is >
  - &amp; is &
  - &nbsp; is non-breaking space
  - &rarr; is ®
  - &alpha; is the Greek letter a
- more at http://www.cse.unsw.edu.au/~billw/symbols.html

Summary of Basic HTML elements

This HTML	produces this…	Comment
<p>text</p>	paragraph with text in it	<p align="center"> etc.
<ul><li>text1<li>text2</ul>	• text1 • text2	Technically, each <li> should have a closing </li>
<ol><li>text1<li>text2</ol>	1 text1 2 text2	<ol type="a"> gives a, b, c <ol type="i"> gives i, ii, iii, iv, …
<img src="camel.gif" alt="picture of camel" />		.gif, .jpg, .png, … Note: ends with />, no matching closing tag.
<h1>Heading</h1>	**Heading**	<h1> > <h2> > <h3> > <h4>
&gt; &amp; &lt; &le; &ouml; etc.	> & < ≤ ö etc.	&nbsp; = non-breaking space
x<sup>2</sup> x<sub>1</sub>	x^2 x_1	
<i>x</i> <b>x</b> <u>x</u>	*x* **x** x	
text1<br />text2	text1 text2	Forces new line. Note: ends with />, no matching closing

"How did they do that?"

- Sometimes you look at a web page with some interesting formatting and you wonder "How did they do that?"
- Most web browsers have a way to view the HTML source of the page – thus you can find out how they did that.
- In Chrome, use View>Developer>View Source
- In Firefox, use Tools>Web Developer>Page Source
- In Safari, the first time, do Safari>Preferences…>Advanced and tick the "Show Develop menu in menu bar" box, then use Develop>Show Page Source
- In Internet Explorer use View>Source
- In Opera, use View > Developer Tools > Source

Deprecated Methods, and Alternatives

- HTML has "deprecated" methods.
- This means formatting commands that they are trying to phase out.
- Phasing out proceeds *very* slowly, as existing HTML out there on the web depends on the deprecated stuff.
- For example, <font …>…</font> is deprecated.
- The current preferred method is to use something like:

  <span style="color: red; ">red stuff</span>
- rather than <font color=red>red stuff</font>
- <span …> does nothing but provide a location for the style=…
- This style=… scheme is part of **CSS**.

CSS = Cascading Style Sheets

- CSS, or Cascading Styles Sheets, is a way to style and present HTML. Whereas the HTML is the meaning or content, the style sheet is the presentation of that document.

- Styles have a format of ‘property: value’ and most properties can be applied to most HTML tags.

CSS = Cascading Style Sheets

- CSS style specifications can be put in three places:
  - Inline style attributes (wrapped around the content)

    ```
    <span style="color: rgb(0, 0, 255); ">blue stuff</span>
    ```
  - Internal HTML style element in <head>…</head>

    ```
    <style type="text/css">
    em {color: red; }
    </style>
    ```
  - External Style sheet (.css file) referenced in <head> of html

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

    The style sheet might specify that <p> text is purple.
  - There are also default styles (defined by the browser).
- Nearest style takes precedence in case of conflict.

Linking CSS style-sheets to your HTML

- To link your .css file, called say my.css, to your .html file, you need to insert the following between <head> and </head> in your html, say after the <title> … </title> element, and before the <style>…</style> element, if any:

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

- Naturally, you'd replace *my*.css by whatever name you choose for your .css file.

CSS

- So, the following HTML …

```
<body>
This is black.
<p>This is purple <em>but this is red
and <span style="color: blue; ">this is blue.</span>
</em>
</p>
</body>
```

- produces (assuming my.css says <p>is purple and is linked and that we have <em> set to red using an internal style element):

This is black.

This is purple *but this is red and this is blue.*

CSS Syntax

- A CSS rule set consists of a selector and a declaration block

-
  ```
  p {
     text-align:center;
      color:red;
  }
  ```

-
  ```
  /*Can also be written more compactly. By the way this is a comment */
  h1 { color:blue; font-size:12px; }
  ```
- Can group selectors

```
h1,h2, p {
    text-align:center;
    color:red;
}
```

CSS

- CSS is *case sensitive*.
- Some style properties can take a sequence of values:

  ```
  p{
    font-family:  “Times New Roman”, Times, serif;
   }
  ```

- This text appears in Times New Roman if available
- The *font-family* property specifies a choice of fonts separated by commas, in priority order.
- Multi-word arguments like “Times New Roman”, must appear in quotes.

