

TAGS

[a](#)
[abbr](#)
[acronym](#)
[address](#)
[applet](#)
[area](#)
[b](#)
[base](#)
[basefont](#)
[bdo](#)
[big](#)
[blockquote](#)
[body](#)
[br](#)
[button](#)
[caption](#)
[center](#)
[cite](#)
[code](#)
[col](#)
[colgroup](#)
[dd](#)
[del](#)
[dfn](#)
[dir](#)
[div](#)
[dl](#)
[DOCTYPE](#)
[dt](#)

[em](#)
[fieldset](#)
[font](#)
[form](#)
[frame](#)
[frameset](#)
[h1,h2,h3,h4,h5,h6](#)
[head](#)
[hr](#)
[html](#)
[i](#)
[iframe](#)
[img](#)
[input](#)
[ins](#)
[isindex](#)
[kbd](#)
[label](#)
[legend](#)
[li](#)
[link](#)
[map](#)
[menu](#)
[meta](#)
[noframes](#)
[noscript](#)
[object](#)
[ol](#)
[optgroup](#)

[option](#)
[p](#)
[param](#)
[pre](#)
[q](#)
[s](#)
[samp](#)
[script](#)
[select](#)
[small](#)
[span](#)
[strike](#)
[strong](#)
[style](#)
[sub](#)
[sup](#)
[table](#)
[tbody](#)
[td](#)
[textarea](#)
[tfoot](#)
[th](#)
[thead](#)
[title](#)
[tr](#)
[tt](#)
[u](#)
[ul](#)
[var](#)

CORE ATTRIBUTES

[class](#)
[dir](#)
[id](#)
[lang](#)
[onclick](#)
[ondblclick](#)
[onkeydown](#)
[onkeypress](#)
[onkeyup](#)
[onmousedown](#)
[onmousemove](#)
[onmouseout](#)
[onmouseover](#)
[onmouseup](#)
[style](#)
[title](#)

COLOR CHART

[HTML Colors](#)

Other HTML Tags

These tags are not recognized by the W3C HTML version 4.01 standard. However, it is quite possible that you may find some of these tags to be of considerable value.

<audioscope>

The **audioscope** tag is used with WebTV to display the audioscope, which is a graphical display of sound waves.

<bgsound>

The **bgsound** tag is placed in the **head** element and is used to add sound to a web site.

<blackface> ... </blackface>

The **blackface** tag is used with WebTV to display text in a double weight bold font face.

<blink> ... </blink>

The **blink** tag causes the selected text to blink.

<bq>

The **bq** tag is used with WebTV to display a block quote.

<comment> ... </comment>

The **comment** tag allows the addition of comments to HTML documents created for display on Internet Explorer and WebTV.

<embed>

The **embed** tag is used with Netscape and allows the automatic display of a plug-in application.

<fn> ... </fn>

The **fn** tag is used with WebTV to display footnotes.

<ilayer> ... </ilayer>

The **ilayer** tag creates an inline layer.

<keygen>

The **keygen** tag is used with Netscape for encryption.

<layer> ... </layer>

The **layer** tag is used with Netscape to create multiple layers of text and images which lay in order, one on top of the other. Usually the lower layers are partially visible through the upper layers.

<limittext>

The **limittext** tag is deprecated by WebTV. It is used to set the width of the text display.

<listing> ... </listing>

The **listing** tag is deprecated by W3C HTML version 3.2. It displays text just like the **pre** tag.

<marquee> ... </marquee>

Code:

```
<marquee width="85%">
```

The **marquee** tag is used with Internet Explorer and WebTV to display a scrolling text.

```
</marquee>
```

Output:

The **marquee** tag is used with Internet Explorer and WebTV to display a scrolling text.

<multicol> ... </multicol>

The **multicol** tag is used with Netscape to format and display a multiple column flow of text and images.

<nobr> ... </nobr>

The **nobr** tag is widely implemented. It prevents any line breaks from occurring in a flow of text.

<noembed> ... </noembed>

The **noembed** tag is used with Netscape to display an alternative text for browsers that do not recognize the **embed** tag.

<nosmartquotes> ... </nosmartquotes>

The **nosmartquotes** tag is used with WebTV to prevent regular single '...' and double "..." quotes from automatically being replaced with smart single ‘...’ and smart double “...” quotes.

<plaintext>

The **plaintext** tag designates that all HTML tags are to be ignored and that the text is to be displayed as it is written.

<rt>

The **rt** tag is used to denote the annotation or pronunciation text used by the **ruby** tag.

<ruby> ... </ruby>

The **ruby** tag is used with Internet Explorer to create an annotation or pronunciation guide for a base text. The guide (called a ruby text) appears either inline or above the base text.

Code:

```
The <ruby>Sotano de las Golondrinas
<span style="color:red; font-size:24px; font-family:arial;">
<rt>Pit of the Swallows
</span>
</ruby> is in Mexico.
```

Output:

The Sotano de las Golondrinas **Pit of the Swallows** is in Mexico.

<server> ... </server>

The **server** tag is used with Netscape to insert server-side JavaScript into an HTML document.

<shadow>

The **shadow** tag is used with WebTV to add a shadow to text.

<sidebar>

The **sidebar** tag is used with WebTV to create a sidebar window that displays control elements.

<spacer>

The **spacer** tag is used with Netscape to add white space between words.

<wbr>

The **wbr** tag is used inside a **nobr** element to add a line break.

`<xml> ... </xml>`

The `xml` tag is used with Internet Explorer to delimit an XML data island inside an HTML page.

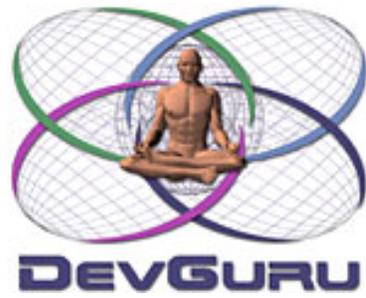
`<xmp> ... </xmp>`

The `xmp` tag is deprecated by W3C HTML version 3.2. It is equivalent to using:

`<pre width="80"> ... </pre>`

Here are a few helpful hints and guidelines to get you started:

- All HTML tags must be contained between an opening < angle bracket and a closing > angle bracket.
`<html>`
- Some tags have one or more associated attributes and, when they are included with the tag, they dictate additional information on how to implement the tag. The attributes of an element can be specified in any order, but only once each.
- Most attributes are composed of a name/value pair separated by an equal sign. Some attributes are composed only of a name and a Boolean value of true is automatically assigned.
``
- It is recommended that all tag and attribute names be in lower case. However, HTML is case insensitive.
- All values assign to an attribute should be enclosed between a pair of double quotes.
`<form action="NextPage.asp" method="post">`
- Many HTML tags require a mandatory closing tag, where the closing tag name is preceded by a forward slash. These tags usually require that some content be placed between the opening and closing tag.
`<title>Helpful Hints</title>`
- For some HTML tags, the closing tag is optional. However, it is recommended that you always include the closing tag when it is optional.
` The closing tags is optional for the li tag.`
- A few HTML tags do not have closing tags and therefore you should not include a closing tag.
`
`
- If there is no closing tag and if you wish to comply with the new XHTML version 1.0 standard, you can include a blank space followed by a forward slash after the tag name.
`
`
- You should nest HTML tags inside of each other in a logical manner. The first example is correctly coded. Note how the last tag, the **i**, is closed first and that the first tag, the **span** is closed last.
`<i>DevGuru</i>`
This is bad coding and may not display the desired results on some browsers.
`<i>DevGuru</i>`
- Some tags cannot be nested inside of themselves. For example, a **form** tag cannot appear inside a pair of opening and closing **form** tags. Another example is the **a** tag since hyperlinks cannot be inside of other hyperlinks.
- It is best for beginners to only change one piece of HTML code at a time and to view or test the results of the change, before proceeding to the next change.
- The appropriate use of the Active Server Pages (ASP) and Cascading Style Sheets (CSS) technologies can aid in creating very spectacular web pages and sites.



Quick Reference Library

HTML Quick Reference

Copyright 1999-2002 by Infinite Software Solutions, Inc. All rights reserved.

Trademark Information.

Welcome to the **DevGuru HTML Quick Reference** guide. This is a handy 163 page reference source that defines and explains all of the HTML version 4.01 tags and their associated attributes. In addition, useful, real world, working examples of code are provided for each tag. Plus, there are write-ups, with code examples, for the 16 core attributes which are available for almost all tags. Browser compatibility for each tag is also included.

HTML is the acronym for the Hypertext Markup Language, which is one of the most widely used computer languages in the world. The popularity and importance of HTML is due to the fact that it is the coding technology used to publish content on the World Wide Web (also referred to as the Internet).

Fortunately, HTML is both a user friendly language and very easy to learn. Here are some [helpful hints](#) to get you started.

The **Guru** has also created a [Color Chart](#) that displays all of the named HTML colors. In addition the hexadecimal code is provided for 256 colors.

On December 24, 1999, the World Wide Web Consortium (W3C) released a set of standards for the latest version of HTML, version 4.01. Note that 4.01 is considered to be a subversion of 4.0 and was published to correct some errors and to make minor revisions. This Quick Reference adheres to these W3C 4.01 standards. No proprietary style tags are documented, but for your reference, we provide a brief list and description for [other HTML tags](#) that are not part of the HTML 4.01 standard.

