Hypertext Markup Language (HTML)

- **H**yperText Markup Language (HTML) is the computer language at the heart of the World Wide Web.

- HTML: Describes the content and structure of information on a web page.

- HTML is a text formatting language.

- HTML Page file end with “.htm” or “.html”

HTML History

Year	Version
1989	Tim Berners-Lee invented www (World Wide Web)
1991	Tim Berners-Lee invented HTML
1993	Dave Raggett drafted HTML+
1995	HTML Working Group defined HTML 2.0
1997	W3C Recommendation: HTML 3.2
1999	W3C Recommendation: HTML 4.01
2000	W3C Recommendation: XHTML 1.0
2008	WHATWG HTML5 First Public Draft
2012	WHATWG HTML5 Living Standard
2014	W3C Recommendation: HTML5
2016	W3C Candidate Recommendation: HTML 5.1

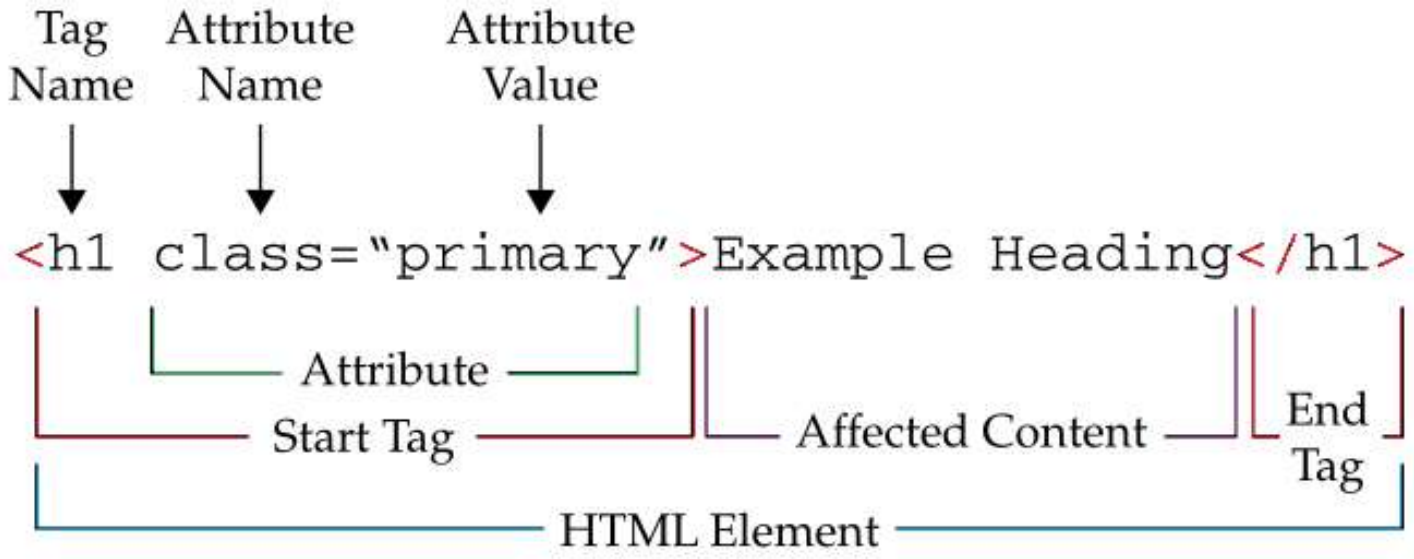
- **W3C : World Wide Web Consortium**
- **WHATWG : The Web Hypertext Application Technology Working Group**

HTML Basic Structure

- An HTML document is made up of **elements.**
- HTML elements are written with a **start** tag, with an **end** tag, with the **content** in between.
  - syntax: < > </ >
  - Example: < tagname >content</ tagname >
    - This is a paragraph
- Nothing within the brackets < > will be displayed by the browser (unless the HTML is incorrectly written and the browser interprets the tags as part of the content).
- most whitespace is insignificant in HTML (ignored or collapsed to a single space).
- Tags are not case sensitive.

HTML Basic Structure *Cont'd*

- Most of the **HTML tags** have **attributes**.
- An **attribute** is used to define the characteristics of an **HTML element** and is placed inside the **element's** opening **tag.**
- The **element** can be consisted of an **attribute/value** pair.



Tags in HTML

- There are two types of markup tags:
  - **Container tags** – Define a section of text using a start tag and an end tag. To create an end tag for a start tag, you insert a forward slash (/) in front of the tag's name. Thus, a start tag has the form <tagname [attributes]> and an end tag has the form </tagname>. The For example, text placed inside of these tags would appear in bold:
    - <B>Hello</B>
  - **Empty tags** – (or Standalone) represent a single occurrence of an instruction. For example, the <BR> or break tag is used to indicate that you want to include a single space following the text.

Example of container and empty tags (Standalone)

Container tags

```
<html>
<head>
<title> The Web Page's Title </title>
</head>
<body>
Text content and body tags inserted here
</body>
</html>
```

**Empty tags (standalone)
**

```
<html>
<head>
<title> Example of the line break tag
</title>
</head>
<body>
This text is displayed on line one
<br>This text is displayed on line two
</body>
</html>
```

Comments in HTML

- It's often useful to embed comments in your documents.
- The comments are not displayed in a browser, but they can be read when viewing the source markup.
- Comments in HTML use a specialized tags <!-- --> as the following:
  - <!-- comments .. comments .. comments -->
- Web browsers will not render any content or elements that occur between those markers.
- Comments can also be useful to temporarily "hide" portions of markup for testing your web pages.

HTML Documents

- The DOCTYPE declaration defines the document type to be HTML.
- he HTML document itself begins with **<html>** and ends with **</html>**.
- There are two sections in the document: HEAD and BODY, marked by <**head**>...</**head**> and <**body**>...</**body**> tags, respectively.
- The head section contains information about the HTML document.
- The visible part of the HTML document is between **<body>** and **</body>**.

```
<!DOCTYPE html>
<html>
<head>
 <meta charset="utf-8">
 <title>Basic HTML Document Layout</title>
</head>
<body>
<h1>My First Web Page</h1>
<hr>
<p>This is my <strong>first</strong>
web page written in HTML.</p> <h3>HTML</h3>
 <p>HTML uses <em>markup tag
</em> to <em>markup</em> a document.</p>
</body>
</html>
```

Head Section

Bo
Secti

HTML Documents


```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>Basic HTML Document Layout</title>
</head>
<body>
<h1>My First Web Page</h1>
<hr>
<p>This is my <strong>first</strong>
web page written in HTML.</p>
<h3>HTML</h3>
 <p>HTML uses <em>markup tag
</em> to <em>markup</em> a document.</p>
</body>
</html>
```

In the Head section

< head > …. </ head >

In the HEAD section: <head>

- The following elements can used inside the <head> element:
  - <meta> <title> <style> <base> <link> <script> <noscript>

- The <meta> tag provides metadata about the HTML document such as:
  - <meta charset="utf-8"> element specifies the character encoding scheme of the document.
    - IT helps the browser determine how to render the content.
    - Today, virtually all (English) HTML documents are encoded using the UTF-8 character encoding scheme, which is compatible with ASCII code for English alphabets and allow you to include other Unicode characters (such as Chinese, Japanese and Korean) efficiently.

In the HEAD section: <head>

2. <meta name="keywords" content="HTML, CSS, XML, XHTML, JavaScript"> defines keywords for search engines.
3. <meta name="description" content="Free Web tutorials on HTML and CSS"> defines a description of your web page.
4. <meta

```
<head>
<meta charset="UTF-8">
<meta name="description" content="Free Web tutorials">
<meta name="keywords" content="HTML,CSS,XML,JavaScript">
<meta name="author" content="Hege Refsnes">
</head>
```

seconds.

In the HEAD section: <head> *Cont'd*

- The <title>...</title> element provides a descriptive title to the page. The browser displays the title on the title-bar of the tab/window.
  - The title identifying a page when users add the page to their list of **Favorites** or **Bookmarks**, enabling them to return to their favorite sites.
  - Search engines use the title for indexing purposes and when displaying results.
- The <base> specify a default URL and a default target for all links on a page.

```
<head>
<base href="http://www.w3schools.com/images/" target="_blank">
</head>

<body>
<img src="stickman.gif" width="24" height="39" alt="Stickman">
</body>
```

➢ How about the other elements <style> <link> <script> <noscript>????

In the BODY section

<body> …. </body>

HTML Headings

- HTML provides six **heading elements** (h1 through h6) for specifying the *relative importance* of information.

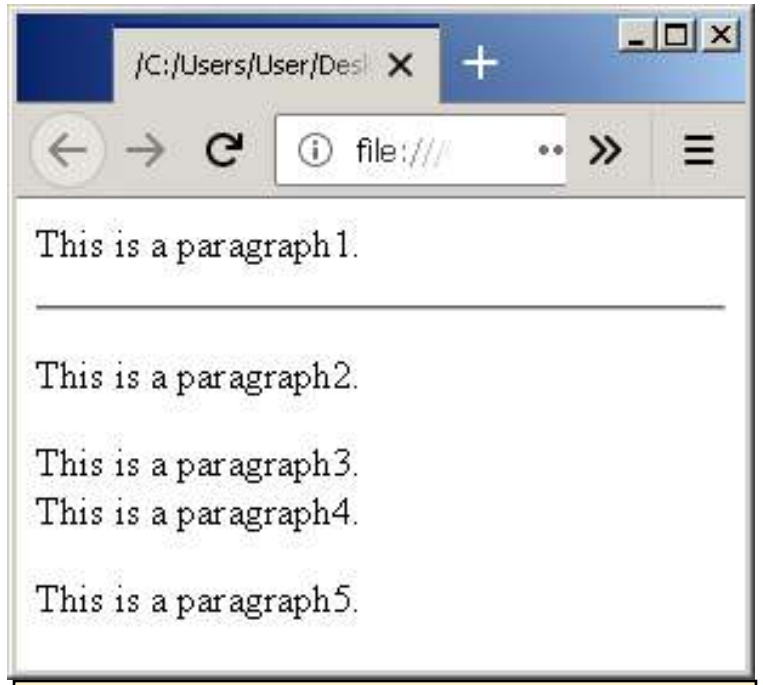
Headings Are Important??

- HTML uses headings for **headings only**. Don't use headings to make text BIG or bold.
- Search engines use your headings to index the structure and content of your web pages.
- Users browse your pages by its headings. It is important to use headings to show the document structure.
- h1 headings should be main headings, followed by h2 headings, then the less important h3, and so on.


```
<body>
<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<h3>This is heading 3</h3>
<h4>This is heading 4</h4>
<h5>This is heading 5</h5>
<h6>This is heading 6</h6>
</body>
```

