



HTML 4.01

HTML Documents

- Web pages are created using Hypertext Markup Language (**HTML**)
- Web pages are commonly referred to as **HTML pages** or **documents**
- A **markup language** is a set of characters or symbols that define a document's logical structure
- HTML is based on an older language called Standard Generalized Markup Language (**SGML**)

HTML Documents (continued)

- Like **SGML**, **HTML** was originally designed as a way of defining the elements in a document independent of how they would appear
- **HTML** has evolved into a language that defines how elements should appear in a Web browser

Basic HTML Syntax

- **HTML documents** are text documents that contain formatting instructions called **tags**
- HTML tags include:
 - Formatting commands (**boldface** or *italic*)
 - Controls that allow user input (option buttons or check boxes)
- Tags are enclosed in brackets (**< >**) and consist of an opening tag and a closing tag

Tag syntax (1)

tag ::= < element-name attribute-pair* >

attribute-pair ::= attribute-name = string

element-name ::= sequence of letters and digits, starting with letter

attribute name ::= same as element-name

string ::= sequence of characters, often quoted

<body bgcolor=white>

<body bgcolor="white">

Tag and Element syntax (2)

- Tags are mostly case-insensitive
 - exception: some attribute values
- Boolean attributes don't need values
- End tags
 - element-name is preceded by a slash (/)
 - never have attributes

```
<body bgcolor = " white">
```

```
....
```

```
</body>
```

Basic HTML Syntax (continued)

- A tag pair and the data it contains are referred to as an **element**
- The information contained within an element's opening and closing tags is referred to as its **content**
- Elements that do not require a closing tag are called **empty elements**

```
<p>data</p>
```

Basic HTML Syntax (continued)

Table 1-1 Common HTML elements

HTML Element	Description
<code></code>	Formats enclosed text in a bold typeface
<code><body></body></code>	Encloses the body of the HTML document
<code>
</code>	Inserts a line break
<code><center></code>	Centers a paragraph in the middle of a Web page
<code><head></head></code>	Encloses the page header and contains information about the entire page
<code><hn></hn></code>	Indicates heading level elements, where n represents a number from 1 to 6
<code><hr></code>	Inserts a horizontal rule
<code><html></html></code>	Begins and ends an HTML document; these are required elements
<code><i></i></code>	Formats enclosed text in an italic typeface
<code></code>	Inserts an image file
<code><p></p></code>	Identifies enclosed text as a paragraph
<code><u></u></code>	Formats enclosed text as underlined

Basic HTML Syntax (continued)

- HTML documents must have a file extension of .html or .htm
- All **HTML** documents must use the `<html>` element as the root element
- A **root element** contains all the other elements in a document

Basic HTML Syntax (continued)

- The **<head>** element contains information that is used by the Web browser
- A `<head>` element must contain a **<title> element**
- The `<head>` element and the elements it contains are referred to as the **document head**
- The `<body>` element and the text and elements it contains are referred to as the **document body**



Basic HTML Syntax (continued)

- The process by which a Web browser assembles or formats an HTML document is called **parsing** or **rendering**
- Example:
`<p>This paragraph will appear in boldface in a Web browser</p>`
- Parameters used to configure many HTML elements are called **attributes**
- Insert line breaks using the paragraph `<p>` and line break `
` elements

Basic HTML Syntax (continued)

Sample HTML Code

```
<html>
<head>
<title>Toner Cartridge Sales</title>
</head>
<body>
<h1>Toner Cartridge Sales</h1>
<hr>
<h2>Lexmark Toner Cartridges</h2>

<p><b>Model #</b>:LEX 1382100<br>
<b>Compatibility</b>: Optra 4049/3112/3116<br>
<b>Price</b>: $189.99</p>
<p><b>Model #</b>:LEX 1380520<br>
<b>Compatibility</b>:Lexmark 4019/4028/4029<br>
<b>Price</b>:$209.00</p>
</body>
</html>
```