Over 30 years ago, in the late 1960s, the Department of Defense funded the creation of a computer network called ARPAnet. The goal was to encourage the free exchange of engineering and scientific information between defense contractors and academic institutions. The concept proved viable and the network slowly expanded during the 1970s and 1980s.

It took the advent of two additional technologies to bring the budding Internet to full flower.

First came the development of the HTML computer language. HTML was proposed in a paper penned in 1989 by Dr. Tim Berners-Lee, a physicist at CERN in Switzerland. It was derived as a subset of the Standard Generalized Markup Language (SGML), a computer language that had been created in 1986, but was little known. HTML was specifically designed to send electronic documents across a network. This language uses embedded tags that provide directions on how to display the contents of the document. HTML version 1.1 specifications were published in January 1992.

Second was the release in February 1993 of the first web browser, called Mosaic, which enabled the display of an HTML document on a computer monitor. Mosaic was written by students and faculty at the National Center for Supercomputing Applications (NCSA) at the University of Illinois.

By the mid 1990s, it became evident that the HTML language needed to be formally standardized. To this end, the Internet Engineering Task Force published such a standard on November 24, 1995 that described HTML version 2.0. The authors were Tim Berners-Lee and Dan Connolly.

For your historical interest, here is a [list of version 2.0 tags](#). Note the absence of such familiar tags as center, font, and table.

In all fairness, version 2.0 was more of a "guideline" rather than a "standard." It took the birth

of the World Wide Web Consortium and the subsequent publication of the "HTML 3.2 Reference Specifications" in January 1997 to finally provide coding standards that were widely recognized and implemented by the computer industry. Indeed, version 3.2 still remains the de facto international standard for HTML.

In the second half of the 1990s, many new technologies, such as Cascading Style Sheets (CSS) and Active Server Pages (ASP), were introduced and had a significant effect on the Internet. During this same time period, W3C continued to update HTML. The specifications for version 4.0 were released in December 1997, were later revised in April 1998, and were finally followed by the publication of version 4.01 just in time for the millennium.

In the foreseeable future, HTML will continue to serve as the basic structure for the Internet. However, a whole suite of new technologies, with acronyms such as XML, XHTML, DOM2, WAP, WML, and ASP+ will surely have a profound effect on the Internet. Already, W3C has released the XHTML version 1.0 standard which is designed to bridge HTML 4.0 and XML, and version 1.1 is on the horizon.

So stay tuned, the future of HTML is guaranteed to be very exciting.

TAG: a

`<a> ... `

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The `<a>` tag is called the anchor tag and it is most commonly used with the `href` attribute to create a hypertext link (usually referred to as just a link). By simply clicking on the link with a mouse or using a keyboard command, you can travel from one page to another page on the same web site or a different one. By default, the new page is normally displayed in the same window or frame, unless you specify otherwise using the `target` attribute.

Note that links cannot be nested. An `<a>...` element cannot contain other `a` elements.

You can insert characters, images, line breaks (`
` or `<p>`), and text between the opening and closing `a` tags. It is recommended that you should not insert Cascading Style Sheets code or any other HTML tags between the opening and closing tags. Rather, place all such tags outside of the `a` element.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#) [onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#) [title](#)

Attributes

accesskey

The `accesskey` attribute allows you to designate a key on the keyboard that when pressed, along with the alt or meta key, activates a link. This attribute is poorly supported.

charset

The `charset` attribute is used to specify the character encoding used on the page that is the target of the link. Character encoding defines how a sequence of bytes is to be converted into characters for display.

coords

The `coords` attribute is currently not supported by most browsers. It is used to define an area of influence around the `a` tag.

href

The `href` attribute is used to specify the target URL address for a link.

hreflang

The `hreflang` attribute can only be used when the `href` attribute is also being used. It specifies the language of the page that is the target of the link.

name

The `name` attribute is a string of characters that is used to label an element with a name. The name must be unique to that document and cannot be reused. The name can be the target of a link.

onblur

The **onblur** attribute is an event that allows a JavaScript code to execute when an element loses focus (for example, the mouse was clicked onto another element or a tab navigation directed the cursor elsewhere).

onfocus

The **onfocus** attribute is an event that allows a JavaScript code to execute when an element comes into focus (for example, the mouse was clicked onto the element or a tab navigation brought the cursor to the element).

rel

The **rel** attribute is currently not fully supported by most browsers. It is a space-separated list of one or more values that specify the relationship from the source page to the target for a link. Some of the proposed values are, **appendix**, **bookmark**, **chapter**, **contents**, **copyright**, **glossery**, **help**, **index**, **next**, **prev**, **section**, **stylesheets**, and **subsection**.

rev

The **rev** attribute is currently not fully supported by most browsers. It is a space-separated list of one or more values that specify the relationship from the target page to the source for a link. Perhaps the most useful value is **relation**.

shape

The **shape** attribute is currently not supported by most browsers. It is used to define an area of influence around the **a** tag.

tabindex

The **tabindex** attribute specifies an integer that defines the rank in the tabbing order for the specified element when you use the keyboard to navigate (tab through) a page. This attribute is poorly supported.

target

The **target** attribute specifies the name of the frame or window in which the target page should appear when a link is clicked. The four reserved names are **_blank**, **_parent**, **_self**, and **_top**.

type

The **type** attribute specifies the content type of the target page of a link. It has a MIME encoding type value.

This example demonstrates the code for creating a link. The **target** attribute is used to open the linked document in a new window.

Code:

```
<a href="http://www.devguru.com" target="_blank">DevGuru</a>
```

Output:

[DevGuru](http://www.devguru.com)

CORE ATTRIBUTE: class

class="name"

Available in versions: 4.0

HTML 4.0 tag compatibility: All tags except **base**, **basefont**, **head**, **html**, **meta**, **param**, **script**, **style**, **title**

The **class** core attribute is used to assign the name of a style sheets class to a tag.

For example, all of the code samples on the **DevGuru** site are in blue colored text. This is done by enclosing the code sample inside a pair of **span** tag elements and designating a class of "CODE". In turn, the "CODE" class has been assigned a text color of blue in the style sheets file (a .css file) that is located in the Include/ directory on the **DevGuru** web site. (Thus, only one file is needed to define the appearance of the entire site.)

Code:

```
<span class="CODE">  
The code example is blue.  
</span>
```

Output:

The code example is blue.

The exact same effect can be accomplished using other permitted tags.

Code:

```
<font class="CODE">  
This text is blue.  
</font>  
This text is not blue.
```

Output:

This text is blue.

This text is not blue.

CORE ATTRIBUTE: dir

dir="direction"

Available in versions: 4.0

HTML 4.0 tag compatibility: All tags except **applet, base, basefont, bdo, br, frame, frameset, iframe, param, script**

The **dir** core attribute is used to declare the direction that the text should run, either left to right (default) or right to left. This allows the internationalization of **HTML** since several important foreign languages read right to left, such as Chinese and Hebrew.

Also, see the **lang** core attribute that defines the language being used. However, note that using **lang** by itself does not specify language direction.

If used as an attribute of the **HTML** tag, it will apply to the entire document. When used with other tags, it only applies to the text under the influence of that tag. For example, when used with the **table** tag, the first column of the table will start on the right side and each additional column will be farther to the left.

This attribute should not be used to simply reverse a string. That is best done by using the VBScript StrReverse() function.

You may only use the following values:

dir="ltr"

The **ltr** value, which directs the text to flow from left to right, is the default.

dir="rtl"

The **rtl** value directs the text to flow from right to left.

Note that .html texts that contain foreign languages that use special characters will have to be saved as a Unicode file, rather than as an ANSI file, in order for the characters to be properly displayed. Please be aware that some browsers may not be capable of correctly displaying a Unicode file.

Code:

```
<html dir="rtl" lang="he">  
<body>
```

place text of right to left language here

```
</body>  
</html>
```

CORE ATTRIBUTE: id

id="name"

Available in versions: 4.0

HTML 4.0 tag compatibility: All tags except **base, head, html, meta, script, style, title**

The **id** core attribute is used to assign a name or id to a tag. This **id** must be unique within the document and can only be assigned to one use of a tag.

Because it is unique, the name of the **id** can be the target of a URL or used to define a style rule.

The use of **id** does not effect the output in any manner.

Code:

```
<input type="checkbox" id="ckbx1">
```

```
<input type="checkbox" id="ckbx2">
```

CORE ATTRIBUTE: lang

`lang="languagecode"`

Available in versions: 4.0

HTML 4.0 tag compatibility: All tags except **applet**, **base**, **basefont**, **br**, **frame**, **frameset**, **iframe**, **param**, **script**

The **lang** core attribute is used to define the base language to be used for displaying text and characters on a Web site. This allows an internationalization of **HTML** for a very large number of languages.

The languages are designated by a two letter code, such as "en" for English or "el" for Greek. One or more hyphenated values can be tacked on to the initial two letter code to specify regional or ethnic variations, such as "en-us" for U.S. English.

Note that .html texts that contain foreign languages that use special characters will have to be saved as a Unicode file, rather than as an ANSI file, in order for the characters to be properly displayed. Please be aware that some browsers may not be capable of correctly displaying a Unicode file.

