ITSE412 – Lecture 2

HTML Review

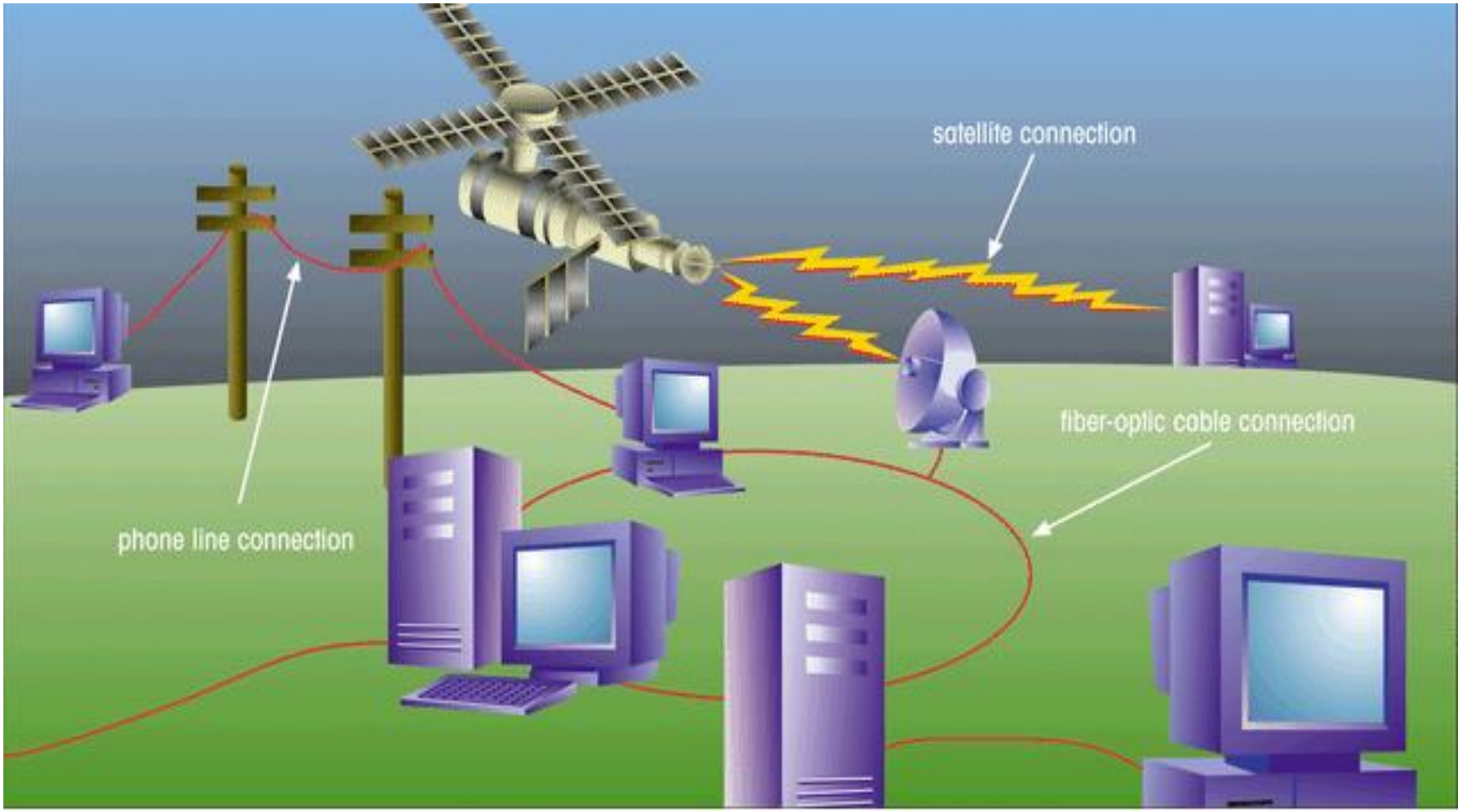
Introducing the World Wide Web

- A computer or other device that requests services from a server is called a **client**.
- One of the most common network structures is the **client-server network**.
- If the computers that make up a network are close together (within a single department or building), then the network is referred to as a **local area network (LAN)**.