HTML Paragraphs

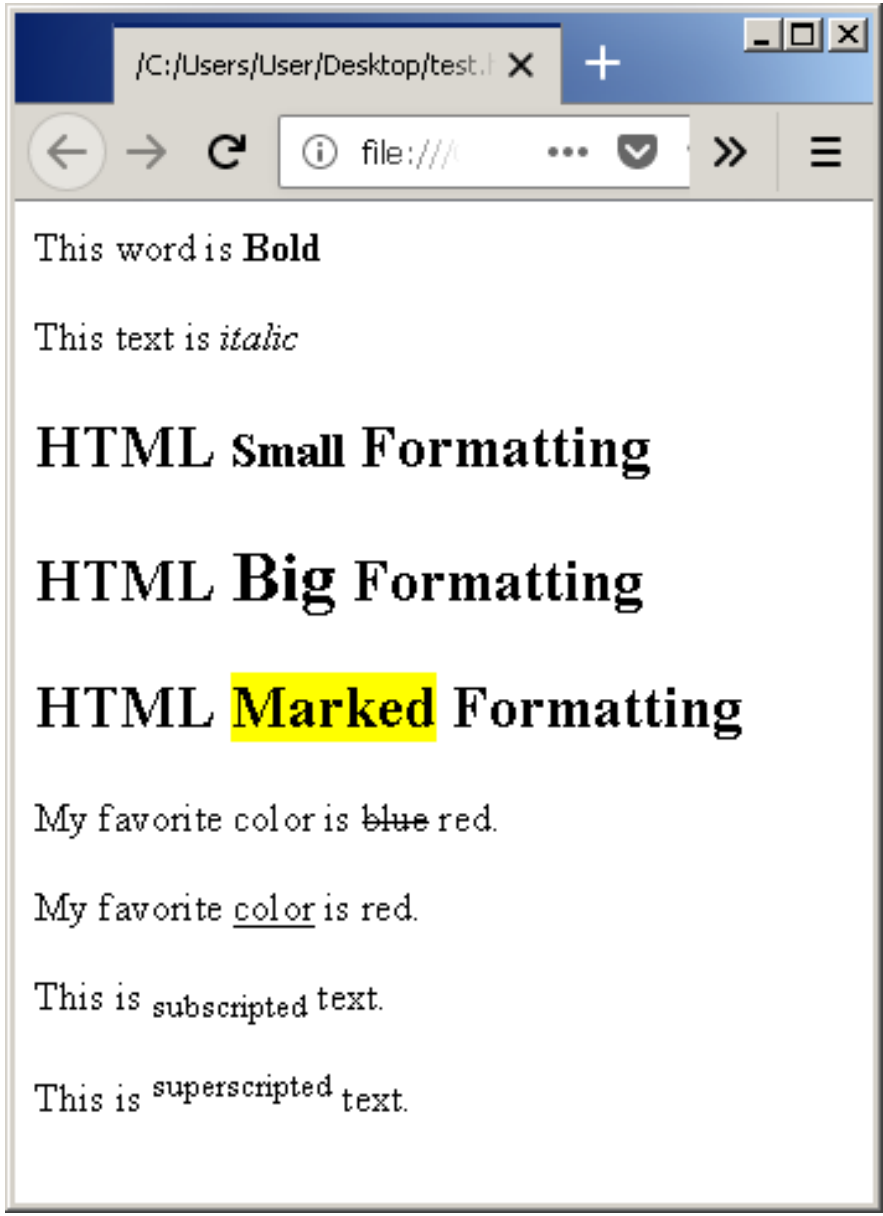
- Paragraph Element (...)
  - All the text placed between the and tags forms one paragraph.
  - When a browser renders a paragraph, it places extra space above and below the paragraph text.
- Line Break <br>: Instruct the browser to break to a new line.
- Horizontal Rule <hr> : is a standalone element, draws a horizontal rule (or line). The rule is full width (100%) across the screen.
- Pre-Formatted Text <pre>...</pre>
  - Container tags are treated as pre-formatted, i.e., white space, tabs, new-line will be preserved and not ignored.
  - The text is usually displayed in a fixed-width font. <pre>...</pre> is mainly used to display program codes or Poem.




```
<body>
 <p>This is a paragraph1.</p>
<hr>
<p>This is a paragraph2.</p>
<p>This is a paragraph3.
 <br>This is a paragraph4.</p>
<p>This is a paragraph5.</p>
</body>
```

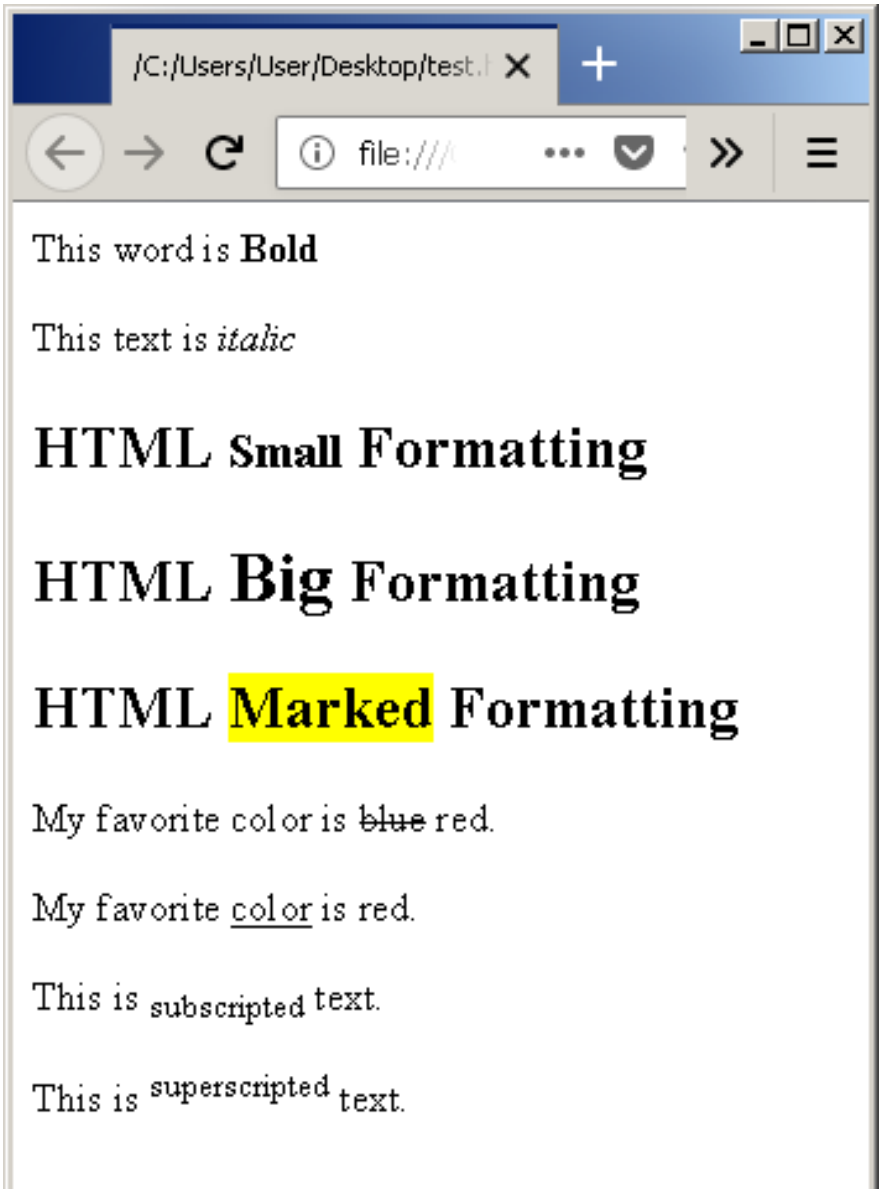
Text Formatting

- Bold: The ... tags specify "strong emphasis" for its content - rendered in bold by the browser. While, ... element defines **bold** text, without any extra importance.
- Italic: The nested ... tags specify "emphasis" - rendered in *italic* by the browser. The tags .. define *italic* text, without any extra importance.
- A smaller text: <small>...</small> element defines small text.
- A Bigger text: <big>...</big> element defines big text.
- Marked or Highlighted text : <mark>...</mark>.
- Deleted (removed) text: <del>... </del>.
- Underline: <u>...</u>.



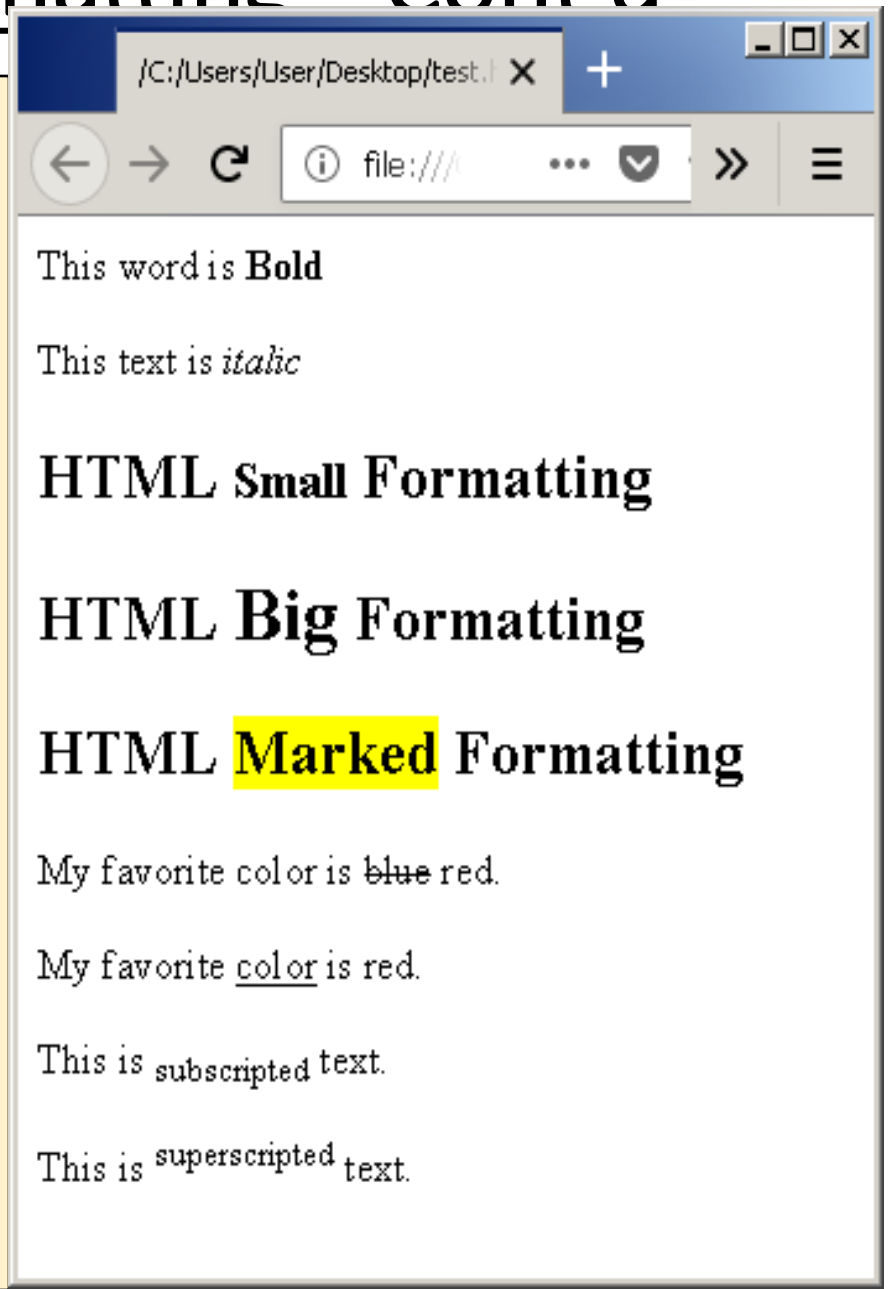
Text Formatting Cont'd

- A Bigger text: <big>…</big> element defines big text.
- Marked or Highlighted text : <mark>…</mark>.
- Deleted (removed) text: <del>… </del>.
- Underline: <u>…</u>.
- Subscripted Text: _{subscripted}
- Superscripted Text: ^{superscripted}