Code:

```
<span lang="el">
```

place text of foreign language here

```
</span>
```

CORE ATTRIBUTE: onclick

onclick="action"

Available in versions: 4.0

HTML 4.0 tag compatibility: All tags except **applet, base, basefont, bdo, br, font, frame, frameset, head, html, iframe, isindex, meta, param, script, style, title**

The **onclick** core attribute is an event handler that is associated with the single click of a mouse on a selected HTML tag (element). Whenever you single click the element, the script code of the **onclick** is executed. This script can also call functions or subroutines which contain code that you want to run when the single click occurs.

The similar **ondblclick** core attribute is an event handler that is associated with the double click of a mouse.

In this example, every time you single click on the **button** element, one line of JavaScript code is executed which causes an alert box to be displayed with a message.

Code:

```
<input type="button" onclick="JavaScript: alert( 'onclick event' )" value="Click On This Button">
```

CORE ATTRIBUTE: ondblclick

ondblclick="action"

Available in versions: 4.0

HTML 4.0 tag compatibility: All tags except **applet, base, basefont, bdo, br, font, frame, frameset, head, html, iframe, isindex, meta, param, script, style, title**

The **ondblclick** core attribute is an event handler that is associated with the double click of a mouse on a selected HTML tag (element). Whenever you double click the element, the script code of the **ondblclick** is executed. This script can also call functions or subroutines which contain code that you want to run when the double click occurs.

The similar **onclick** core attribute is an event handler that is associated with the single click of a mouse.

In this example, every time you double click the **button** element, one line of JavaScript code is executed which causes an alert box to be displayed with a message.

Code:

```
<input type="button" ondblclick="JavaScript: alert( 'ondblclick event' )" value="Double Click Me!">
```

CORE ATTRIBUTE: onkeydown

onkeydown="action"

Available in versions: 4.0

HTML 4.0 tag compatibility: All tags except **applet, base, basefont, bdo, br, font, frame, frameset, head, html, iframe, isindex, meta, param, script, style, title**

The **onkeydown** core attribute is an event handler that is associated with pressing down a key on the keyboard while the mouse is inside a selected HTML tag (element). In other words, the key down event occurred while the HTML element was in focus. Whenever you press down the key, the script code of the **onkeydown** is executed. This script can also call functions or subroutines which contain code that you want to run when this event occurs. For example, if you only want numbers to be entered, you could capture the value of the key and take appropriate action if a non-numeric key was pressed.

The similar **onkeyup** core attribute is an event handler that is associated with the release of a key after it was pressed down, and the **onkeypress** core attribute is an event handler that is associated with the pressing of a key.

In this example, place the mouse inside the **textarea** element and type in a word. Every time you press a key down, one line of JavaScript code is executed which causes an alert box to be displayed with a message. Note how the alert comes up before the character is displayed inside the **textarea**.

Code:

```
<textarea onkeydown="javascript: alert('A key was pressed down')">
</textarea>
```

CORE ATTRIBUTE: onkeypress

onkeypress="action"

Available in versions: 4.0

HTML 4.0 tag compatibility: All tags except **applet, base, basefont, bdo, br, font, frame, frameset, head, html, iframe, isindex, meta, param, script, style, title**

The **onkeypress** core attribute is an event handler that is associated with press of a key on the keyboard while the mouse is inside a selected HTML tag (element). In other words, the key press event occurred while the HTML element was in focus. Whenever you press the key, the script code of the **onkeypress** is executed. This script can also call functions or subroutines which contain code that you want to run when this event occurs. For example, if you only want numbers to be entered, you could capture the value of the key and take appropriate action if a non-numeric key was pressed.

The similar **onkeydown** core attribute is an event handler that is associated with the pressing down of a key, and the **onkeyup** core attribute is an event handler that is associated with the release up of a key.

In this example, place the mouse inside the **textarea** element and type in a word. Every time you press a key, one line of JavaScript code is executed which causes an alert box to be displayed with a message. Note that the character does not appear in the **textarea** until after you have clicked okay in the alert.

Code:

```
<textarea onkeypress="javascript: alert('A key was pressed')">
</textarea>
```

CORE ATTRIBUTE: onkeyup

onkeyup="action"

Available in versions: 4.0

HTML 4.0 tag compatibility: All tags except **applet, base, basefont, bdo, br, font, frame, frameset, head, html, iframe, isindex, meta, param, script, style, title**

The **onkeyup** core attribute is an event handler that is associated with releasing up a key on the keyboard while the mouse is inside a selected HTML tag (element). In other words, the key up event occurred while the HTML element was in focus. Whenever you release the key, the script code of the **onkeyup** is executed. This script can also call functions or subroutines which contain code that you want to run when this event occurs. For example, if you only want numbers to be entered, you could capture the value of the key and take appropriate action if a non-numeric key was pressed.

The similar **onkeydown** core attribute is an event handler that is associated with the pressing down of a key, and the **onkeypress** core attribute is an event handler that is associated with the pressing of a key.

In this example, place the mouse inside the **textarea** element and type in a word. Every time you release a key, one line of JavaScript code is executed which causes an alert box to be displayed with a message. Note how the alert comes up after the character is displayed inside the **textarea**.

Code:

```
<textarea onkeyup="javascript: alert('A key was released up')">
</textarea>
```

CORE ATTRIBUTE: onmousedown

onmousedown="action"

Available in versions: 4.0

HTML 4.0 tag compatibility: All tags except **applet, base, basefont, bdo, br, font, frame, frameset, head, html, iframe, isindex, meta, param, script, style, title**

The **onmousedown** core attribute is an event handler that is associated with clicking down the mouse button while the mouse is inside a selected HTML tag (element). In other words, the mouse down event occurred while the HTML element was in focus. Every time you click down the mouse, while inside the element, the script code of the **onmousedown** is executed. This script can also call functions or subroutines which contain code that you want to run when this event occurs.

The four other similar core attributes are: **onmousemove, onmouseout, onmouseover, and onmouseup**.

In this example, place the mouse inside on the **button** element, click down, and hold it. Every time you click down, one line of JavaScript code is executed which causes an alert box to be displayed with a message. Finally, release the mouse and click okay in the alert.

Code:

```
<input type="button" onmousedown="JavaScript: alert( 'onmousedown event' )" value="Click  
Down With Mouse!">
```

CORE ATTRIBUTE: onmousemove

onmousemove="action"

Available in versions: 4.0

HTML 4.0 tag compatibility: All tags except **applet, base, basefont, bdo, br, font, frame, frameset, head, html, iframe, isindex, meta, param, script, style, title**

The **onmousemove** core attribute is an event handler that is associated with moving the mouse around while inside a selected HTML tag (element). In other words, the mouse move event occurred while the HTML element was in focus. Every time you move the mouse around the element, the script code of the **onmousemove** is executed. This script can also call functions or subroutines that contain code that you want to run when this event occurs.

The four other similar core attributes are: **onmousedown, onmouseout, onmouseover,** and **onmouseup.**

In this example, simply move the mouse inside on the **button** element.

Code:

```
<input type="button" onmousemove="JavaScript: alert( 'onmousemove event' )" value="Move  
The Mouse Inside This Button!">
```

CORE ATTRIBUTE: onmouseout

onmouseout="action"

Available in versions: 4.0

HTML 4.0 tag compatibility: All tags except **applet, base, basefont, bdo, br, font, frame, frameset, head, html, iframe, isindex, meta, param, script, style, title**

The **onmouseout** core attribute is an event handler that is associated with moving the mouse out of a selected HTML tag (element). In other words, the mouse out event occurred while the HTML element was in focus. Every time you move the mouse out of the element, the script code of the **onmouseout** is executed. This script can also call functions or subroutines which contain code that you want to run when this event occurs.

The four other similar core attributes are: **onmousedown, onmousemove, onmouseover,** and **onmouseup**.

In this example, simply place the mouse inside the **button** element and then move it out. As soon as you move out, an alert appears with a message.

Code:

```
<input type="button" onmouseout="JavaScript: alert( 'onmouseout event' )" value="Move The Mouse Out Of This Button! ">
```

CORE ATTRIBUTE: onmouseover

onmouseover="action"

Available in versions: 4.0

HTML 4.0 tag compatibility: All tags except **applet, base, basefont, bdo, br, font, frame, frameset, head, html, iframe, isindex, meta, param, script, style, title**

The **onmouseover** core attribute is an event handler that is associated with moving the mouse over a selected HTML tag (element). In other words, the mouse over event occurred while the HTML element was in focus. Every time you move the mouse over the element, the script code of the **onmouseover** is executed. This script can also call functions or subroutines which contain code that you want to run when this event occurs.

The four other similar core attributes are: **onmousedown, onmousemove, onmouseout, and onmouseup**.

In this example, simply move the mouse over on the **button** element. As soon as you move over, an alert is displayed with a message.

Code:

```
<input type="button" onmouseover="JavaScript: alert( 'onmouseover event' )" value="Move  
The Mouse Over This Button!">
```

CORE ATTRIBUTE: onmouseup

onmouseup="action"

Available in versions: 4.0

HTML 4.0 tag compatibility: All tags except **applet, base, basefont, bdo, br, font, frame, frameset, head, html, iframe, isindex, meta, param, script, style, title**

The **onmouseup** core attribute is an event handler that is associated with the releasing up portion of a click of the mouse while inside a selected HTML tag (element). In other words, the mouse up event occurred while the HTML element was in focus. Every time you release the mouse up, while in the element, the script code of the **onmouseup** is executed. This script can also call functions or subroutines which contain code that you want to run when this event occurs.