Introducing the World Wide Web

- A network that covers a wide area, such as several buildings or cities, is called a **wide area network (WAN)**.
- The largest **WAN** in existence is the **Internet**.

Structure of the Internet



Linking to a Web Page

A sample URL for a Web page

Linking to a Web Page

- If a **URL** includes no path, then it indicates the topmost folder in the server's directory tree.

- If a **URL** does not specify a filename, the server searches for a file named "index.html" or "index.htm".

Understanding Markup Languages

- A Web page is simply a text file written in a language called Hypertext Markup Language (HTML).
- Hypertext: is a method of organizing information that gives the reader control over the order in which the information is presented.
- The key to use Hypertext is the use of Hyperlinks/links, which are the elements in a hypertext document that allow you to jump from one topic to another.

Understanding Markup Languages

- Markup Language: is a language that describes a document's structure and content.
- Web Site: an entire collection of linked documents.
- The hypertext documents within a Web Site are known as Web Pages.
- Styles: are format descriptions written in a separate language from HTML that tell browsers how to render each element.

History of HTML and XHTML

- A group of Web developers, programmers, and authors formed the World Wide Web Consortium or the W3C. (http://www.w3c.org)
- The World Wide Web Consortium (W3C) developed **specifications**, or **sets of standards**, that identify how a browser interprets the HTML code.
- The specifications are voluntary, but most organizations follow the specifications as much as possible.
- The current specification for HTML is 4.01.

Versions of HTML and XHTML

Version	Date	Description
HTML 1.0	1989–1994	The first public version of HTML which included browser support for inline images and text controls.
HTML 2.0	1995	The first version supported by all graphical browsers. It introduced interactive form elements such as option buttons and text boxes. A document written to the HTML 2.0 specification is compatible with almost all browsers on the World Wide Web.
HTML 3.0	1996	A proposed replacement for HTML 2.0 that was never widely adopted.
HTML 3.2	1997	This version included additional support for creating and formatting tables and expanded the options for interactive form elements. It also supported limited programming using scripts.
HTML 4.01	1999	This version added support for style sheets to give Web designers greater control over page layout. It added new features to tables and forms and provided support for international features. This version also expanded HTML's scripting capability and added increased support for multimedia elements.
XHTML 1.0	2001	This version is a reformulation of HTML 4.01 in XML and combines the strength of HTML 4.0 with the power of XML. XHTML brings the rigor of XML to Web pages and provides standards for more robust Web content on a wide range of browser platforms.
XHTML 1.1	2002	A minor update to XHTML 1.0 that allows for modularity and simplifies writing extensions to the language.
XHTML 2.0	2004–	The latest version, designed to remove most of the presentational features left in HTML.

Comparison of HTML and XHTML Syntax Requirements

HTML	XHTML
No document type declaration is required	Required a DOCTYPE declaration at the beginning of the file, such as: <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd>
Two-sided tags are not required a closing tag	Two-sided tags are required both opening tag and closing tag; <element>content</element>, content
One-sided tags (empty elements) are written with the name of the tag only, contained no content, are omitting the space and closing slash.	One-sided tags must be closed by including a space and a forward slash in the tag; <element />,

Comparison of HTML and XHTML Syntax Requirements

HTML	XHTML
Tags can be written in uppercase or lowercase letters; <p> or <P>	Tags must be written in lowercase letters only
Attribute values do not need to be enclosed in quotation marks.	Attribute values must be enclosed in quotation marks.
Attributes can be minimized when the attribute name and value are identical, such as <option selected value="opt 1"> to indicate the status of a selected item in a drop-down list box instead of <option selected="selected" value="opt 1" />	Attribute minimization is prohibited.