HTML Structure

- Minimal legal HTML document:

```
<html>  
<head><title>Test Document</title>  
</head>  
<body>  
<!-- Content goes here -->  
</body>  
</html>
```

- Corresponds to a simple tree

HTML has two parts

- HEAD: Information about the document
 - TITLE (mandatory): appears in title bar
 - META: various other info about document
- BODY: visible content of document
 - For now: we ignore scripts and frames
 - Two kinds:
 - block structure
 - phrase structure

```
<body bgcolor="#FFFFFF">
```

<head> standard format

```
<!-- The site is designed by your name 09/2006 --->
<meta name="author" content="master@xxx.com" />
<meta name="description " content="xxxxxxxxxxxxxx" />
<meta name="keywords" content="xxx,xxxx,xxx " />
<link href="style/style.css" rel="stylesheet"
  type="text/css" />
<title>xxxxxxxxxxxxxxxxxxxx</title>
```

HTML Encoding (Character Sets)

- To display an HTML page correctly, a web browser must know which character set to use.
- The default character set for HTML4.01 is ISO-8859-1, which covers almost all of the characters and symbols in the world.

```
<meta http-equiv = "Content-Type" content="text/html; charset=gb2312" />
```

The UTF-8 Character Set

UTF-8 is identical to ASCII for the values from 0 to 127.

UTF-8 does not use the values from 128 to 159.

UTF-8 is identical to both ANSI and 8859-1 for the values from 160 to 255.

UTF-8 continues from the value 256 with more than 10 000 different characters.

<head> extensible format

```
<meta http-equiv = "expires" content = "Wed, 26 Feb  
2006 08:21:57 GMT" />
```

```
<meta http-equiv = "Pragma" content = "no-cache" />
```

```
<meta http-equiv = "Window-target" content = "_top" />
```

```
<meta http-equiv = "Refresh" content = "5;  
url=http://www.yahoo.com" />
```

```
<meta name = "robots" content = "none" />
```

```
<link rel = "Shortcut Icon" href = "favicon.ico" />
```

```
<script language = "javascript" src = "script/xxxxx.js" ></  
script >
```

Block structure (1)

- Headings
 - Six levels (H1... H6)
 - Intended as levels of section headings
- Lists
 - Simple lists all use “list items” (LI, end optional)
 - Unordered lists (UL)
 - Ordered lists (OL)
 - Description lists
 - Description text (DT)
 - Description data (DD)

```
<h1>This is a heading</h1>  
<h2>This is a heading</h2>  
<h3>This is a heading</h3>  
<h4>This is a heading</h4>  
<h5>This is a heading</h5>  
<h6>This is a heading</h6>
```

Block structure (2)

- Paragraph: P (end optional)
 - for normal text
- Other blocks
 - BLOCKQUOTE for quotations
 - CITE attribute: URI of source document
 - ADDRESS: for e-mail addresses

Here comes a long quotation:

```
<blockquote>
```

This is a long quotation. This is a long quotation. This is a long quotation.

This is a long quotation. This is a long quotation.

```
</blockquote>
```

Block Structure (3)

- Some block structure is for presentation
- BR: forced line break
- HR: horizontal rule

```
<p>
```

```
To break<br>lines<br>in  
a<br>paragraph,<br>use the br tag.
```

```
</p>
```

Block Structure (4)

- PRE: preformatted text
 - whitespace is not collapsed, typewriter font

```
<p>  
This paragraph  
contains      a lot of spaces  
in the source  code,  
but the browser  
ignores it.  
</p>
```

```
<pre>  
for i = 1 to 10  
    print i  
next i  
</pre>
```

Block Structure (5)

- **Comments in HTML**
- The comment tag is used to insert a comment in the HTML source code. A comment will be ignored by the browser. You can use comments to explain your code, which can help you when you edit the source code at a later date.
- Comments are also great for debugging HTML, because you can comment out HTML lines of code, one at a time, to search for errors

```
<!-- This is a comment -->
```