The four other similar core attributes are: **onmousedown, onmousemove, onmouseover,** and **onmouseout**.

In this example, simply place the mouse inside the **button** element, click down, then release up. As soon as you release up, an alert appears with a message.

Code:

```
<input type="button" onmouseup="JavaScript: alert( 'onmouseup event' )" value="Click And Release The Mouse While Inside This Button!">
```


CORE ATTRIBUTE: title

title="string"

Available in versions: 4.0

HTML 4.0 tag compatibility: All tags except **base**, **basefont**, **head**, **html**, **meta**, **param**, **script**, **style**, **title**

The **title** core attribute is used to assign a name to a tag. This name can be any string of characters or words. Further, you can use the HTML character entities in the name. For example, you could use `
` which will cause a line break (just like the **br** tag) in the name.

If you wish, you can reuse the same title values repeatedly inside an HTML document, or you could have each value be unique. In contrast, the **id** attribute requires that each id value must be unique.

Do not confuse this attribute with the **title** tag which is used to assign a title to a web document.

The **title** attribute is used by ToolTips on Internet Explorer to create a small window that automatically displays the text in the **title** when you run the mouse over a hyperlink or a mapped hyperlink image. This can be done with both the **a** and **area** tags. By default, this window is only displayed for about ten seconds. (This concept can also be applied to Netscape by using JavaScript functions. Please read the KB article for details.)

Code:

[Please refer to the DevGuru Knowledge Base article:](#)

```
<br>
<A href="/features/knowledge_base/a100216.html" title="ToolTips are &#10;&#13;really neat!">
A100216: Adding ToolTips to your HTML pages.
</a>
```

Output:

[Please refer to the DevGuru Knowledge Base article:](#)

[A100216: Adding ToolTips to your HTML pages.](#)

As another example, consider the situation where you have a large number of tables on your web page. By placing a **title** in both the opening and closing **table** tags, it becomes much easier to identify and keep track of the start and stop of individual tables inside your code.

Code:

```
<table title="PhoneNumbers">
... table entries
</table title="PhoneNumbers">
```

TAG: abbr

<abbr> ... </abbr>

Available in versions: 4.0

Browser compatibility: None

The **<abbr>** tag is used to signify that the enclosed text is an abbreviation of a longer word or phrase. This tag is very poorly supported and most browsers simply ignore it.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None

Code:

```
<abbr>lb</abbr> is an abbreviation for pound
```

Output:

lb is an abbreviation for pound

TAG: acronym

<acronym> ... </acronym>

Available in versions: 4.0

Browser compatibility: None

The **<acronym>** tag is used to signify that the enclosed text is an acronym which is a special type of abbreviation formed by using the first letter in each word of a phrase. This tag is very poorly supported and most browsers simply ignore it.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None

Code:

`<acronym>MIME</acronym>` means Multipurpose Internet Mail Extension

Output:

MIME means Multipurpose Internet Mail Extension

TAG: address

<address> ... </address>

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<address>** tag is used to signify that the enclosed text is an address. Most browsers display the address in a special manner, such as in italics. However, you can use style sheets to alter the appearance as you see fit. The text between the **address** tags can contain any HTML tag.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None

Code:

```
<address>  
DevGuru<br>  
5106 Kingston Pike<br>  
Knoxville, TN 37919  
</address>
```

Output:

```
DevGuru  
5106 Kingston Pike  
Knoxville, TN 37919
```

TAG: applet deprecated 4.0

<applet> ... </applet>

Available in versions: 3.2

Browser compatibility: Explorer 4, 5 Netscape 6

The **<applet>** tag is officially deprecated effective with HTML 4.0. Instead, you are now to use the **object** tag. However, this tag remains in wide use on the Internet and is recognized by all major browsers.

The **applet** tag is used to download and execute a Java applet. The applet is treated like an image and is rendered into the flow of the text without a line break. You can define the size of the region (the applet window) in which the applet will appear. By using the attributes for **applet**, style sheets code, and other HTML tags, you can create a presentation and appearance for the applet in the text flow to suit your needs.

Java applets can be very slow to load and may not display in many browsers.

Many Java applets require that document-specific parameters be provided to control the execution. You can use the **param** tag, inserted between the opening and closing **applet** tags, to define these parameters.

The closing tag is mandatory.

Core Attributes

[class](#) [id](#) [style](#) [title](#)

Attributes

align

The **align** attribute is used to position the applet window with respect to any surrounding text, images, or other applets.

alt

The **alt** attribute provides a text message that will be displayed (in place of the applet) on browsers that cannot display an applet.

archive

The **archive** attribute is a comma-separated list of URLs of Java classes that must be pre-loaded into the local user cache before the applet can be displayed.

code

The **code** attribute is mandatory and specifies the relative URL of the Java class (a .class type file) that must be executed to display the applet. Specifically, the URL is relative to the applet base URL (otherwise the applet will not load).

codebase

The **codebase** attribute is used to specify the URL of the directory where the Java class is stored. If the directory is different from where the HTML code is stored, you must use this attribute.

height

The **height** attribute is used to set the vertical extent of the applet in pixels.

hspace

The **hspace** attribute is used to add additional blank space in pixels to each side of the applet.

mayscript

The **mayscript** attribute is only recognized by Netscape. It permits the applet to access JavaScript code.

name

The **name** attribute is used to assign a unique string of characters as the name of the applet.

object

The **object** attribute is not supported by most browsers. It is the name of a resource where a serialized version of the applet is stored. Do not confuse this attribute with the **object** tag.

vspace

The **vspace** attribute is used to add additional blank space in pixels both above and below the applet.

width

The **width** attribute is used to set the horizontal extent of the applet in pixels.

This example displays a Java applet clock.

Code:

```
<applet codebase="clock/classes" code="JavaClock.class" width="150" height="150">
</applet>
```

TAG: area

<area> ... </area>

Available in versions: 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<area>** tag is used (along with the **<map>** tag) to create a client-side image map which is divided into two or more mouse-sensitive regions. When you click on a region in the map, an attribute contained in the **area** tag can direct an action to occur. For example, a hyperlink can be created in each region using the **href** attribute. When you click onto a region, the hyperlink will send you to a target page just like any link. Also, since the map is mouse-sensitive, you could also use any of the onmouse core attributes to capture a mouse event.

The **area** tag must be contained inside the **map** element. The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#) [onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#) [title](#)

Attributes

accesskey

The **accesskey** attribute allows you to designate a key on the keyboard that when pressed, along with the alt or meta key, activates a link in a region of the image map. This attribute is poorly supported.

alt

The **alt** attribute allows you to associate a text message with each individual region of the map. When the mouse passes over the region, a small window will display the text.

coords

The **coords** attribute defines the (x,y) coordinates of each shape in the image map. The coordinates are the x and y offset in pixels from the upper left corner (0,0) of the image. The number of coordinates you need to provide depends of the shape specified by the **shape** attribute.

href

The **href** attribute is the URL address for the link. Each **area** tag in the image map can have a link.

nohref

The **nohref** attribute specifies that the region defined by the **area** tag cannot be a link.

onblur

The **onblur** attribute is an event that allows a JavaScript code to execute when an element loses focus (for example, the mouse was clicked onto another element or a tab navigation directed the cursor elsewhere).

onfocus

The **onfocus** attribute is an event that allows a JavaScript code to execute when an element comes into focus (for example, the mouse was clicked onto the element or a tab navigation brought the cursor to the element).

shape

The **shape** attribute defines the geometric shape of the image map. It can be a circle (**circle** or **circ**), polygon (**polygon** or **poly**), or a rectangle (**rectangle** or **rect**)

tabindex

The **tabindex** attribute specifies an integer that defines the rank in the tabbing order for the specified element when you use the keyboard to navigate (tab through) a page. This attribute is poorly supported.

target

The **target** attribute specifies the name of the frame or window in which the target page should appear when a link is clicked. The four reserved names are **_blank**, **_parent**, **_self**, and **_top**.

The **area** and **map** tags are used on the **DevGuru** site to create image mapped links for the indexes to the various Quick References. For example, this is the code used for the ASP Quick Reference.

Code:

```

<br>
<map name="methodproperty">
<area shape="rect" coords="192,0,248,26" href="/technologies/asp/quickref/properties.html">
<area shape="rect" coords="133,0,191,26" href="/technologies/asp/quickref/collections.html">
<area shape="rect" coords="90,0,134,22" href="/technologies/asp/quickref/events.html">
<area shape="rect" coords="43,0,89,22" href="/technologies/asp/quickref/methods.html">
</map>
```

TAG: b

** ... **

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **** tag causes the designated text to be displayed in a bold weight. If bold is not available for a particular font, the browser may impose some other physical change to the text, such as underline.

This can also be done using the style sheets **font-weight** property.

The closing tag is mandatory.

Core attributes:

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None

Code:

```
<b>DevGuru</b> products are great!
```

Output:

DevGuru products are great!