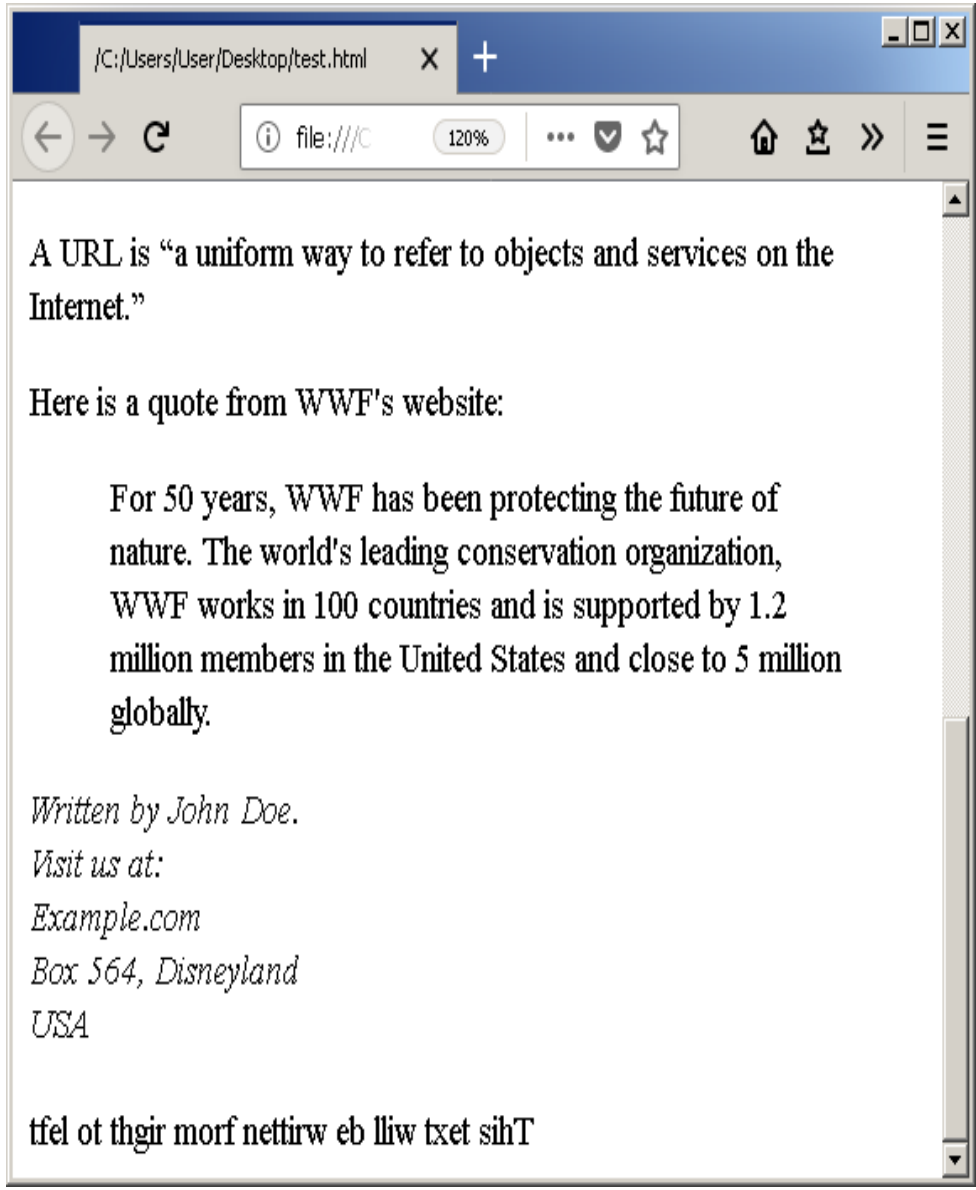
Text Formatting Cont'd

```
<body>
<p>This word is <strong>Bold</strong></p>
<p>This text is <em>italic</em></p>
< h2 >HTML <small>Small</small> Formatting</h2>
< h2 >HTML <big>Big</big> Formatting</ h2 >
< h2 >HTML <mark>Marked</mark> Formatting</ h2 >
<p>My favorite color is <del>blue</del> red.</p>
<p>My favorite <ins>color</ins> is red.</p>
<p>This is <sub>subscripted</sub> text.</p>
<p>This is <sup>superscripted</sup> text.</p>
</body>
```



Quotation and Citation Elements

- Short Quotations: <q>…</q> , browsers usually insert quotation marks "…." around the <q> element.
- Long Quotations: <blockquote>… </blockquote>
- Contact Information: <address>… </address>, it defines contact information (author/owner) of a document or article.
  - The text in the <address> element usually renders in *italic*. Most browsers will add a line break before and after the address element.
- BiDirectional Override: <bdo>…</bdo>, it used to override the current text direction.
  - Required: Specifies the text direction of the text inside the <bdo> element
  - <bdo dir="rtl"> …</bdo> (right to left)
  - <bdo dir="ltr"> … </bdo> (left to right)



Block Elements

- The <header>...</header> and <footer>...</footer> elements can be used to markup the header and footer of a web page.
- The <section>...</section> element can be used to markup each content section in a document (such as each chapter of the book).
  - <section id = "1">…</section>
  - <section id = “2">…</section>
- The <article>...</article> element is used to markup an independent and self-contained article such as a news story, which could have its own header, footer and content sections.
- Main Content <main>...</main> it marks the main
  - content of a web page, excluding the header and footer. There shall NOT be more than one <main> element in a document.

<body>
<header>.....</header>

<main>

<article>.....</article>

<section>.....</section>

<section>.....</section>

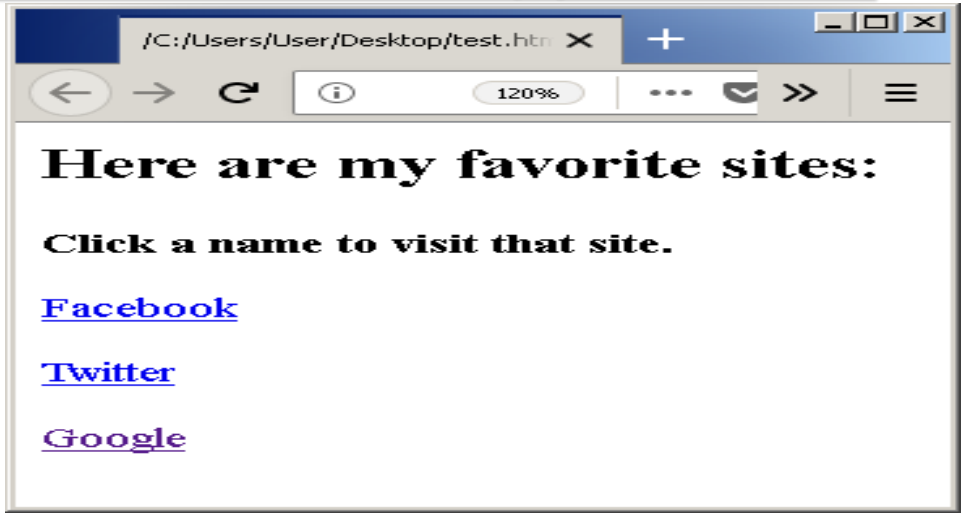
<article>.....</article>

</main>

<footer>.....</footer>
</body>

Anchors and Hyperlinks

- hyperlink, is a reference (or links to) other resources, such as HTML documents and images.
- Web browsers typically underline text hyperlinks and color their text blue by default so that users can distinguish hyperlinks from plain text.
- The browser changes the color of any text link once you've clicked the link.




```
<body>
<h2>Here are my favorite sites:</h2>
<p><strong>Click a name to visit that site.
</strong></p>
<p><a href =
"www.facebook.com">Facebook</a></p>
<p><a href =
"www.twitter.com">Twitter</a></p>
<p><a href =
"www.google.com">Google</a></p>
</body>
```

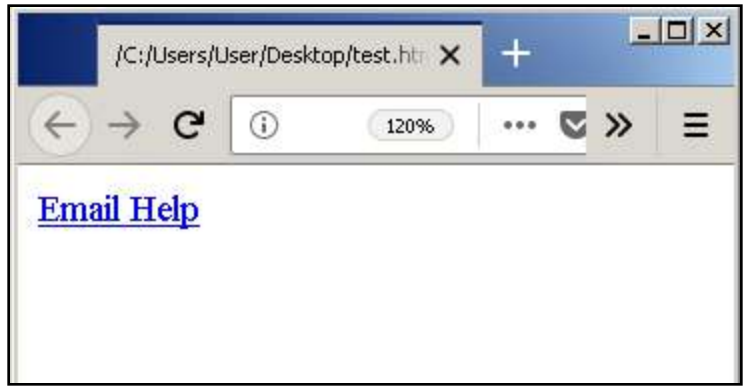
Anchors and Hy

- links are defined with the **<a>..** tag.
  - *link text*
- Example :
  - Google
- The **href** attribute specifies the destination address.
- The *link text* is the visible part (Google).
- *hyperlink* to the URL assigned to attribute **href** , which specifies a location, such asresource's
  - ➢ a web page or location within a web page.
  - ➢ a file.
  - ➢ an e-mail address.


```
<body>
<h2>Here are my favorite sites:</h2>
<p><strong>Click a name to visit that site.
</strong></p>
<p><a href =
"www.facebook.com">Facebook</a></p>
<p><a href =
"www.twitter.com">Twitter</a></p>
<p><a href =
"www.google.com">Google</a></p>
</body>
```

Hyperlinking to an E-Mail Address

- Anchors can *link to e-mail addresses* using a **mailto:**URL.
- When the user clicks this type of anchored link, most browsers launch the user's default e-mail program (for example, Microsoft Outlook or Apple Mail).
- Email Help




```
<body>
<a href="mailto:help@zzz.com">
Email Help</a>

</body>
```

Link to a Specific HTML Element

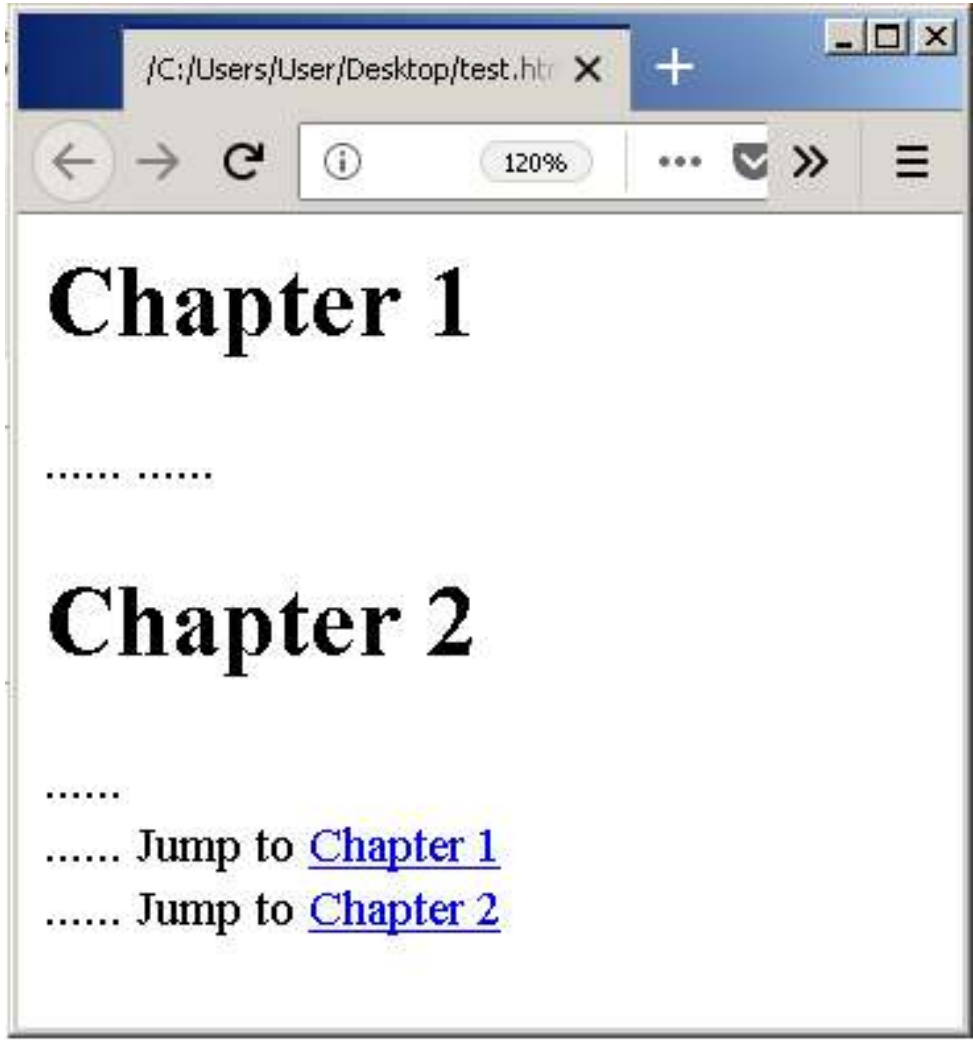
- You can setup a link to target a specific HTML element (typically a heading <h1> to <h6>), similar to bookmark, via:
- Define an id="idName" attribute for the targeted element.
- The id attribute is applicable to ALL HTML elements, including <h1> to <h6>.
- The id-value is supposed to be **unique** in the document (i.e., no two elements shall have the same id-value).

An anchor name (or bookmark) called idname will be set up automatically on the element.

- You can refer to the anchor point via #idName, by prefixing with a # sign.
- Setup a link targeting the anchor point as the following:
  - ... , for the same document.
  - ... , for the anchor point in another document identified via url.