Understanding Tags and Attributes

- Older features of HTML are often deprecated or phased out by the W3C.
- Elements: distinct object in the document, like a paragraph, a heading, or the page's title.
- Tags: codes that indicate different elements on a Web page such as headings, paragraphs, and bulleted lists.
- Nested tags: tags that are included within other tags.
- Tags are either two-sided or one-sided.
  - A two-sided tag: <element>content</element>
    - A new paragraph here
  - A one-sided tag: <element />
    -

Understanding Tags and Attributes

- A third type of tag is the comment tag. They are not displayed or used by the Web browser.
  - <!-- This is a comment line -->
  - <!– this is a multiple
    comment lines -->
- White space: the blank spaces, tabs, and line breaks are collapsed into a single occurrence.
- Some tags include attributes that specify additional information about the content to be formatted by the tag.
  - <element attribute1="value1" attribute2="value2"…>content</element>
  - This is a new paragraph
- NOTE: "element" refers to an object in a Web page, and "tag" refers to the HTML code that creates the object. We can create a (paragraph) element in a Web page by using the tag.

Creating an HTML Document

- It is a good idea to plan out a Web page before you start coding.
- Draw a planning sketch or create a sample document using a word processor.
- Preparatory work can weed out errors or point to potential problems.
- In planning, identify a document's different elements. An element is a distinct object in the document, like a paragraph, a heading, or a page's title.
- Formatting features such as **boldfaced** font, and *italicized* text may be used.

The Structure of an HTML File

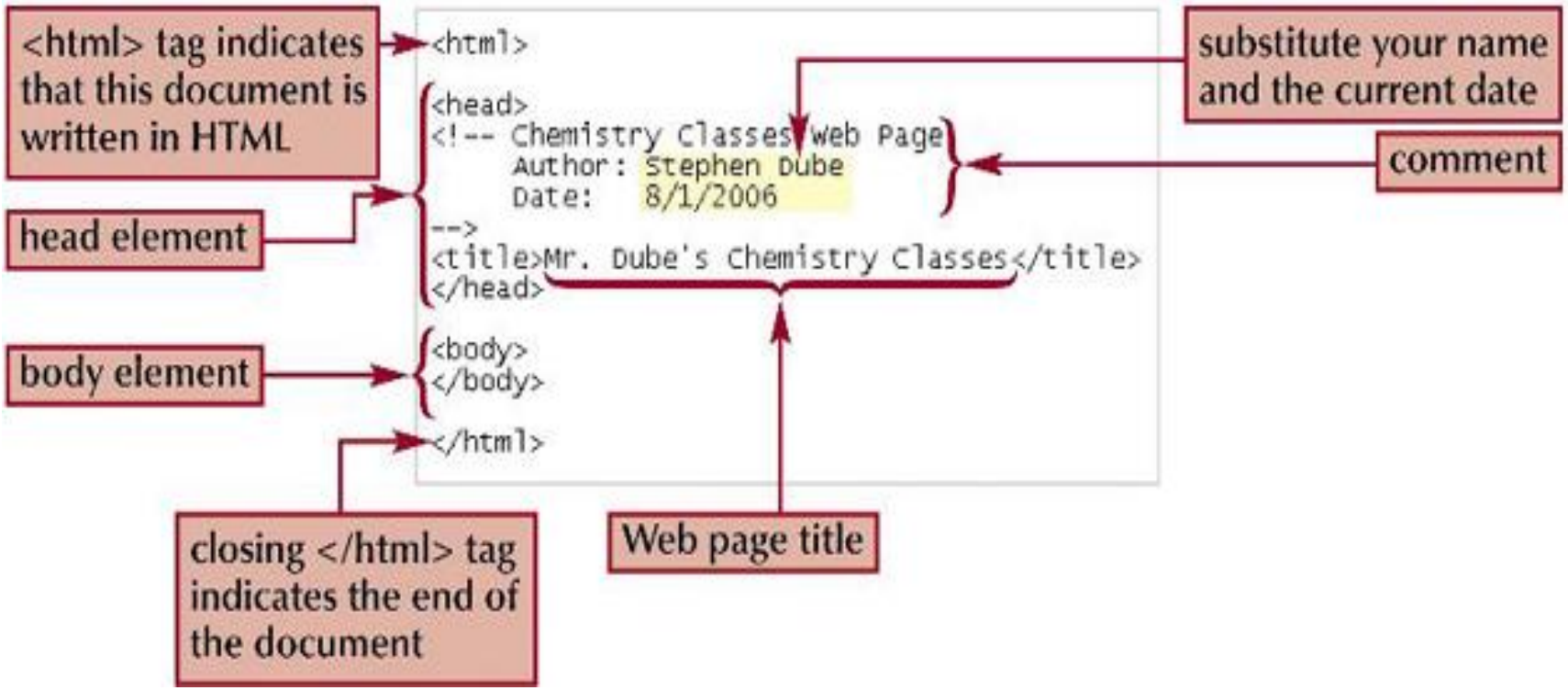
- The opening **<html>** tag marks the start of an HTML document, and the closing **</html>** tag tells a browser when it has reached the end of that HTML document.