TAG: base

<base>

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<base>** tag is used to define the base URL for the HTML document. This allows the use of relative URL addresses elsewhere in the document. Interestingly, this base URL does not have to be an absolute URL, but rather, it can be a relative URL.

Also, when a browser arrives from another site to your site via a link, you can use the **target** attribute of this tag to ensure that your site does not appear inside a frame or window of the previous site. In other words, your site will always appear as the top window.

The **base** tag must be placed between the opening and closing **head** tags.

There is no closing tag.

Core Attributes

none

Attributes

href

The **href** attribute is mandatory. It is used to define the base URL.

target

The **target** attribute specifies the default name of the frame or window in which all of the target pages should appear when a link is clicked on your site. However, this can be overridden by a **target** attribute that is inside an **a** tag. The four reserved names are **_blank**, **_parent**, **_self**, and **_top**.

In this example, the link in the **body** resolves to:

http://www.devguru.com/Technologies/HTML/quikref/html_index.html

Code:

```
<head>
<base href="http://www.devguru.com/">
</head>
<body>
<a href="../Technologies/HTML/quikref/html_index.html">HTML Index</a>
```

This code directs that your site always appears as the top window.

Code:

```
<head>
<base target="_top">
</head>
```

TAG: basefont deprecated 4.0

<basefont> ... </basefont>

Available in versions: 3.2

Browser compatibility: Explorer 4, 5 Netscape 4

The **<basefont>** tag is officially deprecated effective with HTML 4.0. It has been superseded by style sheets.

The **basefont** tag is used to set the font appearance. If placed between the **head** tags, the font values are set for the entire document. This tag can also be placed anywhere inside the **body** of the document where it can be used to set the font values for a local block of text. Under these circumstances, the **basefont** tag in the **head** is overridden.

The closing tag is optional. When the closing tag is used, it causes the font size to revert to the default value of 3. Therefore the following are equivalent:

Code:

```
</basefont>  
<basefont size="3">
```

Core Attributes

[id](#)

Attributes

color

The **color** attribute is used to set the color of the text. This attribute is not recognized by some browsers.

face

The **face** attribute is used to set the type (font face) of text, such as arial or courier. This attribute is not recognized by some browsers.

size

The **size** attribute is used to set the size of the text. The range is 1 to 7, and the default is 3. When you use + or - values to set the size using the **font** tag, the new size will be relative to the default value of 3, or to the value set using this attribute.

In this example, only the word DevGuru is affected by the tag. Beware that the **color** and **face** attributes are not recognized by some browsers.

Code:

```
<basefont color="red" face="arial" size="7">DevGuru</basefont> is the best!
```

Output:

DevGuru is the best!

TAG: bdo

<bdo> ... </bdo>

Available in versions: 4.0

Browser compatibility: None

The **<bdo>** tag is used to change the direction that the text reads, from left to right (the default) over to right to left, or vice versa. The purpose of this tag is to allow HTML to correctly display a great variety of languages, such as Chinese and Hebrew (which are read right to left), in the same document.

The closing tag is mandatory.

Most browsers do not recognize this tag.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [style](#) [title](#)

Attributes

None.

Code:

```
<do dir="rtl">
```

Very few browsers support this tag.

```
</bdo>
```

Output:

Very few browsers support this tag.

TAG: big

<big> ... </big>

Available in versions: 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<big>** tag causes the designated text to be displayed in a larger font size. The actual amount of change will be determined by the browser. If the text is already at maximum size, this attribute will have no effect.

The closing tag is mandatory.

Core attributes:

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None

Code:

```
<small>think</small> very <big>BIG</big>
```

Output:

think very **BIG**

TAG: body

<body> ... </body>

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<body>** tag is used to signify the start of the contents of your HTML document.

If you are using frames, then you use the **frameset** tag in place of the **body** tag.

Effective with version 4.0, many of the regular attributes for this tag are deprecated in favor of using style sheets. However, since this is one of the most important HTML tags, you can be assured that all major browsers will continue to recognize the deprecated attributes.

The closing tag is optional. However, it is recommended that it always be included.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

alink deprecated 4.0

The **alink** attribute is used to set the color that the text of an active link will display when the mouse is clicked down, but not yet released. There is no default color.

background deprecated 4.0

The **background** attribute allows you to use an image for the background of your HTML document. For most browsers, the image can be a .gif or .jpg file. Some browsers recognize other types of image files. If the image is smaller than your document, the image will be tiled (repeated).

bgcolor

The **bgcolor** attribute is used to set the background color for the document. It can be a named color (such as red) or a six character hexadecimal RGB color code (such as #e26fa9).

link deprecated 4.0

The **link** attribute is used to set the color of the link. The default is #2222bb.

onload

The **onload** attribute is an event that allows a JavaScript code to execute after your HTML document has finished loading in a window.

onunload

The **onunload** attribute is an event that allows a JavaScript code to execute when your HTML document is unloaded.

text deprecated 4.0

The **text** attribute is used to set the color of the text. The default is #000000 (black).

vlink deprecated 4.0

The **vlink** attribute is used to set the color that the text of a link will display after it has been

visited. The default color is #8822bb.

This example demonstrates the code structure for creating an HTML document.

Code:

```
<html>  
<head>  
<title>  
... place title here  
</title>  
</head>  
<body>  
... place document contents here  
</body>  
</html>
```

TAG: br

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **
** tag is used to create a line break and is analogous to a carriage return. The flow of the display of the text and any images is halted on the current line, and the flow resumes on the next line starting at the left margin.

There is no closing tag.

Core Attributes

[class](#) [style](#) [title](#)

Attributes

clear **deprecated 4.0**

The **clear** attribute is used to control the flow of text around an image. The permitted values are **all**, **left**, or **right**. This allows you to continue the text after all left-justified images, or after all right-justified images, or after all images. This attribute is deprecated effective with version 4.0. You are now to use style sheets.

Code:

```
My pseudotrophus "Acei" cichlid came from Lake Malawi in Africa.
```

```
<br>
```

```
The Showa Koi in my pond came from Japan.
```

Output:

```
My pseudotropheus "Acei" cichlid came from Lake Malawi in Africa.
```

```
The Showa Koi in my pond came from Japan.
```

TAG: button

<button> ... </button>

Available in versions: 4.0

Browser compatibility: Explorer 4, 5 Netscape 6

The **<button>** tag is used to create a button form control (field). Control is a technical term that refers to the various elements (buttons, check boxes, radio buttons, text areas) that can be used inside a form to gather information.

The four tags that can be used to build a **form** are:

[button](#) [input](#) [select](#) [textarea](#)

Specifically, the **button** tag is used to create a button in a **form** that can have content, consist of an image, and have the illusion of 3-dimensions. The buttons created using the **input** tag, including the type="image", cannot have content or display with a 3-D appearance. (Otherwise, there is little difference.)

Any text or images placed between the opening and closing **button** tags will be displayed on the button. The text can include most HTML tags and they will perform as expected. The one major exception is that you cannot display an image map using the **map** and **area** tags.

A **button** element should not contain other **button** tags, nor should it contain **fieldset**, **form**, **iframe**, **input**, **isindex**, **label**, **select**, or **textarea** tags.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

accesskey

The **accesskey** attribute allows you to designate a key on the keyboard that when pressed, along with the alt or meta key, will bring focus to the **button** element. This attribute is poorly supported.

disabled

The **disabled** attribute is a Boolean value that, if present, prevents the **button** from functioning. In some browsers, the **button** will appear to be greyed out.

name

The **name** attribute is used to assign a unique string of characters as the name of the **button**.

onblur

The **onblur** attribute is an event that allows a JavaScript code to execute when an element loses focus (for example, the mouse was clicked onto another element or a tab navigation directed the cursor elsewhere).

onfocus

The **onfocus** attribute is an event that allows a JavaScript code to execute when an element comes into focus (for example, the mouse was clicked onto the element or a tab navigation brought the cursor to the element).

tabindex

The **tabindex** attribute specifies an integer that defines the rank in the tabbing order for the specified element when you use the keyboard to navigate (tab through) a page. This attribute is poorly supported.

type

The **type** attribute defines the type of button. The permitted values are **button**, **reset**, or **submit**.

value

The **value** attribute assigns a value to the **button** element. This can be changed later using script code.

Code:

```
<form action="html_button.html" method="get">
<button type="button" onclick="JavaScript: alert('Image Button')">

</button>
<p>
<button type="button" onclick="JavaScript: alert('Text Button')">
Please Click This Button!
</button>
</form>
```

TAG: input

<input>

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<input>** tag is used to create individual form controls (fields). Control is a technical term that refers to the various elements (buttons, check boxes, radio buttons, text areas) that can be used inside a **form** to gather information.

The four tags that can be used to build a **form** are:

[button](#) [input](#) [select](#) [textarea](#)

The **form** is the heart of an e-commerce page and the **input** tag is the heart of a **form**. There are ten types of form controls that can be created using the **input** tag. Appropriately, you must use the **type** attribute to determine the specific type of the control (field). The types are:

[button](#)
[checkbox](#)
[file](#)
[hidden](#)
[image](#)
[password](#)
[radio](#)
[reset](#)
[submit](#)
[text](#)

Any number of **input** tags can be placed anywhere between a pair of opening and closing **form** tags to create the desired appearance of the form.