Link to a Specific HTML Element Cont'd

```
<body>
<h1 id="ch1">Chapter 1</h1>
  ......
  ......
<h1 id="ch2">Chapter 2</h1>
  ......<br>
  ......
Jump to <a href="#ch1">Chapter 1</a><br>
  ......
Jump to <a href="#ch2">Chapter 2</a>
</body>
```



Image

- **Image Tag <img>**

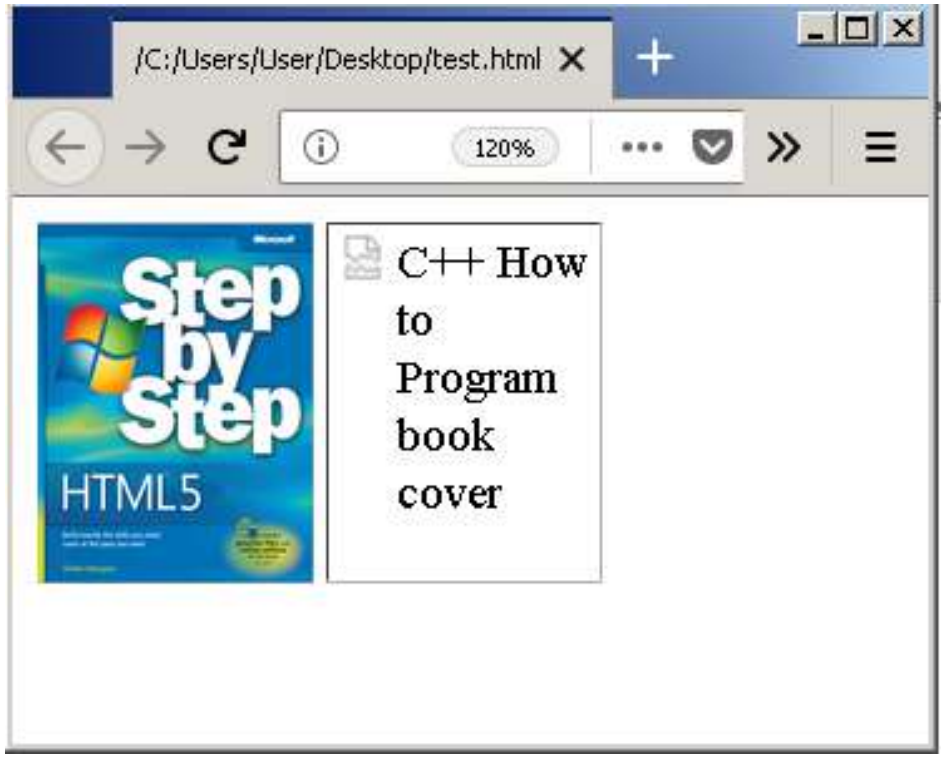
Example: <img src="logo.gif" alt="Logo" width="200" height="100">

Attributes:

- src="imageUrl" (required): gives the URL of the image.
- alt="text" (required): alternative text to be displayed if the image cannot be displayed.
- width="n|n%", height="n|n%": specify the width and height of the image display area (in pixels or percentage).
  - Browsers use these values to reserve space for the image (before the image is downloaded) and continue rendering the rest of the contents.
  - Using of the width and height tags is recommended for images, so that the browser can reserve spaces for the images. This is more efficient and could avoid a jerky display if you page contains many images.
  - You can find out the width and height of an image easily by checking the "Properties" of the image.

Image example

```
<body>

<img src = "ebook1.png" width = "92" height
= "120"
 alt = "Step by Step HTML5">

<img src = “www.amazon.com/ebook2.png"
width = "92" height = "120"
alt = "C++ How to Program book cover">

</body>
```



- This first image is located in the *same* directory as the HTML document, so only the image’s file name is required.
- The second image is not available, thus the alternative text was displayed.

Using Image as Hyperlink

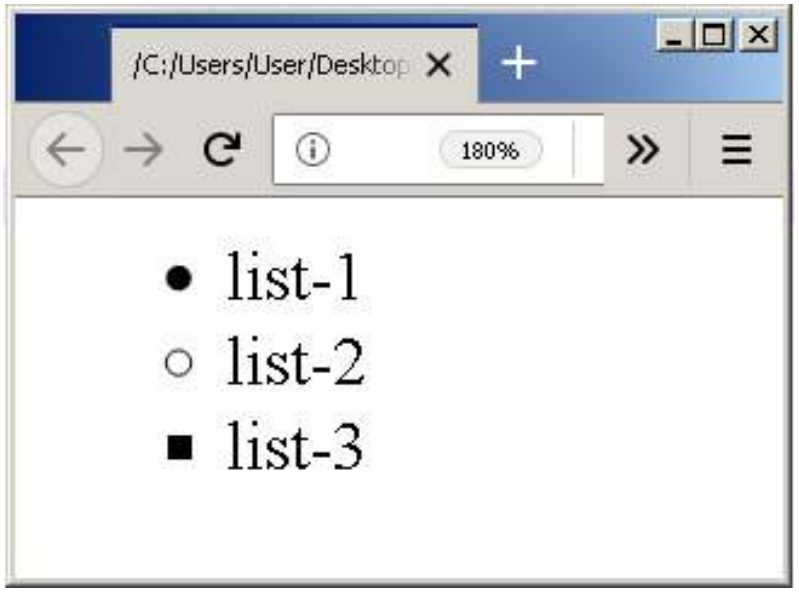
- To use an image as a hyperlink, put the image tag <img> between and .
- For example:

```
<body>

<a href="http://abc.com/">
<img src="ebook1.png"  width="120“ height="160"
alt = "Step by Step HTML5">
</a>
<p>click the above image to visit us</p>

</body>
```

- List-related tags are meant for marking up a list of items.
- HTML supports three types of lists:
  1) **Ordered list**
  2) **Unordered list**
  3) **Definition list**.

```
<body>
<ul>
<li type="disc" >list-1</li>
<li type=" circle " >list-2</li>
<li type=" square " >list-3</li>
</ul>
</body>
```

1. **Unordered list** is shown with a bullet in front of each item. The <ul>... contains an unordered list. Each of items in the list is enclosed in <li>...</li.



✓An attribute type in <ul> tag is used to choose the style of the bullets:
- type="disc": a black dot (default).
- type="circle": an empty circle.
- type="square": a filled square.

Lists Cont'd

✓ A **style** attribute can be also added to an **unordered list**, to define the style of the marker as the following:

```
<ul style="list-style-type:disc">
```

Style	Description
list-style-type:disc	The list items will be marked with bullets (default)
list-style-type:circle	The list items will be marked with circles
list-style-type:square	The list items will be marked with squares
list-style-type:none	The list items will not be marked

Lists Cont'd

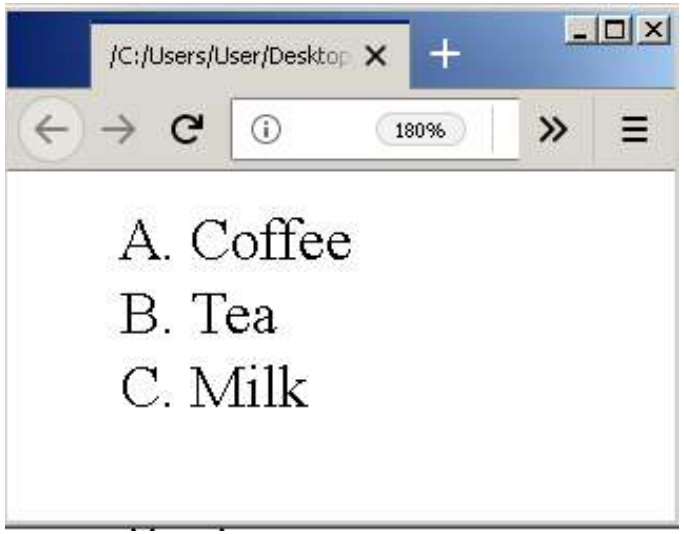
2. **Ordered List** ... Items in an ordered list are numbered automatically by the browser. Each item of the list is contained inside a <li>... container tag. The syntax is similar to the unordered list.

```
<body>
<ol type="A">
<li>Coffee</li>
<li>Tea</ li >
<li>Milk</li >
</ol>
</body>
```

The attribute start="number" can be used in the <ol> tag to specify the starting number (which default to 1).

The type attribute of the <ol> tag can be used to choose the numbering style:

- ❑ type="1": numbers 1, 2, 3, ... (default)
- ❑ type="a": lowercase letters a, b, c, ...
- ❑ type="A": uppercase letters A, B, C, ...
- ❑ type="i": lowercase Roman numerals i, ii, iii, iv, ...
- ❑ type="I": uppercase Roman numerals I, II, III, IV, ...



Lists Cont'd

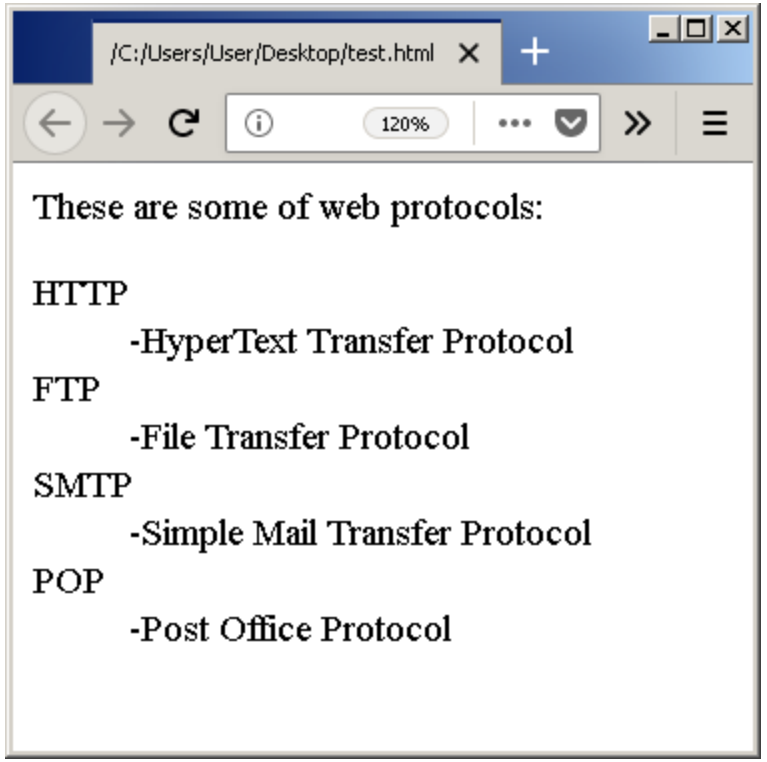
3. **Description Lists** is a list of terms, with a description of each term.

The <dl> … </dl> tag defines the description list.

The <dt>… </dt> tag defines the term (name).

The <dd>… </dd> tag describes Detail.

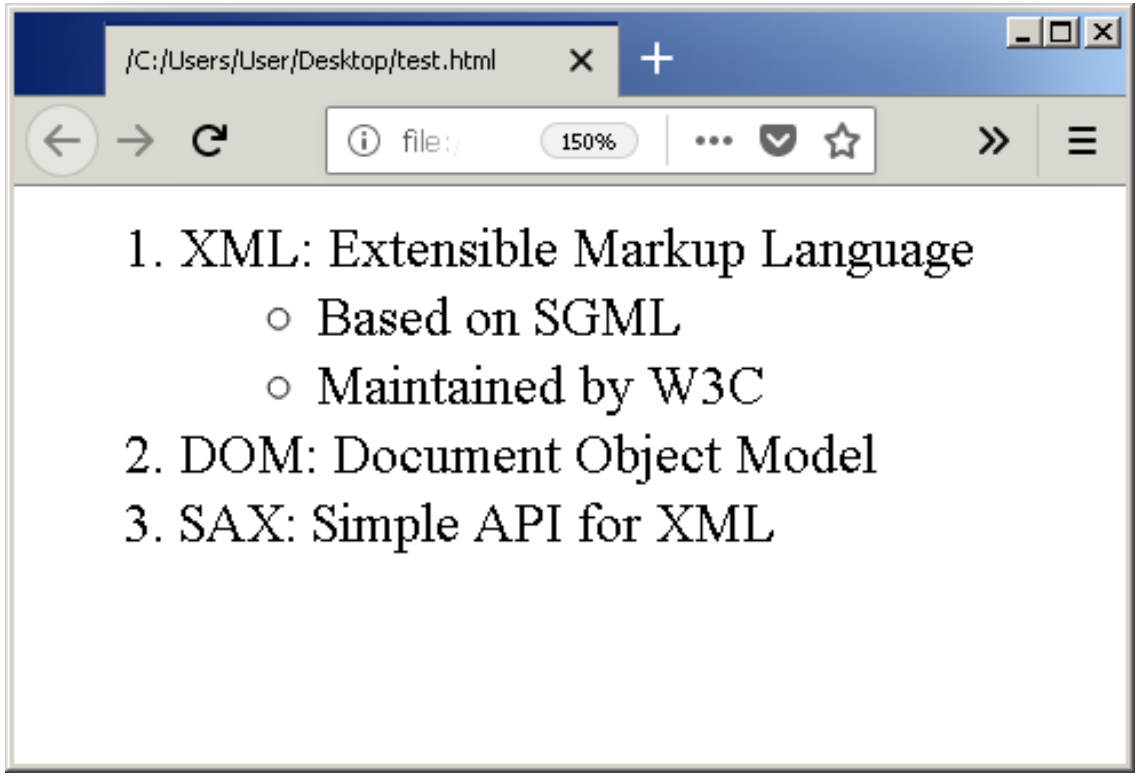
```
<body>
<p>These are some of web protocols:</p>
<dl>
<dt>HTTP</dt><dd>-HyperText Transfer Protocol</dd>
<dt>FTP</dt><dd>-File Transfer Protocol</dd>
<dt>SMTP</dt><dd>-Simple Mail Transfer Protocol</dd>
<dt>POP</dt><dd>-Post Office Protocol</dd>
</dl>
</body>
```