- Anything between these two tags makes up the content of the document, including all other elements, text, and comments.

The Structure of an HTML File

- An HTML document is divided into two parts: the **head** and the **body**.
- The **head** element contains information about the document, for example the document title or the keywords.
- The content of the **head** element is not displayed within the Web page.
- The **body element** contains all of the content to be displayed in the Web page.

Sample HTML Code

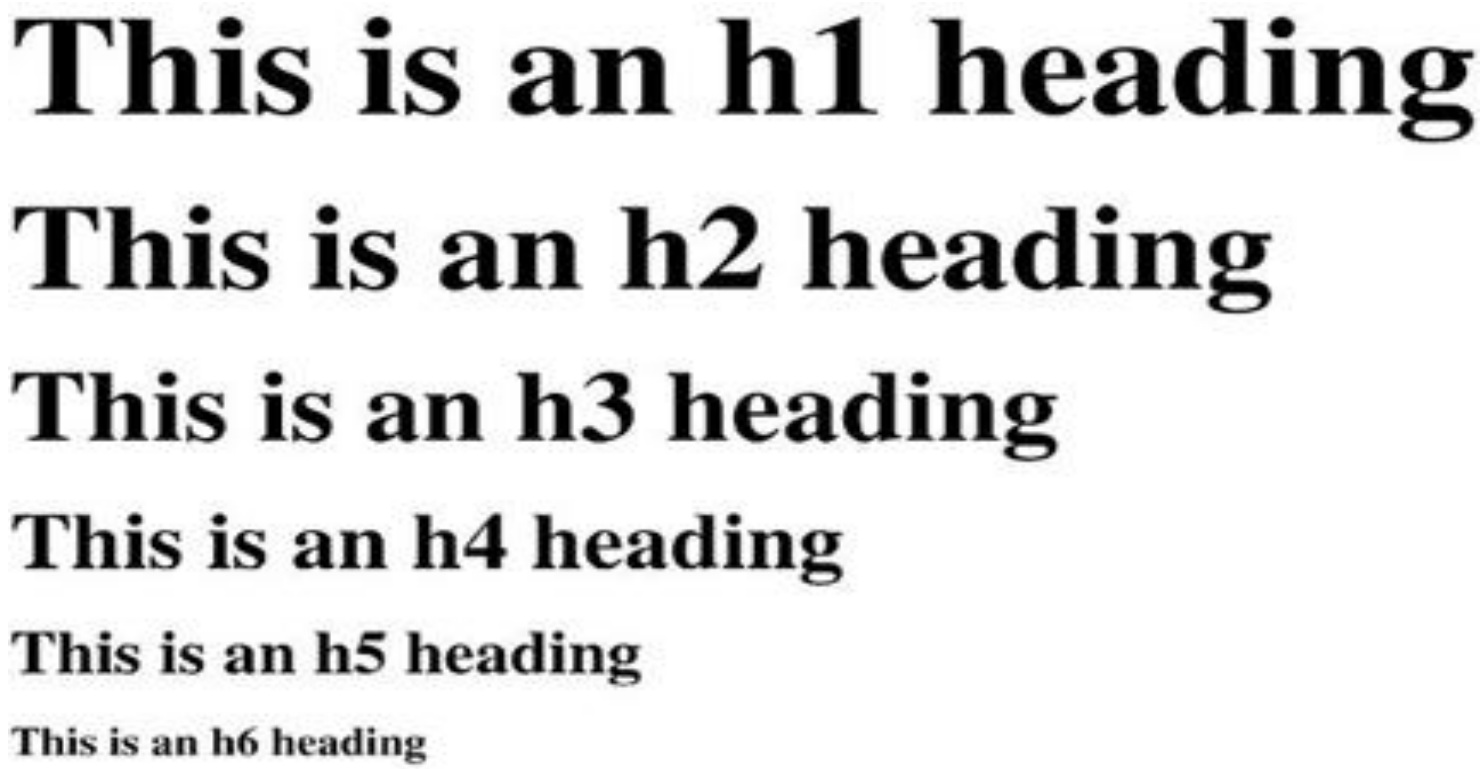


Block-Level Elements and Inline Elements

- In a Web page, most content is marked as either a **block-level** element or an **inline** element.
- A **block-level** element contains content displayed in a separate section within the page, setting it off from other blocks. For example, Paragraph , Heading <h1..6> </h1..6>, Ordered List <ol> , Unordered List <ul>
- An **inline element** is part of the same block as its surrounding content—for example individual words or phrases within a paragraph. Such as, Bold text , Citation text <cite> </cite>, Deleted text <del> </del>, Italic text , Subscript text , Superscript text

Block-Level Elements - Heading

- HTML supports six heading elements.



Inserting an Inline Style

- Use the **style attribute** to control the appearance of an element, such as text alignment.
  - <element style="style1: value1; style2: value2; …">
- Styles specified as attributes in a tag are also referred to as **inline styles**.
- The **text-align style** tells the browser how to horizontally align the contents of an element.
  - <h1 style="text-align: center">Text Align Center</h1>
- **Presentational attributes** specify exactly how the browser should render an element. For example the align attribute is a deprecated Presentational attribute.
  - <h1 align="center">Header align in the middle</h1>
- NOTE: HTML should inform the browser about the content of the document, and you should use styles to inform the browser how to render that content. For this reason, almost all presentational attributes have been deprecated in favor of styles.