There is no closing tag.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

accept

The **accept** attribute is used with the **type="file"** control to specify what kind of files (including the path) that can be attached to the **form**. It is a comma-delimited list. For example:

accept="images/*.gif"

accesskey

The **accesskey** attribute allows you to designate a key on the keyboard that when pressed, along with the alt or meta key, will bring focus to the **input** form element. This attribute is poorly supported.

align deprecated 4.0

The **align** attribute is used with a **type="image"** control to place the image with respect to any surrounding text. The permitted values are **bottom**, **left**, **middle**, **right**, and **top**. This attribute is deprecated effective with version 4.0. However, most browsers continue to recognize this attribute. You are now to use style sheets.

alt

The **alt** attribute displays an alternative text for browsers that cannot display an image.

checked

The **checked** attribute is a Boolean value that, if present, allows a checkbox or radio button to be preselected (and have a preselected **value**).

disabled

The **disabled** attribute is a Boolean value that, if present, prevents the form control (field) from functioning. In some browsers, the control (field) will appear to be greyed out.

ismap

The **ismap** attribute is a Boolean value that, when present, signifies that the image is being used as a mouse-clickable server-side image map.

maxlength

The **maxlength** attribute sets the maximum number of characters that can be entered into a one line text window.

name

The **name** attribute is a string of characters that is used to label a form control (field) with a name. The name must be unique to that document and cannot be reused.

onblur

The **onblur** attribute is an event that allows the execution of JavaScript code when an element loses focus (for example, the mouse was clicked onto another element or a tab navigation directed the cursor elsewhere).

onchange

The **onchange** attribute is an event that allows the execution of JavaScript code when the form control (field) has been changed and loses focus.

onfocus

The **onfocus** attribute is an event that allows the execution of JavaScript code when an element comes into focus (for example, the mouse was clicked onto the element or a tab navigation brought the cursor to the element).

onselect

The **onselect** attribute is an event that allows the execution of JavaScript code when the form control (field) gains focus.

readonly

The **readonly** attribute allows you to display a value in a form control (field) that cannot be changed by the user.

size

The **size** attribute sets the width of a one line text window by defining how many characters can fit in the window.

src

The **src** attribute is the URL address or directory/file (path/name) where an image file is being stored.

tabindex

The **tabindex** attribute specifies an integer that defines the rank in the tabbing order for the specified element when you use the keyboard to navigate (tab through) a page. This attribute is poorly supported.

type

The **type** attribute determines the specific control.

Note that by using style sheets you can effect the appearance of the various types of controls. However, the exact appearance will be browser dependent.

type="button"

The **type="button"** control creates a rectangular button that can be clicked when you wish a desired action to occur. For example, you could use buttons to navigate through a **recordset** data source being displayed on a web page. (See the RDS objects in the ADO Quick Reference.) If you want to reset or submit the **form**, you should use the **type="reset"** and **type="submit"** controls.

A value for the **name** attribute is required. You can use the **value** attribute to assign a text that will be displayed on the face of the button. The default color is gray, but that can be changed using style sheets (see example at bottom of page). The text displayed on the button will set the minimum size, but you can set a different size using style sheets or the **size** attribute.

Code:

```
<input type="button" name="btn1" value="click me">
```

type="checkbox"

The **type="checkbox"** control creates a small, square button that can be selected or unselected by the user with a click of the mouse. When the checkbox is selected, a check mark appears within the box.

A value for the **name** attribute is required. You must use the **value** attribute to assign a value that is associated with the checkbox. You can also use the **checked** attribute to precheck the checkbox. When the checkbox has been selected, the **value** is saved as part of the form information that is passed to the appropriate file and/or database when the form is submitted. However, the **value** for any unselected checkbox is not saved or passed.

Code:

```
<input type="checkbox" name="ckbx1" value="true">
```

type="file"

The **type="file"** control is used to attach a file to a **form**. When the **form** is submitted, the file is passed along with all the other contents of the **form**.

This control displays an input window and a browse button. By clicking on the browse button, the user can search your directories for the desired file. When you find it, a double mouse click will select the file. Alternately, you can enter the path and name of the file into the input window. A value for the **name** attribute is required. You do not use the **value** attribute.

To use the **type="file"**, you must set both the **enctype** and **method** attributes of the **form** tag exactly as shown in the following code snippet.

Code:

```
<form method="post" enctype="multipart/form-data" action="next.asp">
```

```
...  
<input type="file" name="file1">
```

type="hidden"

The **type="hidden"** control is used to add content to the **form** that cannot be viewed or changed by the user. When the **form** is submitted, the hidden content is passed along with all the other contents of the **form**.

The hidden content is essentially a name/value pair. You must provide a name using the **name** attribute. You also must provide a **value**. This value can be any textual or numeric information.

Code:

```
<input type="hidden" name="SocSecNum" value="123-45-6789">
```

type="image"

The **type="image"** control allows you to use a graphic image or a picture as the equivalent of a **type="submit"** button. For example, you could create your own submit gif. Indeed, the specified image will perform exactly as a submit button, when you click the image, the **form** is submitted.

The **type="image"** also allows you to display a mouse-sensitive image map within a **form**. (This requires that the browser be able to capture and process the X/Y coordinates of the mouse.)

You must use the **src** attribute to list the location of the image file. You can use the **alt** attribute to provide text for browsers that cannot display an image. Some browsers, by default, place a border around the image.

Code:

```
<input type="image" name="submitgif" src="images/submit.gif" onclick="submitform()"  
border="0">
```

type="password"

The **type="password"** control displays a one line input window which is used to collect a password as a masked text. A masked text means that when the user types in the password, the actual characters being typed do not appear in the window. Rather an asterisk * is displayed in place of each character in the password.

You can use a **type="text"** control to gather the user name.

The default window size is browser dependent and is usually 20-30 characters in length. However, you can set the size using style sheets or the **size** attribute. The **maxlength** attribute can be used to set the maximum number of characters that will be accepted by the control (field).

This is only a semi-secure method for gathering a password. While the password is not visible on screen, when the form is submitted, the password is passed unencrypted as part of the contents of the form.

Code:

```
<input type="password">
```

type="radio"

The **type="radio"** control is used to create a small round circular button with a white center that can be selected or unselected by the user with a click of the mouse. When the radio button is selected, a black dot appears in the center of the circle.

The concept is to present the user with a selection of choices, but only to allow one choice to be made, for example, picking the color of a tee-shirt being ordered on a web site from a selection of twelve different colors.

Each related group of radio buttons must have the same **name**. Further, each radio button in a related group must have a different **value**. When a specific radio button is selected, only the **value** for the selected button is passed as part of the contents of the **form** upon submission. The values for the unselected radio buttons are not passed.

You cannot change the size or color of a radio button. You can use the **checked** attribute to preselect one of the radio buttons.

Code:

```
<input type="radio" name="radio1" value="one" checked>  
<input type="radio" name="radio1" value="two">  
<input type="radio" name="radio1" value="three">  
<input type="radio" name="radio1" value="four">
```

type="reset"

The **type="reset"** control creates a rectangular button that can be clicked when the user wants to clear all of the fields in a **form**. The **form** is redisplayed in its original presentation.

By default the reset button is a gray color and has the text "Reset" displayed inside the button. However, you can change the color using style sheets and you can use the **value** attribute to provide a different text for the button. The text displayed on the button will set the minimum size, but you can set a different size using style sheets or the **size** attribute.

A **name** is not required, but it is becoming a very common coding practice to give a unique name to every control (field) in a **form**.

Code:

```
<input type="reset" value="Clear The Form" name="reset1">
```

type="submit"

The **type="submit"** control creates a rectangular button that can be clicked when the user wants to submit the contents of a **form** for processing.

A **form** can be submitted to the same page or to another URL. This is set inside the **form** tag by using the **action** attribute. However, if it is important that certain contents of the **form** be filled out correctly, then you could use the **onclick** core attribute in the submit button to direct the submission to a JavaScript function that proofs the contents. Alternatively, you could use an **onsubmit** attribute inside the **form** tag to direct the submission to the JavaScript function. This is called client-side form verification.

By default the submit button is a gray color and has the text "Submit Query" displayed inside the button. However, you can change the color using style sheets and you can use the **value** attribute to provide a different text for the button. The text displayed on the button will determine the minimum size, but you can set a different size using style sheets or the **size** attribute.

A **name** is not required, but it is becoming a very common coding practice to give a unique

name to every control in a **form**.

If you assign a **value**, when the **form** is submitted, that value will be passed as part of the contents.

Code:
`<input type="submit" name="submit" onclick="submitform()" value="Submit">`

type="text"

The **type="text"** control displays a one line text input window into which the user can type the requested information (such as an address or phone number). If the requested information will require more than one line to enter, you should use a **textarea** tag rather than a **type="text"** control.

The default window size is browser dependent and is usually 20-30 characters in length. It is strongly recommended that you set the size of a text window rather than using the default. This will allow you to effect the appearance of the **form** in a pleasing and useful manner. Further, the appearance of the **form** will be fairly independent of the browser being used.

The default window size is browser dependent and is usually 20-30 characters in length. However, you can set the size using style sheets or the **size** attribute. The **maxlength** attribute can be used to set the maximum number of characters that will be accepted by the control (field). You can use the **value** attribute to display a text inside the window, but this is optional.