Nested Lists

Lists can be *nested* to represent *hierarchical* relationships, as in a multilevel outline (place a list inside another list).

```
<body>
<ol>
<li>XML: Extensible Markup Language
<ul>
<li>Based on SGML</li>
<li>Maintained by W3C</li>
</ul>
</li>
<li>DOM: Document Object Model</li>
<li>SAX: Simple API for XML</li>
</ol>
</body>
```



Reserved & Special Characters

Character	Entity Reference
"	&quot;
<	&lt;
>	&gt;
&	&amp;
non-breaking space	
→ ⇒ ↔ ⇔	→ ⇒ ↔ ⇔
° (degree)	&deg;
© ® € ¢ ¥	© ® € ¢ ¥
~	˜
× ± ∞	× ± ∞
π Π σ Σ ω Ω	π Π σ Σ ω Ω
≥ ≤ ≡ ≈	≥ ≤ ≡ ≈
⊂ ⊃ ⊆ ⊇ ∈	⊂ ⊃ ⊆ ⊇ ∈

Lec-4+5

- Introduction To **C**ascading **S**tyle **S**heets (**CSS**)

Introduction to Cascading Style Sheets

- A *Style Sheet* is *a collection of style rules* that can be applied to *a selected set of HTML elements*.

- A style rule is used to control the appearance of HTML elements such as their **font properties** (e.g., type face, size and weight), **color properties** (e.g., background and foreground colors), **alignment**, **margin**, **border**, **padding**, and **positioning**.
- The word *cascading* means that multiple style rules can be applied to the same HTML element.

CSS Specifications

- The W3C (World-Wide Web Consortium) defines three CSS levels:

1) CSS Level 1 (December 1996): CSS1 laid the ground work and introduced the selectors and most of the properties.
2) CSS Level 2 (May 1998) and CSS Level 2.1 (Last revised on June 2011): CSS2 added new features such as targeting devices and printers, and absolute positioning.
3) CSS Level 3: CSS3 is not a single piece of specification. As CSS grows, W3C decided to break it into modules, such as the Selectors module, the Values and Units Modules, the Box Alignment module, and so on. Each module is then developed independently. The CSS3 Selectors module (@ http://www.w3.org/TR/selectors/) and CSS3 Colors module (@ http://www.w3.org/TR/css3-color/).

First Example

A CSS style sheet provides style rules to HTML documents. The example below shows how to apply style attribute to an HTML element <h1>.

```
<body>

<h1 style="color:blue;"> This is a Blue h1 Heading</h1>
```

The Way to Insert CSS into HTML

- There are three places where you can define style rules for CSS:

1) Inline style- using a **style attribute** in HTML elements. The rules are applicable to that particular HTML element only.

2) Internal style- (Embedded) - using a **<style>…</style>** element in the HTML <head> section. The styles are applicable to that entire document.

3) External style- using one or more **external CSS files**, which is then linked to HTML documents via a <link> element in the HEAD section. The same external style sheet can be applied to all HTML pages in your website to ensure uniformity in appearance.

1- Inline Style

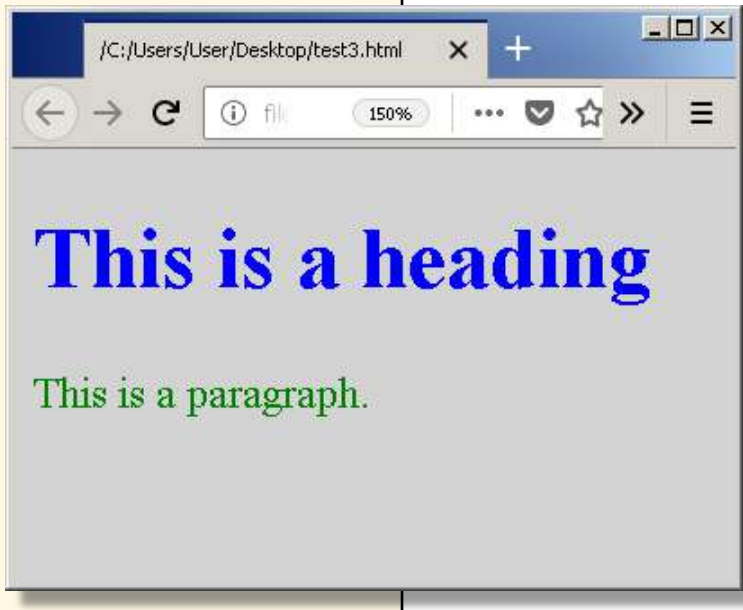
- **Inline styling** is used to apply a unique style to a single HTML element:
- Inline styling uses the **style** attribute.
- This example changes the text color of the <h1> element to blue:

```
<!DOCTYPE html>
<html>
<body>

<h1 style="color:blue;">This is a Blue h1 Heading</h1>
<h1>This is an h1 Heading</h1>

</body>
<html>
```

2- Internal style

- Internal styling is used to define a style for one HTML page.
- **Internal styling** is defined in the **<head>** section of an HTML page, within a **<style>** element:

```
<!DOCTYPE html>
<html> <head>
<style>
        body {background-color:lightgrey;}
        h1   {color:blue;}
        p    {color:green;}
</style>
</head>

<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```



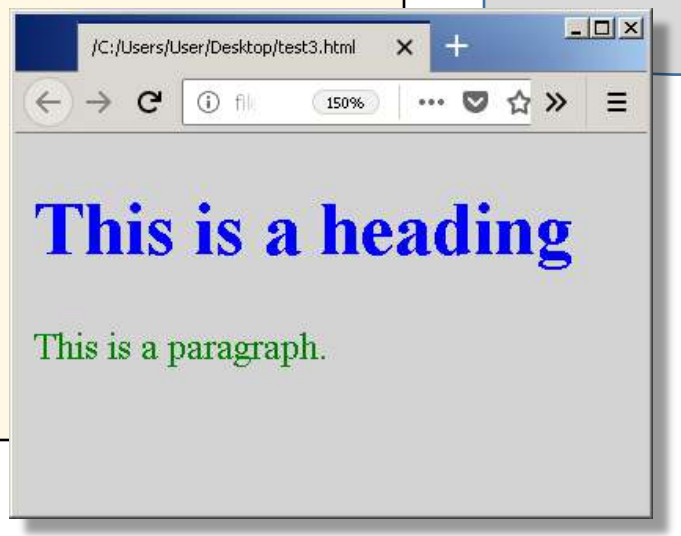
3- External style

- An external style sheet can be written in any text editor.
- The file should not contain any html tags. The style sheet file must be saved with a .css extension.

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

<body>
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
</body>
</html>
```

```
body { background-color: lightgrey; }

h1 { color: blue; }

p { color:green; }
```

style sheet file
"styles.css"



3- External style *Cont'd*

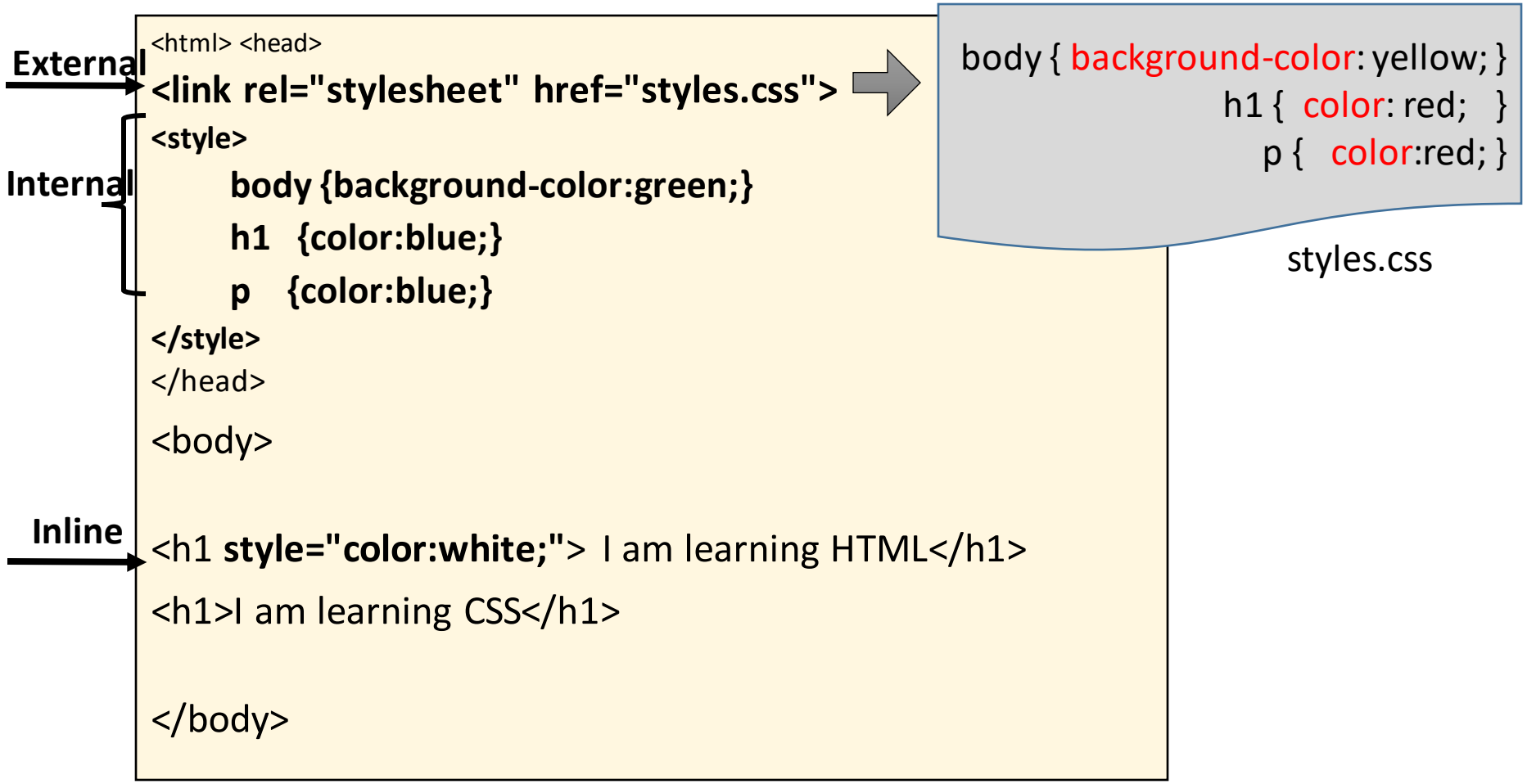
- Linking to External Style Sheet using CSS @import Directive.
- Besides the HTML <link> element, you can use CSS's @import directive to link to an external style sheet, as follows:

```
<style>
@import url("cssURL1.css");
@import url("cssURL2.css");
</style>
```

@import is part of the CSS language. It can also be used in one CSS file to include rules from another CSS file.

Priority in CSS

- Inline styles have the highest priority followed by internal styles and followed by external styles as their ordered in the head section.



Priority in CSS

- Inline styles have the highest priority followed by internal styles and followed by external styles as their ordered in the head section.