Block-Level Element - Paragraph

- A paragraph element is using the tag.
- content
- Where content is the content of the paragraph.
- When a browser encounters the opening tag, it starts a new line with a blank space above it.
- In the earlier versions of HTML, you can omit the closing tag
- However, if you wish to write XHTML-compliant code, then you must include the closing tag.

Block-Level Element - Lists

- HTML supports three kinds of lists: **ordered, unordered, and definition**.
- You use an **ordered list** for items that must appear in a particular sequential order.

```
<ol>
<li>item 1</li>
<li>item 2</li>
</ol>
```

- You use an **unordered list** for items that do not need to occur in any special order.

```
<ul>
<li>item 1</li>
<li>item 2</li>
</ul>
```

- One **list** can contain another list. This is called a nested list.

```
<ol>
<li>item 1</li>
    <ul>
    <li>sub1</li>
    <.li>sub2</li>
    </ul>
<li>item 2</li>
</ol>
```

Block-Level Element - Lists

- The **definition list** contains a list of definition terms, each followed by a definition description.

```
<dl>
<dt>Term1</dt>
<dd>definition1</dd>
<dt>Term2</dt>
<dd>definition2</dd>
</dl>
```

- Web browsers typically display the definition description below the definition term and slightly indented.

Term1
    definition1
Term2
    definition2

Applying a Style to a List

- If you don't want your list items marked with either numbers or bullets, you can specify a different marker by applying the following style to either the ordered or unordered list
  - List-style-type: type
  - Where type is one of the markers listed below:

List-Style-Type	Marker (s)
disc	•
circle	o
square	■
decimal	1, 2, 3, 4, ...
decimal-leading-zero	01, 02, 03, 04, ...
lower-roman	i, ii, iii, iv, ...
upper-roman	I, II, III, IV, ...
lower-alpha	a, b, c, d, ...
upper-alpha	A, B, C, D, ...
none	*no marker displayed*

Applying a Style to a List: list-style-type

- Using different markers:

```
<ol style="list-style-type: lower-alpha">
<li>Item a</li>
<li>Item b</li>
</ol>

<ul style="list-style-type:  square">
<li>square item 1</li>
<li>square item 2</li>
</ul>
```