Selectors

selector { property$_1$: value$_1$;
property$_2$: value$_2$; …
property$_m$: value$_m$;
}

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

The selector will be the name of an HTML element, pseudo-element, class, pseudo-class or id of an element

Selector	Examples
element	p p strong (contextual selector)
pseudo-element	p:first-letter p:first-line
class	.highlight
pseudo-class	a:link a:hover a:active a:visited
id	#para1

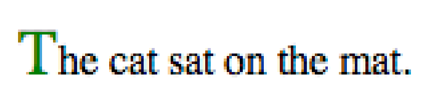
Using Element Styles

Aspect	Example
specify	p {font-weight: bold; font-size: 18pt; }
use	<p>content</p>
remark	Modifies every instance of the <p>…</p> tag, unless overridden by an in-line style.

Using Contextual Element Styles

Aspect	Example
specify	p b, p strong {color: orange;}
use	<p>content<b>bold stuff</b>more content</p>
remark	Modifies instances of the <b> tag, but only if they occur inside a <p>…</p> context, (and unless overridden by an in-line style).
appearance	<b>regular bold</b><p>and <b>bold</b> inside p tag</p> → **regular bold** and **bold** inside p tag

Using Pseudo-element Styles

Aspect	Example
specify	p:first-letter {color: green; font-size: 24px; }
use	<p>The cat sat on the mat. </p>
appearance	→  The cat sat on the mat.

Using Class Styles

Aspect	Example
specify	.highlight {background-color: green; }
use	<span class="highlight">content</span> <td class="highlight">content</td> <h1 class="highlight">heading</h1>
remark	Can be used to modify style in any tag: <h1>heading</h1> works as usual, but <h1 class="highlight">heading</h1> highlights the heading in green. You use a class style when different instances of elements in the same document need different formatting.

Using Pseudo-class Styles

Aspect	Example	

specify	a:link {color: blue; }
	a:visited {color: purple; }
	a:hover {color: green; size: 150%; }
	a:active {color: orange; }	

use	The meaning of life can be found <a href="*URL*">here</a>	
appearance	Before visiting:	The meaning of life can be found here.
… tested	After visiting:	The meaning of life can be found here.
… with	When hovering:	The meaning of life can be found here.
… Chrome	Mouse clicked but not released:	The meaning of life can be found here.

Using Element Id Styles

Aspect	Example
specify	#para1 {background-color: blue; }
use	<p id=“para1">content</p>
remark	Can be used to modify style in any tag: <p>hello</p> works as usual, but <p id=“para1">hello</p> sets background to blue. An id should be unique within a page, so you should use the id selector when you want to find a single, unique element.

CSS Font Properties

Font Property	Alternatives/Examples
font-family	font-family: Times, serif
font-style	font-style: italic font-style: oblique font-style: normal
font-variant	font-variant: small-caps font-variant: normal
font-weight	font-weight: bold font-weight: bolder font-weight: lighter font-weight: 300 font-weight: normal
font-size	font-size: 14px font-size: 18pt font-size: xx-small | x-small | small | medium | … xx-large font-size: larger | smaller font-size: 80%

CSS Text Properties

Text Property	Examples / Alternatives
line-height	line-height: 1.5 | 18 px | 110% | normal
text-decoration	text-decoration: underline | line-through | overline text-decoration: blink | none
text-transform	text-transform: lowercase | uppercase | capitalize | none
text-align	text-align: left | right | center | justify
text-indent	text-indent: 15px | 7%
white-space	white-space: pre | normal

Properties

Color or Background Property	Examples / Alternatives
color	color: red | rgb(255, 0, 0) | rgb(100%, 0%, 0%) color: #FF0000
background-color	background-color: transparent background-color: *alternatives as for* color
background-image	background-image: none | *url*
background-repeat	background-repeat: repeat | repeat-x | repeat-y background-repeat: no-repeat
background-attachment	background-attachment: scroll | fixed
background-position	background-position: top | center | bottom background-position: left| right background-position: 25px 25px ←*x y coordinates* background-position: 0% 10% *ditto*

CSS List Properties

List Property	Examples / Alternatives	Meaning
list-style-type	list-style-type: disc list-style-type: circle list-style-type: square list-style-type: decimal list-style-type: lower-roman list-style-type: upper-roman list-style-type: lower-alpha list-style-type: upper-alpha	• item ○ item ■ item 1. item i. item I. item a. item A. item
list-style-position	list-style-position: inside *(Windows only)* list-style-position: outside *(Windows only)*	1. blah blah blah blah blah 2. more blah 1. blah blah blah blah blah 2. more blah
list-style-image	list-style-image: *url* list-style-image: none	blah blah blah

CSS Summary

- Put the main style definitions that apply across all your web pages in a .css file.
- Link this to a particular web page by using a <link…> element in the <head> of the .html file.
- Also, in the <head> of the .html file – possibly put a <style> element to hold extra style definitions that apply just to this .html file.
- In the <body> of the .html file, here and there: in-line style definitions of a one-off nature.

Summary

In this section, you learned:

- the structure of an HMTL page
- about some web page construction tools
- basic web page elements
- how to create and use CSS style definitions