Presentation Phrase Structure

- Doesn't create new paragraphs
- Whitespace is collapsed
- Font changes
 - Bold (B), italic (I), teletype (TT)
 - discouraged, use style sheets now
 - FONT: specifies font to use, **deprecated**
 - Emphasis (EM), STRONG, CODE, KBD, etc.
 - Subscripts (SUB) and superscripts (SUP)

Example:

```
<b>This text is bold</b>
```

```
<br>
```

```
<strong>This text is strong</strong>
```

```
<br>
```

```
<em>This text is emphasized</em>
```

```
<br>
```

```
<i>This text is italic</i>
```

```
<br>
```

```
<big>This text is big</big>
```

```
<br>
```

```
<small>This text is small</small>
```

```
<br>
```

```
This text contains<sub>subscript</sub>
```

```
<br>
```

```
This text contains<sup>superscript</sup>
```


HTML Entities

- **Character Entities**
- Some characters like the < character, have a special meaning in HTML, and therefore cannot be used in the text.
- To display a less than sign (<) in HTML, we have to use a character entity.
- A character entity has **three parts**: an ampersand (&), an entity name or a # and an entity number, and finally a semicolon (;).
- To display a less than sign in an HTML document we must write: **<** or **<**;

< or **<**;

HTML Entities(2)

Result	Description	Entity Name	Entity Number
	non-breaking space	 	
<	less than	<	<
>	greater than	>	>
&	ampersand	&	&
"	quotation mark	"	"
©	copyright	©	©
®	registered trademark	®	®

HTML Entities(3)

- **Non-breaking Space**

The most common character entity in HTML is the non-breaking space.

Normally HTML will truncate spaces in your text. If you write 10 spaces in your text HTML will remove 9 of them. To add spaces to your text, use the ` ` character entity.

- The advantage of using a name instead of a number is that a name is easier to remember. The disadvantage is that not all browsers support the newest entity names, while the support for entity numbers is very good in almost all browsers.
- **Note** that the entities are case sensitive.

HTML Links

- HTML uses a hyperlink to link to another document on the Web.
- **The Anchor Tag**
- HTML uses the `<a>` (anchor) tag to create a link to another document.

An anchor can point to any resource on the Web: an HTML page, an image, a sound file, a movie, etc.

The syntax of creating an anchor:

```
<a href="url">Text to be displayed</a>
```

HTML Links(2)

- The `<a>` tag is used to create an anchor to link from, the `href` attribute is used to address the document to link to, and the words between the open and close of the anchor tag will be displayed as a hyperlink.
- **the Href and Target Attribute:**

```
<a href=" http://www.hunau.net/" target="_blank">Hunau</a>
```

<code>_self</code>	Default. Opens the document in the same window/tab as it was clicked
<code>_blank</code>	Opens the document in a new window or tab
<code>_parent</code>	Opens the document in the parent frame
<code>_top</code>	Opens the document in the full body of the window

HTML Links(3)

- **the Name Attribute**
- The name attribute is used to create a named anchor. When using named anchors we can create links that can jump directly into a specific section on a page, instead of letting the user scroll around to find what he/she is looking for.
- Below is the syntax of a named anchor:

```
<a name="label">Text to be displayed</a>
```

Note: To link directly to the “label” section, add a href attribute and a # sign and the name of the anchor .