Applying a Style to a List: list-style-image

- You can also substitute a graphic image for a unordered list marker by using the style:
  - List-style-image: url(filename)

```
<ul style="list-style-image: url(c:/abc.jpg)">
<li>item1</li>
<li>item2</li>
</ul>
```

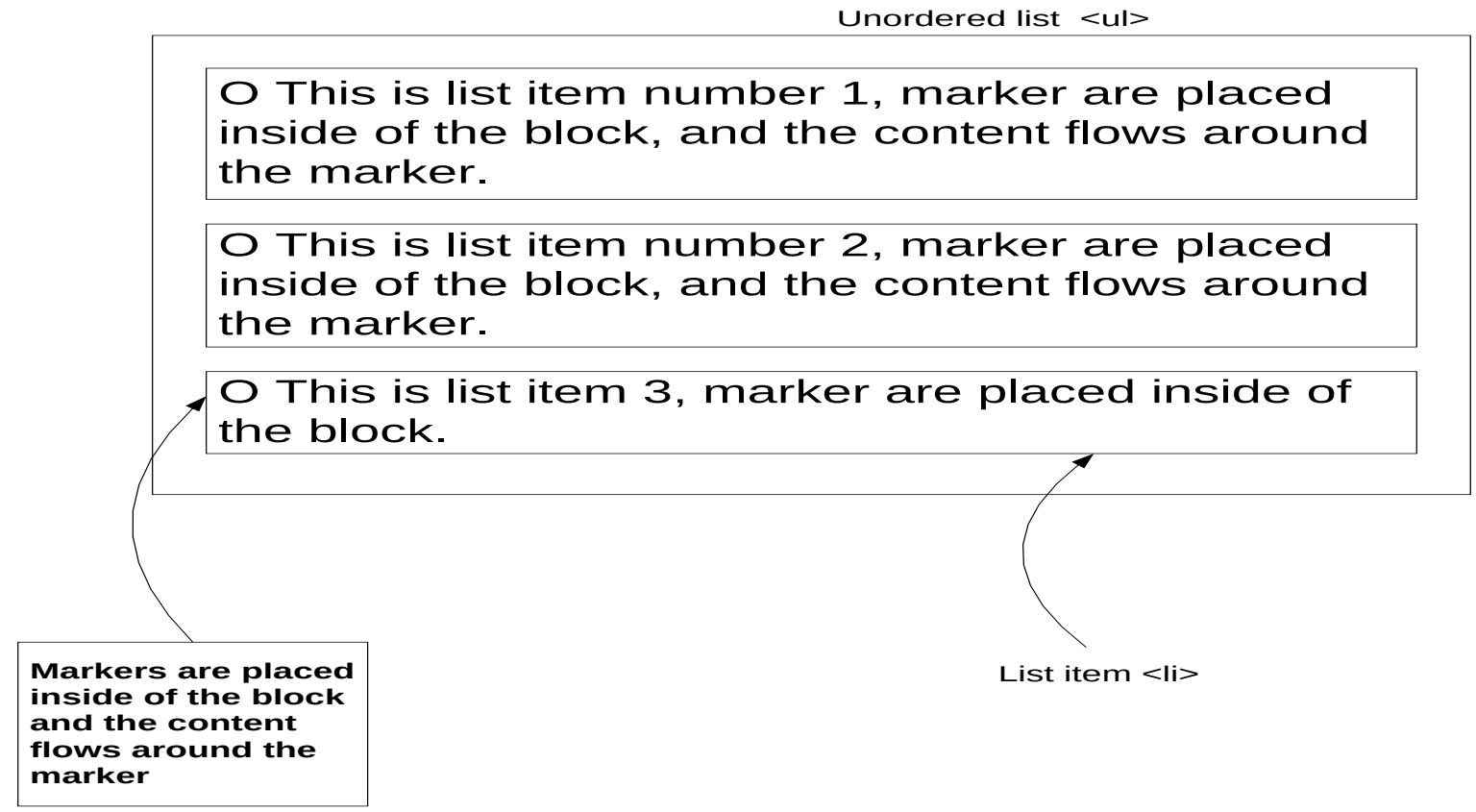
Applying a Style to a List: list-style-position