External

Internal

Inline

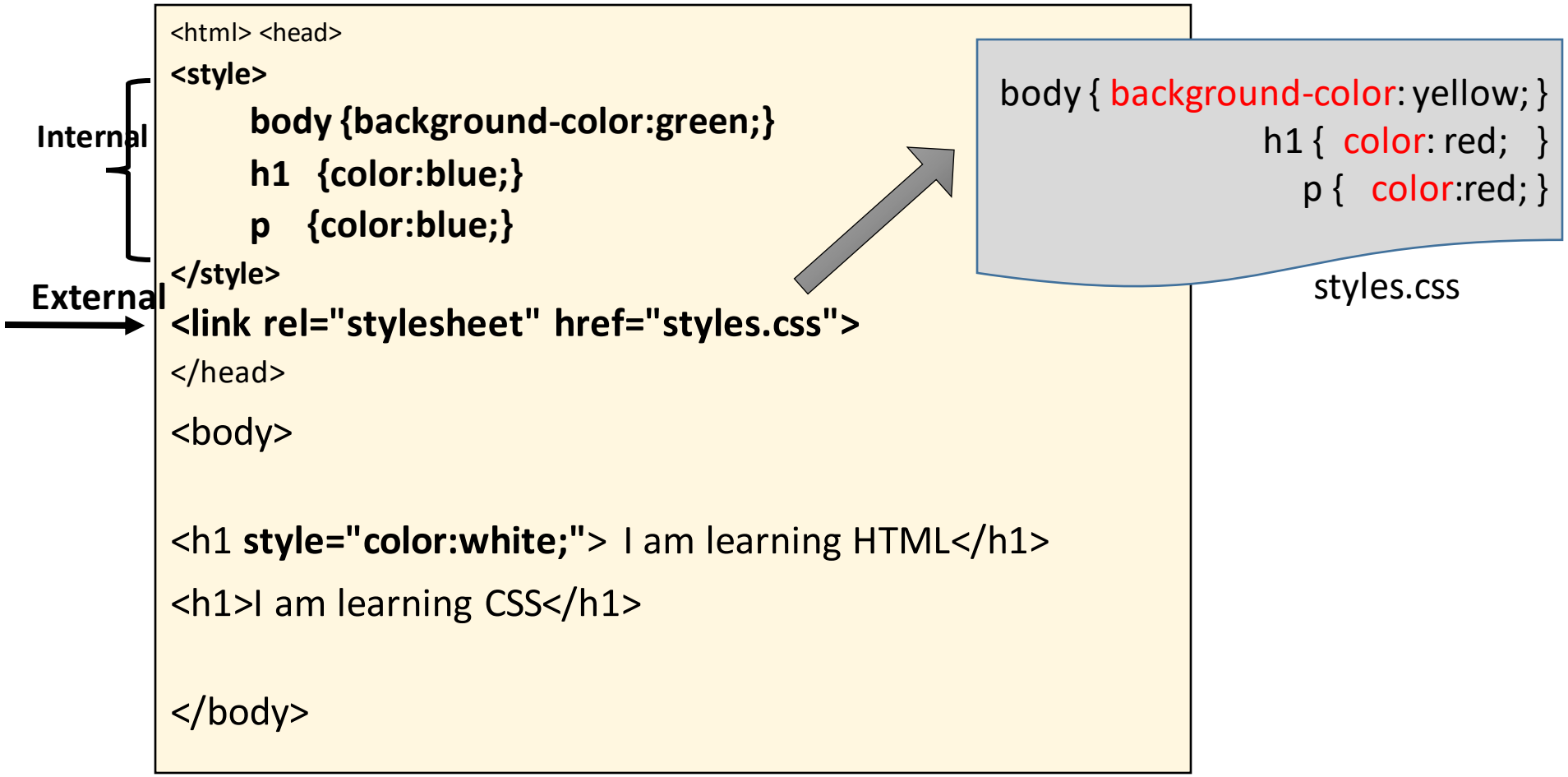
```
<html> <head>
<link rel="stylesheet" href="styles.css">
<style>
        body {background-color:green;}
        h1   {color:blue;}
        p    {color:blue;}
</style>
</head>
<body>

<h1 style="color:white;"> I am learning H
<h1>I am learning CSS</h1>

</body>
```

```
body { background-color: yellow; }
h1 {  color: red;   }
p {   color:red; }
```

styles.css

Priority in CSS *Cont'd*

- What is the output in this case?



Inheritance in CSS

- Inheritance means that styles are inherited from their parent elements. Consider the following:

```
<body>
<h1>My header</h1>
<p>Subsequent  Text</p>
</body>
```

- Both the <h1> and <p> tags are considered children of the <body> tag.
- The styles you give to the <body> will be inherited by the children until you make another rule that overrides the inherited style.
- Some properties (such as border, margin, padding, width, height, background-color) are not inherited.

```
body {
    background-color: yellow;
}
p {
background-color: inherit;
}
```

ex. Inheritance properties

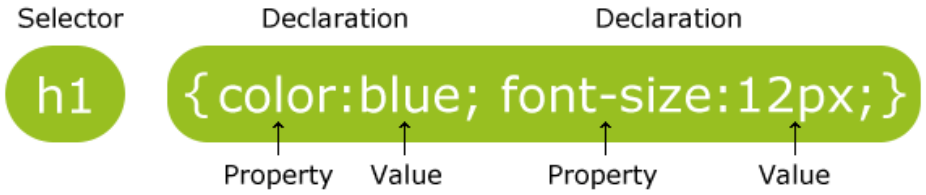
- border-collapse
- border-spacing
- caption-side
- color
- cursor
- direction
- empty-cells
- font-family
- font-size
- font-style
- font-variant
- font-weight
- font-size-adjust

- line-height
- list-style-image
- list-style-position
- list-style-type
- list-style
- orphans
- quotes
- tab-size
- text-align
- text-align-last
- text-decoration-color
- text-indent

- text-justify
- text-shadow
- text-transform
- visibility
- white-space
- widows
- word-break
- word-spacing
- font-stretch
- font
- letter-spacing

CSS Syntax in General

- CSS has its own syntax, which is totally different from HTML
- A CSS rule-set consists of a selector and a declaration block:



- The selector points to an HTML element. In the example above the selector is **h1** tag.
- The declaration block contains one or more declarations separated by semicolons "**;**".
- Each declaration includes a CSS property name and a value (color**:**blue), separated by a colon "**:**".
- A CSS declaration always ends with a semicolon "**;**", and declaration blocks are surrounded by curly braces "**{ }**".

CSS Syntax *Cont'd*

Example2 :

```
h1, h2, h3, h4 { text-align: center; font-family: "Trebuchet MS", "Segoe UI",
                 Helvetica, Tahoma;
                 margin: 10px auto 10px auto;}
```

- Multiple values for the same property name are separated by commas "," (as in the font-family).
- multiple parts of the same property value are separated by space " " (as in the margin, which has a value with 4 parts).
- Values containing space must be quoted, e.g., " Segoe UI" or 'Segoe UI'
- If the same set of styles is applicable to more than one elements, the selectors can be grouped together in one single rule (e.g., h1, h2, h3, h4)

CSS Syntax vs. HTML Syntax

- HTML's attributes uses "=" to separate the name and value.
- The name-value pairs in HTML are separated by spaces.

HTML Style:

```
<img src="logo.gif" alt="Logo" height="10" width="20">
```

CSS Style:

```
h1 {   text-align: center;   font-family: "Trebuchet MS", margin: 10px auto 10px auto;}
```

➢ Any other differences?

CSS comments

- Comments can be inserted inside the style sheet enclosed between /* and */
- the <!-- ... --> HTML comment style is NOT supported in CSS.
- Example:

```
/* global styles */
Body
{
background-color: orange; font-family:
Arial, Helvetica, sans-serif;
color: white;
}
```

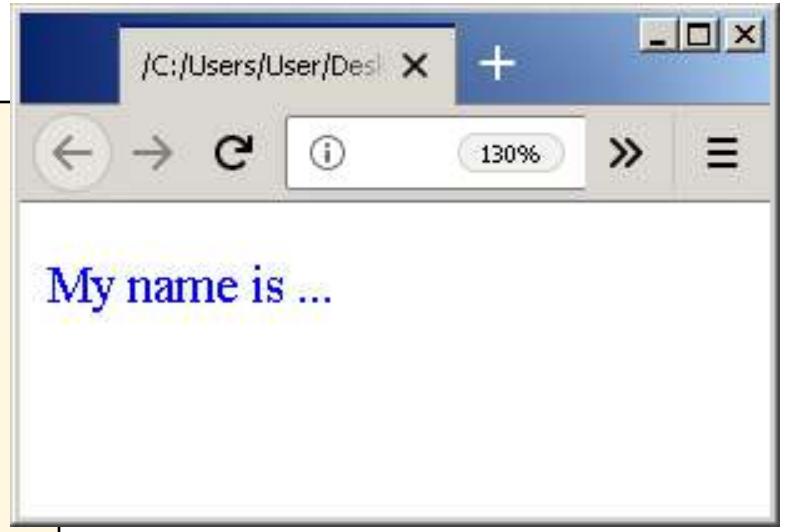
CSS Selector

- CSS selectors are used to "find" (or select) HTML elements/tags based on their element name, id, and class.

❑ The element/tag Selector

- The element selector selects elements based on the element name.

```
<head> <style>
        p  {color:blue;}
</style> </head>
<body>
<p>My name is …..</p>
</body>
```



CSS Selector *Cont'd*

❑ The id Selector

- The id selector uses the id attribute of an HTML element to select a specific element.

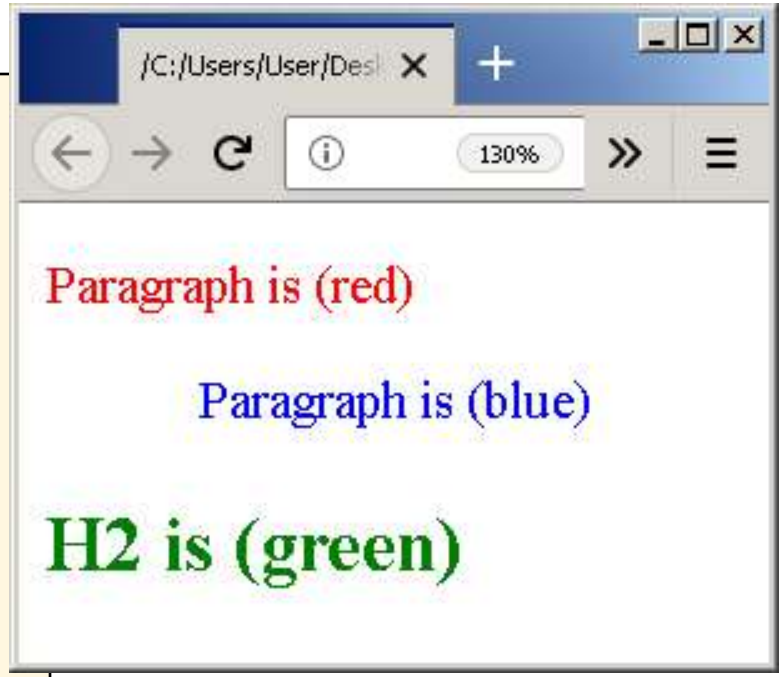
```
<head> <style>
p        { color:red; }
p#id1  { color:blue;  }
p#id2  { color:green; }
#Orng   { color:orange; }
</style> </head>
<body>
<p id="id1">Paragraph  is (blue)</p>
<p id="id2">Paragraph  is (green)</p>
<p id="Orng">Paragraph  is (orange)</p>
<p>Paragraph  is (red)</p>
</body>
```

CSS Selector *Cont'd*

❑ The class Selector

- The class selector selects elements with a specific class attribute. write a period (.) character, followed by the name of the class.

```
<head> <style>
p              { color:red;  }
p.blue { color:blue; text-align: center; }
.green { color:green; }
</style> </head>
<body>
<p>Paragraph  is (red)</p>
<p class="blue">Paragraph  is (blue)</p>
<h2 class="green">H2  is (green)</h2>
</body>
```



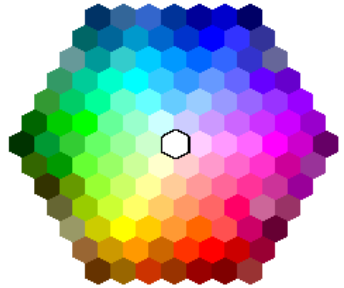
CSS Selector *Cont'd*

Universal Selector: *❑

- The universal selector * selects ALL the elements in the document.

```
<head> <style>
* { color:red;  }
</style> </head>

<body>

<p>This  is Paragraph  </p>
<h2> This is Heading 2  </h2>

</body>
```

Color Properties



- Color can be expressed as:

1- RGB in the form of rgb(r, g, b). The r, g, b can be expressed in a decimal value between 0 and 255; or in percentage between 0% and 100%.

Ex: color:rgb(20, 130, 58);

2- RGB hexadecimal triplets in the form of #rrggbb, where rr, gg, bb are values of red, green and blue. The values are between 00 and FF, in hexadecimal.