This example demonstrates how to use a link to jump to another part of a document.

```
<a name="top"></a>
<h2>table of contents</h2>
<a href="#c01">chapter 01</a><br />
<a href="#c02">chapter 02</a><br />
<a href="#c03">chapter 03</a><br />
<a href="#c04">chapter 04</a><br /><br />
<a name="c01">chapter 01  xxx</a>
<p>.....</p><br />
<a name="c02">chapter 02  xxy</a>
<p>.....</p><br />
<a name="c03">chapter 03  xxz</a>
<p>.....</p><br />
<a href="#top">Back to TOP</a><br />
<h3>Other article:</h3>
<a href="http://www.hunau.net/html_links.asp#tips1">Jump to the Useful
Tips Section</a>
```

This example demonstrates a more complicated mailto link.

```
<p>
```

This is another mailto link:

```
<a
```

```
href="mailto:someone@microsoft.com?cc=someoneelse@
microsoft.com&bcc=andsomeoneelse2@microsoft.com&s
ubject=Summer%20Party&body=You%20are%20invited%
20to%20a%20big%20summer%20party!">Send mail!</a>
```

```
</p>
```

```
<p>
```

Note: Spaces between words should be replaced by %20 to **ensure** that the browser will display your text properly.

```
</p>
```


HTML Frames

- With frames, you can display more than one Web page in the same browser window. Each HTML document is called a frame, and each frame is independent of the others.
- The disadvantages of using frames are:
The web developer must keep track of more HTML documents .
It is difficult to print the entire page.

HTML Frames(2)

- **The Frameset Tag**
- The `<frameset>` tag defines how to divide the window into frames
- Each frameset defines a set of rows **or** columns
- The values of the rows/columns indicate the amount of screen area each row/column will occupy
- **The Frame Tag**
- The `<frame>` tag defines what HTML document to put into each frame

HTML Frames(3)

In the example below we have a frameset with two columns. The first column is set to 25% of the width of the browser window. The second column is set to 75% of the width of the browser window. The HTML document "frame_a.htm" is put into the first column, and the HTML document "frame_b.htm" is put into the second column:

```
<frameset cols="25%,75%">  
  <frame src="frame_a.htm">  
  <frame src="frame_b.htm">  
</frameset>
```

This example demonstrates how to use the `<noframes>` tag.

```
<html>
<frameset cols="25%,50%,25%">
  <frame src="frame_a.htm">
  <frame src="frame_b.htm">
  <frame src="frame_c.htm">
<noframes>
<body>Your browser does not handle frames!</body>
</noframes>
</frameset>
</html>
```

Important: You cannot use the `<body></body>` tags together with the `<frameset></frameset>` tags! However, if you add a `<noframes>` tag containing some text for browsers that do not support frames, you will have to enclose the text in `<body></body>` tags! See how it is done in the first example below.

This example demonstrates how to make a navigation frame. The navigation frame contains a list of links with the second frame as the target .

```
<html>
<frameset cols="120,*">
<frame src="tryhtml_contents.htm">
<frame src="frame_a.htm"
name="showframe">
</frameset>
</html>
```

tryhtml_contents.htm

```
<a href="frame_a.htm" target="showframe"> Frame a </a><br>
<a href="frame_b.htm" target="showframe"> Frame b </a><br>
<a href="frame_c.htm" target="showframe"> Frame c </a>
```

HTML Tables

- Tables are defined with the `<table>` tag. A table is divided into rows (with the `<tr>` tag), and each row is divided into data cells (with the `<td>` tag). The letters td stands for "table data," which is the content of a data cell. A data cell can contain text, images, lists, paragraphs, forms, horizontal rules, tables, etc.

How it looks in a browser?

```
<table border="1">  
<tr>  
<td>row 1, cell 1</td>  
<td>row 1, cell 2</td>  
</tr>  
<tr>  
<td>row 2, cell 1</td>  
<td>row 2, cell 2</td>  
</tr>  
</table>
```

Tables and the Border Attribute

- If you do not specify a border attribute the table will be displayed without any borders. Sometimes this can be useful, but most of the time, you want the borders to show.
- To display a table with borders, you will have to use the border attribute:

```
<table border="1">  
<tr>  
<td>Row 1, cell 1</td>  
<td>Row 1, cell 2</td>  
</tr>  
</table>
```


Headings in a Table

- Headings in a table are defined with the `<th>` tag.