- Each list item is itself a block-level element.
- By default, most browsers place each list marker outside of its corresponding block.
- However, you can change it by using the following style:
  - List-style-position: position
  - Where position is either “outside” (the default) or “inside”.

```
<ul style=“list-style-position: inside”>
<li>Item 1</li>
<li>item 2</li>
</ul>
```

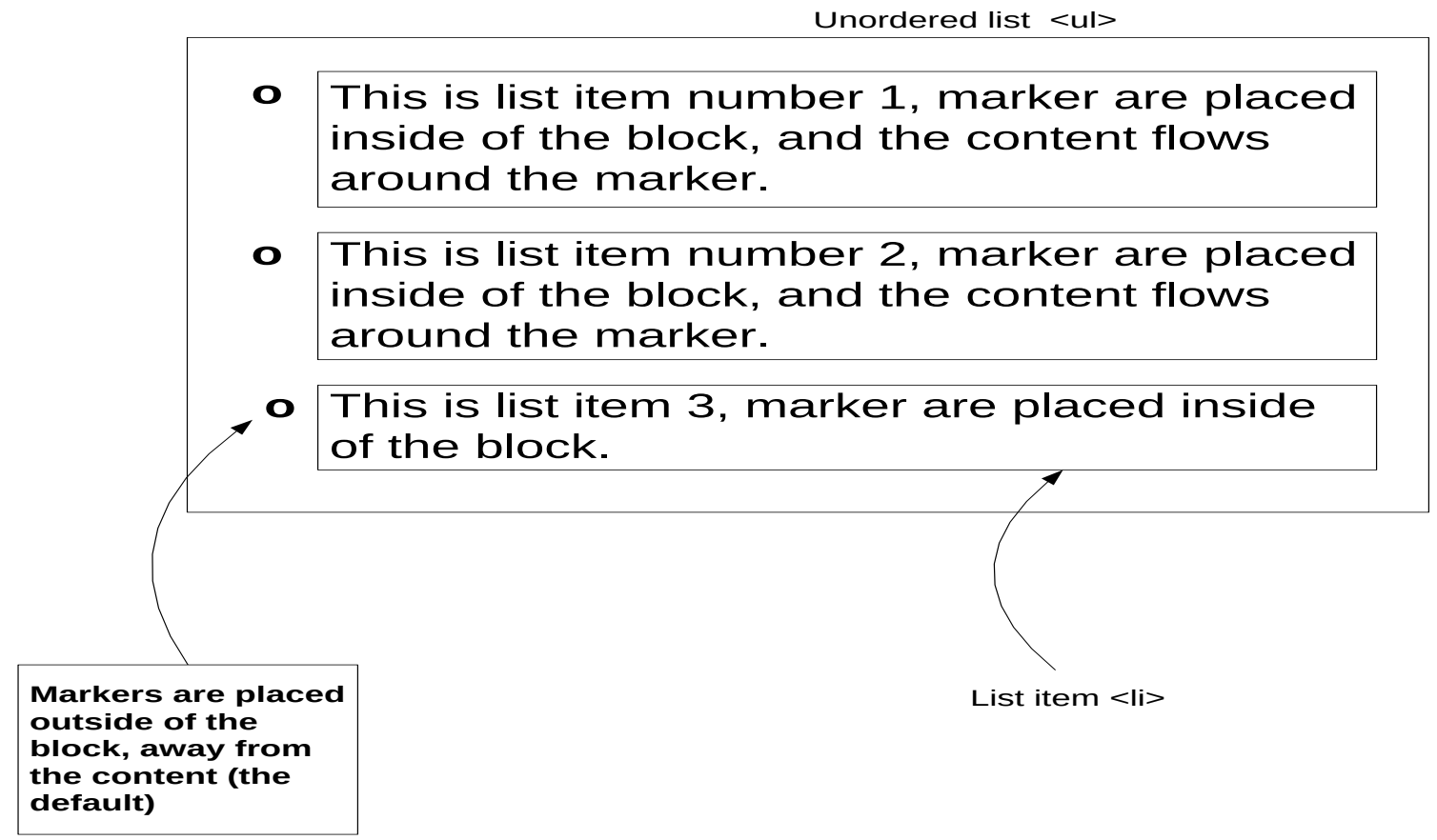
Applying a Style to a List: list-style-position