Code:
`<input type="text" name="text1" size="50">`

value

The **value** attribute assigns an initial value, such as a text or number, to a form control (field). This can be changed later using script code.

usemap

This **usemap** attribute specifies the name of an image map that you wish to associate with the element. This indicates that the image is being used as a mouse-clickable client-side image map.

Code:
MY SAMPLE FORM:

```
<p>
<form method="post" enctype="multipart/form-data" action="next.asp">
<input type="button" name="btn2" value="click me" style="background-color:yellow;"> Button
<p>
<input type="checkbox" name="ckbx2" value="true"> Check Box
<p>
<input type="file" name="file2"> Attach a File
<p>
<input type="hidden" name="SocSecNum" value="123-45-6789">
<p>
<input type="password"> Password
<p>
<input type="radio" name="radio2" value="one" checked> First Radio Button
<br>
<input type="radio" name="radio2" value="two"> Second Radio Button
<br>
<input type="radio" name="radio2" value="three"> Third Radio Button
```

```
<br>
<input type="radio" name="radio2" value="four"> Forth Radio Button
<p>
<input type="text" name="submit" name="text2" size="50"> Text
<br>
<input type="text" name="submit" name="text3" size="25"> Text
<br>
<input type="text" name="submit" name="text4" size="2"> Text
<p>
<input type="reset" value="Clear The Form" style="color:white; background-color:blue;"> Reset
Button
<br>
<input type="submit" name="submit" name="submit2" onclick="submitform()"
value="Submit" style="color:white; background-color:red;">
Submit Button Or Use An Image Submit Button
<input type="image" src="images/submit.gif" onclick="submitform()" name="submitgif">
border="0"
</form>

<script language="JavaScript">
function submitform()
{ alert("Thank You For Submitting The Form") }
</script>
```

TAG: select

`<select> ... </select>`

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<select>** tag is used to create a pull-down list form control (field). Control is a technical term that refers to the various elements (buttons, check boxes, radio buttons, text areas) that can be used inside a form to gather information.

The four tags that can be used to build a **form** are:

[button](#) [input](#) [select](#) [textarea](#)

The **select** tag is used to delimit the start and stop of a multiple choice pull-down menu. The default for this tag is to create a one line window with a button on the left side. (Optionally, you can display a multi-line window.) When you single click onto the button with a mouse, the one line window expands to display a list of items. You can then select one item from the list by simply clicking on the item with the mouse. (Optionally, you can allow more than one item to be selected.) Note that the user can select or unselect the item(s) any number of times. When an item(s) is selected, it is added to the contents of the **form** and when the **form** is submitted, the selected item(s) will be passed along with all of the other contents.

The beauty of the **select** form control is that it can occupy as little as one line of space on a **form**, yet allow the user to choose from a very large selection of items. For example, your pull-down could be a list of all fifty states. As another example, note the pull-down list at the top of this HTML Quick Reference page. When you select an item, it acts as a link and directs you to another page in this HTML Quick Reference.

You must use the **option** or the **optgroup** tags to define each item in the menu.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

disabled

The **disabled** attribute is a Boolean value that, if present, prevents the form control (field) from functioning. In some browsers, the control (field) will appear to be greyed out.

multiple

The **multiple** attribute allows the user to choose more than one item in the pull-down. The default is to be able to choose only one item.

name

The **name** attribute is a string of characters that is used to label a form control (field) with a name. The name must be unique to that document and cannot be reused.

onblur

The **onblur** attribute is an event that allows the execution of JavaScript code when an element loses focus (for example, the mouse was clicked onto another element or a tab navigation directed the cursor elsewhere).

onchange

The **onchange** attribute is an event that allows the execution of JavaScript code when the form control (field) has been changed and loses focus.

onfocus

The **onfocus** attribute is an event that allows the execution of JavaScript code when an element comes into focus (for example, the mouse was clicked onto the element or a tab navigation brought the cursor to the element).

size

The **size** attribute allows the user to have more than one item on display in the pull-down window. Up and down buttons will allow you to scroll through the list. The default is to display only one item.

tabindex

The **tabindex** attribute specifies an integer that defines the rank in the tabbing order for the specified element when you use the keyboard to navigate (tab through) a page. This attribute is poorly supported.

This is a simple pull-down that allows the choice of one item.

Code:

```
<select>
<option selected>Angel fish</option>
<option>Cat fish</option>
<option>Discus</option>
<option>Neon tetra</option>
</select>
```

Here is the code for the pull-down at the top of the online version of this page.

Code:

```
<select name="SeeAlsoSelect" onChange=ViewCrossReference(this); style="height: 22px;
width: 240px">
<option value="html_form.html" selected>form tag</option>
<option value="html_input.html">input tag</option>
<option value="html_button.html">button tag</option>
<option value="html_optgroup.html">optgroup tag</option>
<option value="html_option.html">option tag</option>
<option value="html_textarea.html">textarea tag</option>
</select>
<a href="javascript:ViewCrossReference(document.MainForm.SeeAlsoSelect)" border=0>

</a>
```

This JavaScript function is placed in the [head](#) element.

```
<script language="JavaScript">
function ViewCrossReference (selSelectObject)
{
  if (selSelectObject.options[selSelectObject.selectedIndex].value != "")
```

```
{
  location.href=selSelectObject.options[selSelectObject.selectedIndex].value
}
}
</script>
```

TAG: textarea

`<textarea> ... </textarea>`

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<textarea>** tag is used to create a multi-line text input window form control (field). Control is a technical term that refers to the various elements (buttons, check boxes, radio buttons, text areas) that can be used inside a form to gather information.

The four tags that can be used to build a **form** are:

[button](#) [input](#) [select](#) [textarea](#)

The **textarea** control should be used when the user needs to input more than one line of data. If you only expect the user to input a few words, you should use the **input** tag with **type="text"** control that creates a one line text input window.

Any text or HTML code that occurs between the opening and closing **textarea** tags will appear inside the textarea window. Therefore, the closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

accesskey

The **accesskey** attribute allows you to designate a key on the keyboard that when pressed, along with the alt or meta key, will bring focus to the **textarea** element. This attribute is poorly supported.

cols

The **cols** attribute sets how many characters will be visible across the width of the **textbox** window.

disabled

The **disabled** attribute is a Boolean value that, if present, prevents the form control (field) from functioning. In some browsers, the control (field) will appear to be greyed out.

name

The **name** attribute is a string of characters that is used to label a form control (field) with a name. The name must be unique to that document and cannot be reused.

onblur

The **onblur** attribute is an event that allows the execution of JavaScript code when an element loses focus (for example, the mouse was clicked onto another element or a tab navigation directed the cursor elsewhere).

onchange

The **onchange** attribute is an event that allows the execution of JavaScript code when the form control (field) has been changed and loses focus.

onfocus

The **onfocus** attribute is an event that allows the execution of JavaScript code when an element comes into focus (for example, the mouse was clicked onto the element or a tab navigation brought the cursor to the element).

onselect

The **onselect** attribute is an event that allows the execution of JavaScript code when the form control (field) gains focus.

readonly

The **readonly** attribute allows you to display a text or value in a form control (field) that cannot be changed by the user.

rows

The **rows** attribute sets how many rows will be displayed in the **textarea** window.

tabindex

The **tabindex** attribute specifies an integer that defines the rank in the tabbing order for the specified element when you use the keyboard to navigate (tab through) a page. This attribute is poorly supported.

wrap

The **wrap** attribute allows a text that is longer than the window size to be displayed on more than one line. This is called wrapping. The default is to not wrap. Permitted values are **off** (default), **physical**, and **virtual**. This is a Netscape proprietary attribute and is not part of version 4.01.

This example shows two **textarea** windows, the first is not sized and the second is sized.

Code:

```
<form action="html_textarea.html" method="post">
<textarea name="textarea12">
This is the default size.
</textarea>
<br>
<textarea cols="45" rows="4" name="textarea13">
</textarea>
</form>
```

TAG: head

`<head> ... </head>`

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<head>** tag is the HTML document header. It serves as a container for other tags that control the contents and appearance of the main body of the document. Therefore, its use is optional.

This tag is inserted immediately after the **html** tag, but before the **body** or **frameset** tags.

The **head** element can contain any of the following HTML tags in any order:

[base](#) set base URL

[basefont](#) set document font **deprecated 4.0**

[isindex](#) keywords for searching **deprecated 4.0**

[link](#) set document link

[meta](#) document keywords

[script](#) script code

[style](#) set style sheet rules

[title](#) name of document

The closing tag is mandatory.

Core Attributes

[dir](#) [lang](#)

Attributes

profile

The **profile** attribute is a list of one or more (comma separated) URL addresses of the meta data profiles. The format of these profiles is not defined in version 4.0.

Code:

```
<html>
<head>
<base href="http://www.devguru.com/">
<link rel="stylesheet" type="text/css" href="include/StylesDefinitions.css">
<title> DevGuru </title>
</head>
<body>
...
</body>
</html>
```

TAG: isindex **deprecated 4.0**

<index>

Available in versions: 2.0, 3.2

Browser compatibility: Explorer 4, 5 Netscape 4

The **<isindex>** tag is officially deprecated effective with HTML 4.0. Instead, you are now to use the **form** and **input** tags to create search forms. However, this tag remains in minor use on the Internet and is still recognized by all major browsers.