Ex: color:#12ABFF

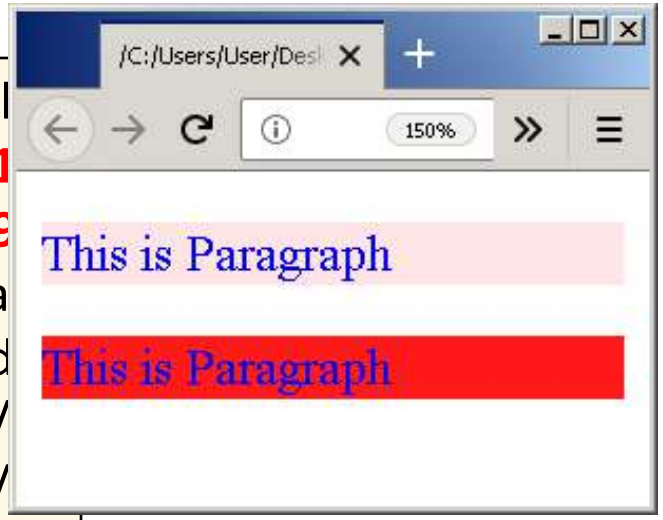
Color Properties in CSS *Cont'd*

3- RGBA in the form of rgba(r, g, b, a): RGB with an additional A (alpha channel).

- The A is used to control the transparency/opacity, with a=1.0 for opaque; and a=0.0 for totally transparent.

  Ex: background:rgba(**255**, **0**, **0**, **0.5**);

```
<head> <styl
p#id1 { color:blue; background:rgba(255, 0, 0, 0.1
p#id2 { color:blue; background:rgba(255, 0, 0, 0.9
</style> </hea
<bod
<p id="id1">This is Paragraph</
<p id="id2">This is Paragraph</
</body>
```



Color Properties in CSS *Cont'd*

4- Valid color name: There are 16 pre-defined English color names as follows.

Length Measurements

- Many CSS properties, such as width, height, margin, border, padding, font-size and line-height, require a length measurement.

- There are two types of length :

  1- The absolute: in (inch), cm (centimeter), mm (millimeter) and, pt (point ≈ 0.35mm).

  2- The relative: px (pixel), % (percent) and, em.

Note: em: the width of the letter 'm' of a referenced font, generally, the current font. For example, margin:2em means that the margins are twice the current (referenced) font-size.

Font Properties

The frequently-used font properties are:

- font-family: font-name|generic-family

  Ex: p {font-family: "Times New Roman", “Georgia”, “Serif”;}

- font-size: n|n%|xx-small|x-small|small|medium|large|x-large|xx-large|smaller|larger

  Ex: font-size:11px;

- font-weight: normal|bold|bolder|lighter|100|200| ...|800|900

  Where, 100 to 900, in multiple of 100. The value of 400 is the normal weight; while 700 is bold.

- font-variant: normal|small-caps

  The small-caps is smaller than the uppercase.

Font Properties *Cont'd*

- font: font-style font-variant font-weight font-size/line-height font-family

Example

```
p {
        font-size: 14px;
        font-weight: bold;
        line-height: 140%;
        font-family: Arial, sans-serif;
}
```

is the same as:

```
p { font: bold 14px/140% Arial, sans-serif; }
```

Text Properties

- text-align: left | right | center | justify
- line-height: normal | n | n% | factor
  - Set the height of the line. The factor gives the factor to be multiplied by the current font-size. E.g., factor of 1.5 means 1.5 times of the current font.
- text-decoration: none|underline|overline|line-through
- text-transform: none|uppercase|lowercase|capitalize
  - The capitalize transforms the first letter to uppercase.
- text-indent: n|n%

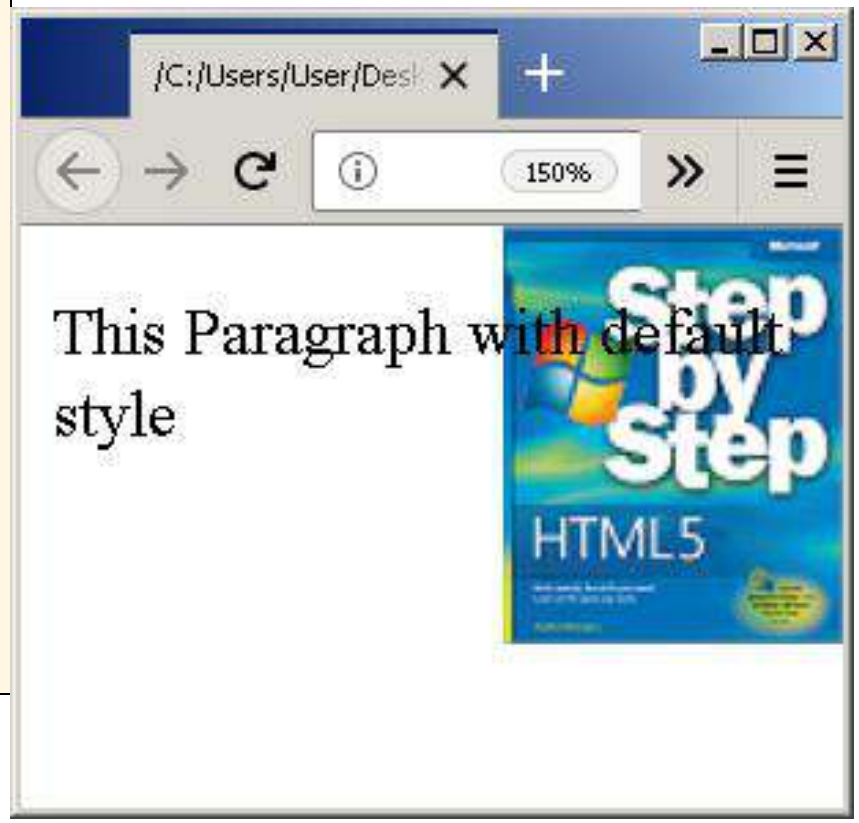
  Indent the first-line of the paragraph. To indent all the lines of a paragraph (i.e., the whole block), use padding or margin. n% : is n% of the width of the parent element
- letter-spacing: normal|n
- word-spacing: normal|n
- color: color

Background Properties

- background-color: #rrggbb|rgb(r,g,b)|rgba(r,g,b,a)|color-name|transparent
  - Set the background color of an element. The default is transparent.
- background-image: url(imageURL)|none
  - Use an image as the background.
- background-repeat: repeat|repeat-x|repeat-y|no-repeat
  - Define how the background image shall be repeated in x and y direction or both.
- background-attachment: scroll|fixed
  - Define whether background image shall scroll with the page or fixed.
- background-position: x y|x% y%|top left|top center|top right|center left|center center|center right|bottom left|bottom center|bottom right
  - Set the initial position of the background image. Note that there are two values, specifying the x and y position respectively.

Background Properties *example*

```
<head> <style>
body {
background-image: url("ebook1.png");
background-repeat: no-repeat;
background-position: right top;
}
</style> </head>

<body>

<p>This Paragraph with default style</p>

</body>
```



Border Style

The “border-style” property specifies what kind of border to display.

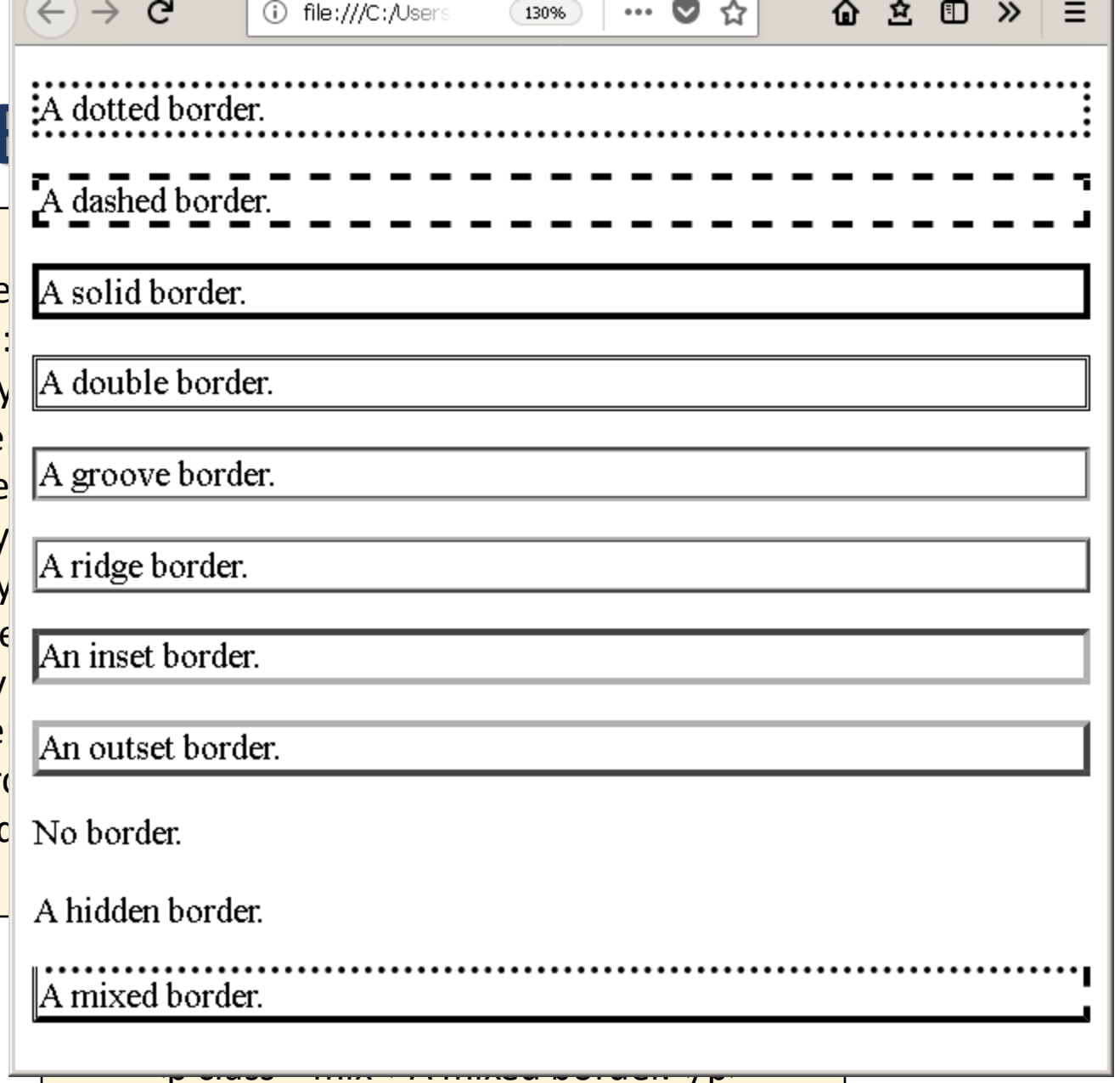
- dotted - Defines a dotted border
- dashed - Defines a dashed border
- solid - Defines a solid border
- double - Defines a double border
- groove - Defines a 3D grooved border. The effect depends on the border-color value
- ridge - Defines a 3D ridged border. The effect depends on the border-color value
- inset - Defines a 3D inset border. The effect depends on the border-color value
- outset - Defines a 3D outset border. The effect depends on the border-color value
- none - Defines no border
- hidden - Defines a hidden border

Border Style *example*

```
<style>
p.dotted {border-style: dotted;}
p.dashed {border-style: dashed;}
p.solid {border-style: solid;}
p.double {border-style: double;}
p.groove {border-style: groove;}
p.ridge {border-style
p.inset {border-style
p.outset {border-style:
p.none {border-style
p.hidden {border-style: h
p.mix {borde
dotted dashed solid d
```

```
<body>
<p class="dotted">A dotted border.</p>
<p class="dashed">A dashed border.</p>
<p class="solid">A solid border.</p>
<p class="double">A double border.</p>
<p class="groove">A groove border.</p>
<p class="ridge">A ridge border.</p>
<p class="inset">An inset border.</p>
<p class="outset">An outset border.</p>
<p class="none">No border.</p>
<p class="hidden">A hidden border.</p>
<p class="mix">A mixed border.</p>
</body>
```

```
p.dotted {border-style
p.dashed {border-style:
p.solid {border-sty
p.double {border-style
p.groove {border-style
p.ridge {border-sty
p.inset {border-sty
p.outset {border-style
p.none {border-sty
p.hidden {border-style
p.mix {bord
dotted dashed solid
```




```
</body>
```

List Properties

In HTML, there are two main types of lists:

- unordered lists (<ul>) - the list items are marked with bullets
- ordered lists (<ol>) - the list items are marked with numbers or letters