List-style-position: inside



Applying a Style to a List: list-style-position

List-style-position: outside



Applying a Style to a List

- The three previous styles can be combined in the following single style:
  - List-style: type url(filename) position
  - Where type is one of the marker type, filename is the location of a graphic file that can be used for a marker, and position is either “outside” (the default) or “inside”.

```
<ul style=“list-style: square url(a.jpg) inside”>
<li>item 1</li>
<li>item 2</li>
</ul>
```

Note: it would create an unordered list with a square marker for text-based browsers and the a.jpg image for graphical browsers, and the marker will appear inside of each list item.

Using Other Block-Level Elements

<address>…</address>	Identifies contact information	Italicized text
<blockquote>…</blockquote>	Identifies a long quotation	Plain text indented from the left and right
…	Identifies a generic block-level element	Plain text
<dl>…</dl>	Identifies a definition list	Plain text
<dt>…</dt>	Identifies a definition term (part of definition list)	Plain text
<dd>…</dd>	Identifies a definition description (part of definition list)	Plain text
<hy>…</hy>	Identifies a heading, where y is a value from 1 to 6	Boldfaced text of various font sizes

Using Other Block-Level Elements

…	Identifies an ordered list	Plain text
…	Identifies an unordered list	Plain text
…	Identifies a list item in an ordered or unordered list	Bulleted or numbered text
…	Identifies a paragraph	Plain text
<pre>…</pre>	Retains all white space and special characters in preformatted text	Fixed width text

Working with Inline Elements

- Character formatting elements are one of HTML's set of inline elements. This element allows you to format text characters.

<abbr>…</abbr>	An abbreviation	Plain text
<acronym>…</acronym>	An acronym	Plain text
…	Boldfaced text	**Boldfaced text**
<big>…</big>	Big text	Large text
<cite>…</cite>	A citation	*Italicized text*
<code>…</code>	Program code text	Fixed width text
~~…~~	Deleted text	Strikethrough text
<dfn>…</dfn>	A definition term	*Italicized text*

Working with Inline Elements

…	Emphasized content	*Italicized text*
…	Italicized text	*Italicized text*
<ins>…</ins>	Inserted text	Underlined text
<kbd>…<kbd>	Keyboard-style text	Fixed width text
<q>…</q>	Quoted text	“quoted text”
<samp>…</samp>	Sample computer code text	Fixed width text
…	Small text	Smaller text
…	A generic inline element	Plain text
…	Strongly emphasized content	**Boldfaced text**
_…	Subscripted text	Subscripted text
[…]	Superscripted text	Superscripted text

Working with Inline Elements

<tt>…</tt>	Teletype text	Fixed width text
<var>…</var>	Programming variables	*Italicized text*

Working with Empty Elements

- To display a graphic, you insert an **inline image** into the page. An **inline image** displays a graphic image located in a separate file within the contents of a block-level element.
  -
- You can insert a horizontal line by using the one-sided tag **<hr />.**
  - <hr style="color: red; background-color: red; width: 50%; height: 5" />
- A **pixel** is a dot on your computer screen that measures about 1/72" square.

Working with Special Characters

- Occasionally you will want to include special characters in your Web page that do not appear on your keyboard.

£ ®

- HTML supports the use of character symbols that are identified by a code number or name.

&code

Working with Special Characters

Symbol	Code	Name	Description
©	&#169;	&copy;	Copyright symbol
®	&#174;	&reg;	Registered trademark
•	&#183;	&middot;	Middle dot (bullet)
°	&#176;	&deg;	Degree symbol
			Nonbreaking space, used to insert consecutive blank spaces
<	&#60;	&lt;	Less than symbol
>	&#62;	&gt;	Greater than symbol
&	&#38;	&amp;	Ampersand