- **Empty Cells in a Table**

Table cells with no content are not displayed very well in most browsers.

To avoid this, add a non-breaking space (` `) to empty data cells, to make the borders visible:

```
<table border="1">
  <tr>
    <th>Heading</th>
    <th>Another Heading</th>
  </tr>
  <tr>
    <td>row 1, cell 1</td>
    <td>row 1, cell 2</td>
  </tr>
  <tr>
    <td>row 2, cell 1</td>
    <td>row 2, cell 2</td>
  </tr>
</table>
```

Table Tags

Tag	Description
<code><table></code>	Defines a table
<code><th></code>	Defines a table header
<code><tr></code>	Defines a table row
<code><td></code>	Defines a table cell
<code><caption></code>	Defines a table caption
<code><colgroup></code>	Defines groups of table columns
<code><col></code>	Defines the attribute values for one or more columns in a table
<code><thead></code>	Defines a table head
<code><tbody></code>	Defines a table body
<code><tfoot></code>	Defines a table footer

Table Examples Advanced

```
<h4>Cell that spans two columns:</h4>
<table border="1">
<tr>
  <th>Name</th>
  <th colspan="2">Telephone</th>
</tr>
<tr>
  <td>Bill Gates</td>
  <td>555 77 854</td>
  <td>555 77 855</td>
</tr>
</table>
```

Table Examples Advanced

```
<h4>Cell that spans two rows:</h4>
<table border="1">
<tr>
  <th>First Name:</th>
  <td>Bill Gates</td>
</tr>
<tr>
  <th rowspan="2">Telephone:</th>
  <td>555 77 854</td>
</tr>
<tr>
  <td>555 77 855</td>
</tr>
</table>
```

HTML Lists

- HTML supports ordered, unordered and definition lists.
- **Unordered Lists**
- An unordered list is a list of items. The list items are marked with bullets (typically small black circles).
- An unordered list starts with the `` tag. Each list item starts with the `` tag.

```
<ul>  
  <li>Coffee</li>  
  <li>Milk</li>  
</ul>
```

Ordered Lists

- An ordered list is also a list of items. The list items are marked with numbers.
- An ordered list starts with the `` tag. Each list item starts with the `` tag.

```
<ol>  
  <li>Coffee</li>  
  <li>Milk</li>  
</ol>
```

Definition Lists

- A definition list is **not** a list of items. This is a list of terms and explanation of the terms.
- A definition list starts with the `<dl>` tag. Each definition-list term starts with the `<dt>` tag. Each definition-list definition starts with the `<dd>` tag.

```
<dl>  
<dt>Coffee</dt>  
<dd>Black hot drink</dd>  
<dt>Milk</dt>  
<dd>White cold drink</dd>  
</dl>
```

Lists Examples Advanced

```
<h4>A nested List:</h4>
<ul>
  <li>Coffee</li>
  <li>Tea
    <ul>
      <li>Black tea</li>
      <li>Green tea
        <ul>
          <li>China</li>
          <li>Africa</li>
        </ul>
      </li>
    </ul>
  </li>
  <li>Milk</li>
</ul>
```


HTML Forms and Input

- HTML Forms are used to select different kinds of user input.
- A form is an area that can contain form elements.
- Form elements are elements that allow the user to enter information (like text fields, textarea fields, drop-down menus, radio buttons, checkboxes, etc.) in a form.

```
<form>  
  <input>  
  <input>  
</form>
```

- **Input**
- The most used form tag is the `<input>` tag. The type of input is specified with the `type` attribute. The most commonly used input types are explained below.
- **Text Fields**
- Text fields are used when you want the user to type letters, numbers, etc. in a form.

```
<form> First name:  
<input type="text" name="firstname"> <br>  
Last name:  
<input type="text" name="lastname">  
</form>
```