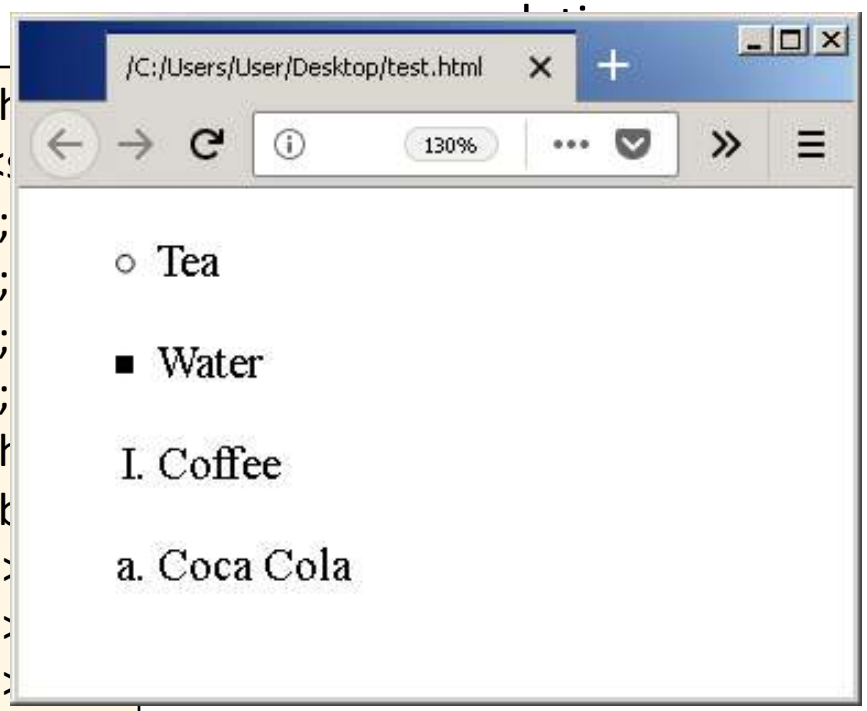
The CSS list properties allow you to:

- Set different list item markers for ordered lists
- Set different list item markers for unordered lists
- Set an image as the list item marker
- Add background colors to lists and list items

List Properties *Cont'd*

- Different List Item Markers
- **list-style-type**: none|disc|circle|square|
list-style-type: lower-alpha|upper-alpha|decimal|decimal-leading-zero|lower-roman|upper-roman|lower-greek|lower-latin|upper-

```
<h
<s
        ul.a {   list-style-type: circle;
      ul.b {   list-style-type : square;
   ol.c {   list-style-type : upper-roman;
    ol.d {   list-style-type : lower-alpha;
                                  </style></h
                                  <b
      <ul class="a"> <li>Tea</li> </ul>
</ul>      <ul class="b"> <li>Water</li>
    <ol class="c">  <li>Coffee</li> </ol>
 <ol class="d">  <li>Coca Cola</li></ol>
                                  </body>
```

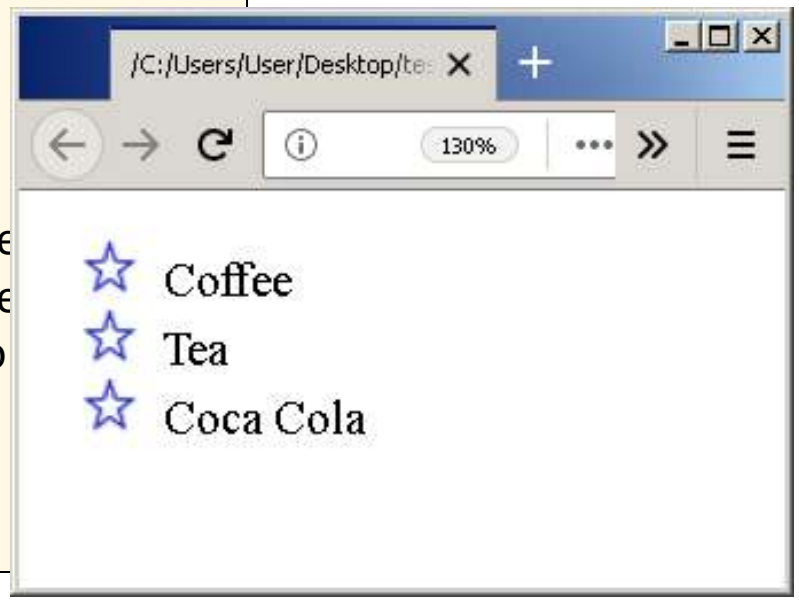


List Properties *Cont'd*

- An Image as The List Item Marker
  - **list-style-image**: none | *imageURL*)

star.png
image

```
<head>
<style>
ul {    list-style-image: url('star.png');  }
</style></head>

<li>Coffe
<li>Te
<li>Coca Co
```



Margin properties

- The CSS margin properties are used to generate space around elements.
- The margin properties set the size of the white space OUTSIDE the border.
- The margin are:
  - **margin-top**: auto|*n*|*n*%
  - **margin-right**: auto|*n*|*n*%
  - **margin-bottom**:auto|*n*|*n*%
  - **margin-left**: auto|*n*|*n*%

The values are:

auto - the browser calculates the margin

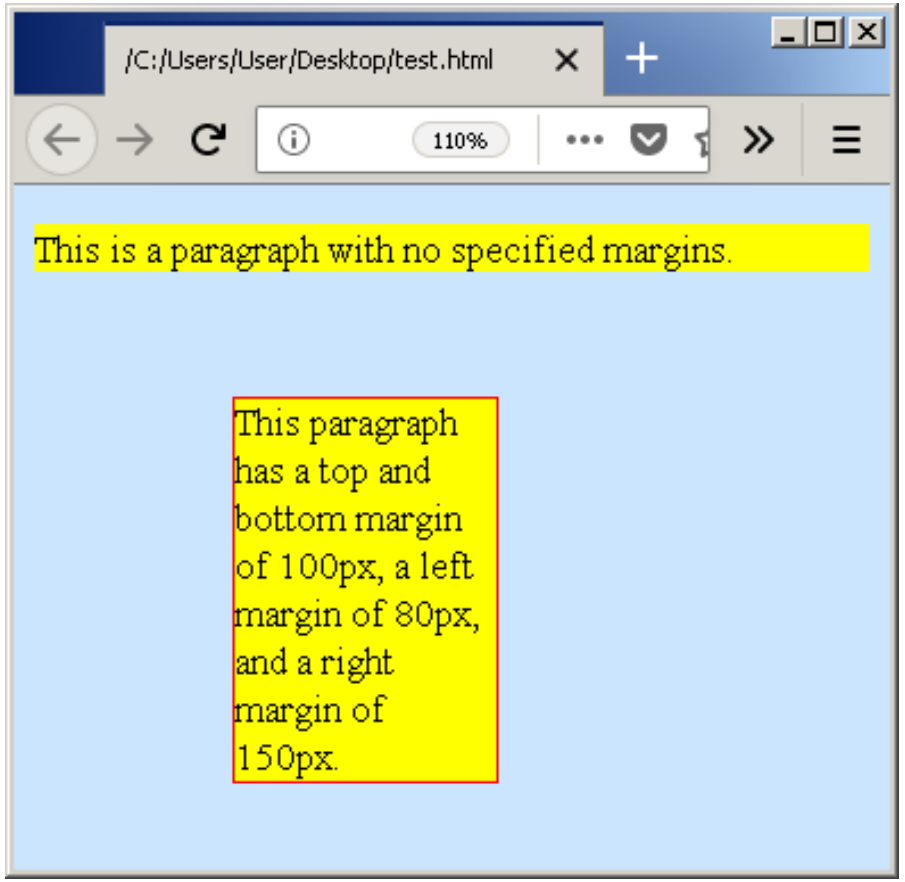
n - specifies a margin in px, pt, cm, etc.

n% - specifies a margin in % of the width of the containing element

- it is possible to specify all the margin properties in one property
  - **margin**: margin-top margin-right margin-bottom margin-left

Margin properties *Example*

```
<style>
body { background-color:#cce5ff; }
p { background-color: yellow;}
p.ex {border:1px solid red;
margin-top: 50px;
margin-bottom: 10px;
margin-right: 150px;
margin-left: 80px;}
</style></head>

<body>

<p>This is a paragraph with no specified margins.</p>
<p class="ex">This paragraph has a top and bottom margin of 100px, a left margin of 80px, and a right margin of 150px.</p>

</body>
```



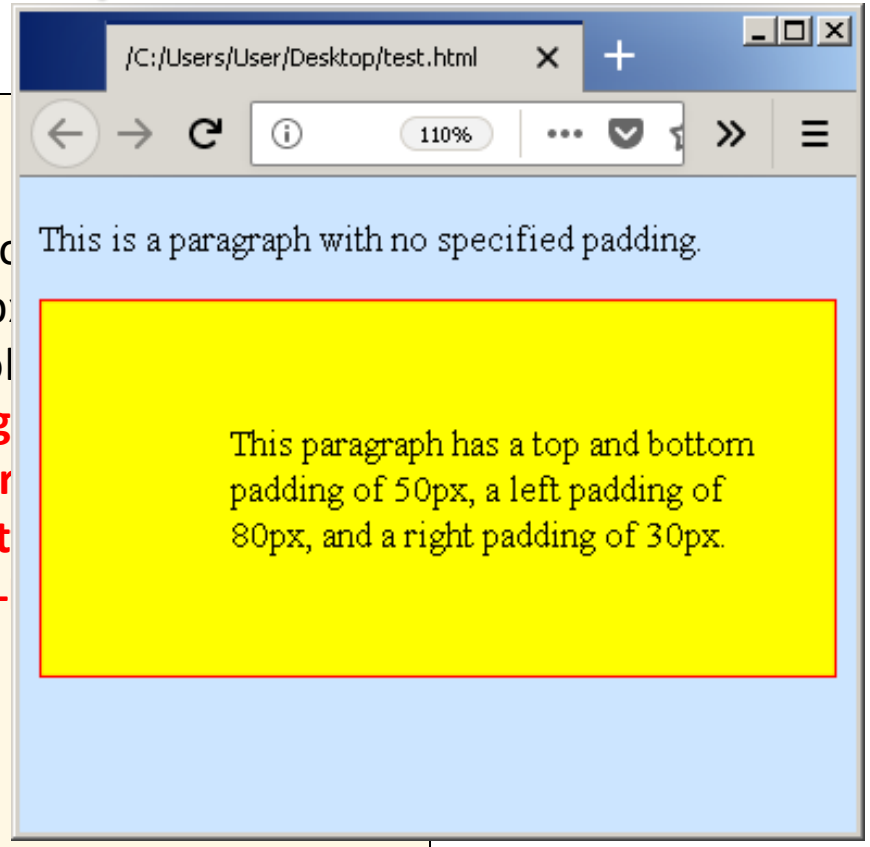
Padding Properties *Example*

- The CSS padding properties define the white space between the element content and the element border.
- The properties are :
  - **padding-top**: *n*|*n*%
  - **padding-right**: *n*|*n*%
  - **padding-bottom**: *n*|*n*%
  - **padding-left**: *n*|*n*%
- The values are:
  - *n* - specifies a padding in px, pt, cm, etc.
  - *n*% - specifies a padding in % of the width of the containing element
- it is possible to specify all the margin properties in one property
  - **padding**: padding-top padding-right padding-bottom padding-left

Padding Properties *Example*

```
body {background-colo
p.one {border: 1p
background-col
padding
padding-r
padding-bot
padding-
```

```
<p>This is a paragraph with no specified padding.</p>
<p class="one">This paragraph has a top and bottom padding of
50px, a left padding of 80px, and a right padding of 30px.</p>

</body>
```



Styling Links

- Links can be styled with any CSS property (e.g. color, font-family, background, etc.)
- Ex:

```
<head>
<style>
a {
color: hotpink;
}
</style>
</head>
<body>

<p><a href="default.html"
target="_blank">This is a link</a></p>

</body>
```

Styling Links *cont'd*

links can be styled differently depending on what **state** they are in as the following:

a:link - a normal, unvisited link

a:visited - a link the user has visited

a:hover - a link when the user mouses over it

a:active - a link the moment it is clicked

Styling Links *example*

```
<head>
<style>
a:link {color: red;}
a:visited {color: green;}
a:hover {color: yellow;}
a:active {color: gray;}
</style>
</hea
<bod

<p><a href="default.htm
target="_blank">This is a link</a></

</bod
```