- **Radio Buttons**
- Radio Buttons are used when you want the user to select one of a limited number of choices.
- **Checkboxes**
- Checkboxes are used when you want the user to select one or more options of a limited number of choices.

```
<form>  
<input type="radio" name="sex" value="male"> Male <br>  
<input type="radio" name="sex" value="female"> Female  
</form>
```

The Form's Action Attribute and the Submit Button

- When the user clicks on the "Submit" button, the content of the form is sent to another file. The form's action attribute defines the name of the file to send the content to. The file defined in the action attribute usually does something with the received input.

```
<form name="input" action="search.asp" method="get">  
  Username: <input type="text" name="user">  
<input type="submit" value="Submit">  
</form>
```

HTML Images

- **The Image Tag and the Src Attribute**
- In HTML, images are defined with the `` tag.
- The `` tag is empty, which means that it contains attributes only and it has no closing tag.
- To display an image on a page, you need to use the `src` attribute. `Src` stands for "source". The value of the `src` attribute is the URL of the image you want to display on your page.

```

```

- **The Alt Attribute**
- The alt attribute is used to define an "alternate text" for an image. The value of the alt attribute is an author-defined text:

```

```

- The "alt" attribute tells the reader what he or she is missing on a page if the browser can't load images. The browser will then display the alternate text instead of the image. It is a good practice to include the "alt" attribute for each image on a page, to improve the display and usefulness of your document for people who have text-only browsers.

HTML Scripts

- Add scripts to HTML pages to make them more dynamic and interactive.
- **Insert a Script into HTML Page**
- A script in HTML is defined with the `<script>` tag. Note that you will have to use the `type` attribute to specify the scripting language.

```
<script type="text/javascript">  
    document.write("Hello World!")  
</script>
```

How to Handle Older Browsers

- A browser that does not recognize the `<script>` tag at all, will display the `<script>` tag's content as text on the page. To prevent the browser from doing this, you should hide the script in comment tags. An old browser (that does not recognize the `<script>` tag) will ignore the comment and it will not write the tag's content on the page, while a new browser will understand that the script should be executed, even if it is surrounded by comment tags.

JavaScript:

```
<script type="text/javascript">
```

```
<!--
```

```
document.write("Hello World!")
```

```
//-->
```

```
</script>
```

VBScript:

```
<script type="text/vbscript">
```

```
<!--
```

```
document.write("Hello World!")
```

```
'-->
```

```
</script>
```

The <noscript> Tag

- In addition to hiding the script inside a comment, you can also add a <noscript> tag.
- The <noscript> tag is used to define an alternate text if a script is NOT executed. This tag is used for browsers that recognize the <script> tag, but do not support the script inside, so these browsers will display the text inside the <noscript> tag instead. However, if a browser supports the script inside the <script> tag it will ignore the <noscript> tag.

```
<noscript>Your browser does not support JavaScript!</noscript>
```

HTML 4.0 Standard Attributes



- HTML tags can have attributes. The special attributes for each tag are listed under each tag description. The attributes listed here are the core and language attributes that are standard for all tags (with a few exceptions):

HTML 4.0 Standard Attributes

Core Attributes

Attribute	Value	Description
class	<i>class_rule</i> or <i>style_rule</i>	The class of the element
id	<i>id_name</i>	A unique id for the element
style	<i>style_definition</i>	An inline style definition
title	<i>tooltip_text</i>	A text to display in a tool tip

Not valid in base, head, html, meta, param, script, style, and title elements.

HTML 4.0 Standard Attributes

Language Attributes

Attribute	Value	Description
dir	ltr rtl	Sets the text direction
lang	<i>language_code</i>	Sets the language code

Not valid in base, br, frame, frameset, hr, iframe, param, and script elements.

HTML 4.0 Standard Attributes

Keyboard Attributes

Attribute	Value	Description
accesskey	<i>character</i>	Sets a keyboard shortcut to access an element
tabindex	<i>number</i>	Sets the tab order of an element

HTML 4.0 Standard Attributes

Keyboard Attributes

Attribute	Value	Description
<code>accesskey</code>	<i>character</i>	Sets a keyboard shortcut to access an element
<code>tabindex</code>	<i>number</i>	Sets the tab order of an element

HTML 4.0 Event Attributes

- New to HTML 4.0 is the ability to let HTML events trigger actions in the browser, like starting a JavaScript when a user clicks on an HTML element. Below is a list of attributes that can be inserted into HTML tags to define event actions.

HTML 4.0 Event Attributes

Window Events

Attribute	Value	Description
onload	<i>script</i>	Script to be run when a document loads
onunload	<i>script</i>	Script to be run when a document unloads

Only valid in body and frameset elements.

HTML 4.0 Event Attributes

Form Element Events

Attribute	Value	Description
onchange	<i>script</i>	Script to be run when the element changes
onsubmit	<i>script</i>	Script to be run when the form is submitted
onreset	<i>script</i>	Script to be run when the form is reset
onselect	<i>script</i>	Script to be run when the element is selected
onblur	<i>script</i>	Script to be run when the element loses focus
onfocus	<i>script</i>	Script to be run when the element gets focus

Only valid in form elements.

HTML 4.0 Event Attributes

Form Element Events

Attribute	Value	Description
onchange	<i>script</i>	Script to be run when the element changes
onsubmit	<i>script</i>	Script to be run when the form is submitted
onreset	<i>script</i>	Script to be run when the form is reset
onselect	<i>script</i>	Script to be run when the element is selected
onblur	<i>script</i>	Script to be run when the element loses focus
onfocus	<i>script</i>	Script to be run when the element gets focus

Only valid in form elements.

HTML 4.0 Event Attributes

Keyboard Events

Attribute	Value	Description
onkeydown	<i>script</i>	What to do when key is pressed
onkeypress	<i>script</i>	What to do when key is pressed and released
onkeyup	<i>script</i>	What to do when key is released

Not valid in base, bdo, br, frame, frameset, head, html, iframe, meta, param, script, style, and title elements.

HTML 4.0 Event Attributes

Mouse Events

Attribute	Value	Description
onclick	<i>script</i>	What to do on a mouse click
ondblclick	<i>script</i>	What to do on a mouse double-click
onmousedown	<i>script</i>	What to do when mouse button is pressed
onmousemove	<i>script</i>	What to do when mouse pointer moves
onmouseout	<i>Script</i>	What to do when mouse pointer moves out of an element
onmouseover	<i>Script</i>	What to do when mouse pointer moves over an element
onmouseup	<i>script</i>	What to do when mouse button is released

Ready to Publish Your Work?

- **Your First Step: A Personal Web Server**
- If you want other people to view your pages, you must publish them.
- To publish your work, you have to copy your files to a web server.
- Your own PC can act as a web server if it is connected to a network.
- If you are running Windows 98, you can use the PWS (Personal Web Server).
- If you are running Windows 2000 or xp, you can use the Internet Information Server (IIS).

Homework 1

- 1.what is HTML?
 - 2.what is HTTP?
 - 3.What is the difference between Internet, WWW and Web?
-
- Translate
 - Web pages are usually written in HTML, which can markup content and supply hyperlinks using partial and full URLs. Web pages are placed under Web servers and become available on the web. Commercial hosting companies provide hosts and servers to serve Web pages online.

Homework 1

- Translate
- The strength of the web lies in its openness, speed, and low cost of entry. Enabling technologies for the web include networking protocols, data encoding formats, clients (browsers), servers, web page markup and styling languages, and client-side and server-side programming. The web can deliver text, images, animation, audio, video, and other multimedia content. Standard and proprietary media formats, tools, and players are also part of the web. These technologies are still developing and improving. The World Wide Web Consortium(W3C) is a nonprofit organization leading the way in developing open Web standards.



The End