The **isindex** tag is used to create a search element with a prompt text and a one line input window that displays on the Web page. The exact appearance of the search element is browser dependent. For example, the default prompt text for IE 5.0 is:

You can search this index. Type the keyword(s) you want to search for:

The concept is that the user enters a space-separated list of one or more keywords into the search window and then hits the enter key. A search query URL is created that is composed of a URL, followed by a question mark, and then the list of keywords (each keyword is preceded by a plus sign). The default URL is the document URL. However, you can use the **base** tag to point to a search page. Whatever URL is designated must be able to handle a search query.

This tag can be placed in either the **head** or **body** element. However, you cannot place it inside a **form** element.

There is no closing tag.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [style](#) [title](#)

Attributes

prompt **deprecated 4.0**

The **prompt** attribute is used to write the text that you wish to appear for the search prompt. Each browser has a default text. This attribute is deprecated effective with version 4.0. However, most browsers continue to recognize this attribute.

Code:

```
<isindex>
```

TAG: link

<link>

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<link>** tag is used to establish a relationship between the current document and one or more other related documents. For example, if you have a Cascading Style Sheets .css file that defines the style for an entire web site, then each file in the site would be linked to the .css file using the **link** tag.

This tag can be used more than once (i.e., one link per tag). Each occurrence of this tag must be placed inside the **head** element. A value for the **href** attribute is required.

There is no closing tag.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#) [onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#) [title](#)

Attributes

charset

The **charset** attribute is used to specify the character encoding used on the page that is the target of the link. Character encoding defines how a sequence of bytes is to be converted into characters for display.

href

The **href** attribute is a valid URL address for another document. This attribute is required.

hreflang

The **hreflang** attribute can only be used when the href attribute is also being used. It specifies the language of the page that is the target of the link.

media

The **media** attribute specifies which medium the link apply to. Permitted values include: **all**, **aural**, **braille**, **handheld**, **print**, **projection**, **screen**, **tty**, and **tv**.

rel

The **rel** attribute is currently not fully supported by most browsers. It is a space-separated list of one or more values that specify the relationship from the source page to the target for a link. Some of the proposed values are, **appendix**, **bookmark**, **chapter**, **contents**, **copyright**, **glossary**, **help**, **index**, **next**, **prev**, **section**, **stylesheets**, and **subsection**.

rev

The **rev** attribute is currently not fully supported by most browsers. It is a space-separated list of one or more values that specify the relationship from the target page to the source for a link. Perhaps the most useful value is **relation**.

target

The **target** attribute specifies the name of the frame or window in which the target page should appear when a link is clicked. The four reserved names are **_blank**, **_parent**, **_self**, and **_top**.

type

The **type** attribute specifies the MIME (Multipurpose Internet Mail Extension) type of the linked document. Two permitted type values include: **text/css** and **text/javascript**.

This example demonstrates the code for creating two different types of links.

Code:

```
<head>  
<link rel="stylesheet" type="text/css" href="include/StylesDefinitions.css">  
<link rev="relation" href="http://www.fly-by-day-computing.com">  
<title> Fly-By-Night-Computing </title>  
</head>
```

TAG: meta

<meta>

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<meta>** tag is primarily used to list information (meta data) about the current HTML document. This information can be anything you want users to know about your site including author names, keywords, subject matter, and topics of interest. This meta data is used extensively by the various Internet search engines.

The **meta** tag can also be used with the **http-equiv** attribute to simulate HTTP headers.

This tag presents the information as name/content paired values by using the **name** and **content** attributes. You can have more than one **meta** element per page.

This tag can only be placed inside the **head** element. There is no closing tag.

Core Attributes

[dir](#) [lang](#)

Attributes

content

The **content** attribute is the content portion of the name/content value. There are no defined values for this attribute. You may choose any words or text you desire. For example, you could choose "keywords" for the **name** and then use the **content** attribute to list keywords that you feel are important to your site.

http-equiv

The **http-equiv** attribute can be used in place of the **name** attribute to create an http-equiv/content value pair. It provides information used to generate HTTP headers (however, most proxy servers ignore this attribute). The permitted values are **content-type**, **expires**, **refresh**, and **set-cookie**.

name

The **name** attribute is the name portion of the name/content value. There are no defined values for this attribute. You may choose any word you desire. For example, you could choose "keywords" for the **name** and then use the **content** attribute to list keywords that you feel are important to your site.

scheme

The **scheme** attribute sets the scheme (the procedure) for interpreting the meta data. This must agree with the profiles listed by the **profile** attribute of the **head** tag.

Code:

```
<head>
<meta name="author" content="The Guru">
<meta name="description" content="DevGuru Cascading Style Sheets Quick Reference is a 40
page reference that discusses the style properties that set specific visual parameters for a Web
page.">
<meta name="keywords" content="DevGuru, CSS, css, Cascading Style Sheet, cascading
style sheet, style sheet, background, border, floating elements, font, margin, padding, text">
```

```
<title>DevGuru Cascading Style Sheets Quick Reference</title>  
</head>
```

The **http-equiv** attribute can be used to update a page. In this example, the update occurs every thirty seconds.

```
<meta http-equiv="refresh" content="30">
```

TAG: script

`<script> ... </script>`

Available in versions: 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<script>** tag is used to place script code inside the **head** and **body** elements of an HTML code document. For all practical purposes, this tag is primarily used to allow the execution of JavaScript code inside HTML code (and, to a lesser extent, VBScript).

Unfortunately, not all browsers (especially older browsers) recognize this tag. It is recommended that you place an HTML opening comment tag right after the opening **script** tag and a closing comment tag immediately before the closing **/script** tag. A JavaScript enabled browser will ignore these comment tags and execute the code. Conversely, a browser that does not recognize JavaScript will treat the contents as simply a comment and not crash your page.

The closing tag is mandatory.

Core Attributes

None.

Attributes

charset

The **charset** attribute is used to define the character set that was used to write the script.

defer

The **defer** attribute informs the server that the script will not alter the contents of the page on the server-side. This is intended to allow the server to load the page faster. (In JavaScript, this means no document.write statements.)

language deprecated 4.0

The **language** attribute is the name of the script language, such as **VBScript**, **JavaScript**, **JavaScript 1.1**, and **JavaScript 1.2**. This attribute is deprecated effective with version 4.0. You are now to use the **type** attribute. However, most browsers continue to recognize this attribute.

src

The **src** attribute is the URL of a file that contains a script code. This is useful when you need to use repeatedly a long script program. The browser will load and execute the script as a separate file.

type

The **type** attribute is used to define the script language being used. The two most common values are **text/javascript** and **text/vbscript**

This example uses the **script** tag to place a JavaScript function in the **head** element to see if two form fields (firstname and lastname) have been left blank.

Code:

```
<head>  
<title>DevGuru JavaScript Example</title>
```

```
<script language="JavaScript">
<!--
function checksubmit()
{
  if (document.formname.firstname.value == "")
  {
    alert("Please enter your full name")
    document.formname.firstname.focus()
    return false
  }
  if (document.formname.lastname.value == "")
  {
    alert("Please enter your email address")
    document.formname.lastname.focus()
    return false
  }
  return true
}
-->
</script>
</head>
```

TAG: style

`<style> ... </style>`

Available in versions: 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<style>** tag is used to create document-level style sheet rules. It is referred to as document-level because the style rules will apply to the entire HTML page. In contrast, the **style** core attribute behaves as an inline style rule and only effects a small portion the HTML document.

The **style** tag must appear inside the **head** element. The code, contained between the opening and closing **style** tags, is not HTML, but is CSS (Cascading Style Sheets). The general concept is that a CSS property is assigned to a selector keyword, such as an HTML tag.

Unfortunately, not all browsers (especially older browsers) recognize this tag. It is recommended that you place an HTML opening comment tag right after the opening **style** tag and a closing comment tag immediately before the closing **/style** tag. A browser that recognizes CSS will ignore these comment tags and execute the code. Conversely, a browser that does not recognize CSS will treat the contents as simply a comment.

The closing tag is mandatory.

Core Attributes

[dir](#) [lang](#) [title](#)

Attributes

media

The **media** attribute is a comma-separated list of one or more types of media in which the HTML document may appear (**all**, **aural**, **braille**, **handheld**, **print**, **projection**, **screen**, **tv**, and **tty**). The default is **screen**. This attribute is very poorly implemented.

type

The **type** attribute is used to define the type of style being used. The two permitted values are **text/css** and **text/javascript**. Since there is no default value, it may prove wise to always include this attribute in the **style** tag.

Here is a simple example of creating document-level style rules. A series of CSS properties defining font face, color, and size is assigned to the **br**, **pre**, and **code** tags. For more information about style sheets, please click on the **Style Sheets** link in the left menu frame under the **Technologies** heading.

Code:

```
<style type="text/css">
<!--
br { font-face: arial; color: black; font-size: 12px" }
pre { font-face: arial; color: red; font-size: 18px" }
code { font-face: courier; color: blue; font-size: 15px" }
-->
</style>
```


TAG: title

<title> ... </title>

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<title>** tag is used to provide a name or a title for your HTML document. There can only be one **title** element per document. The title is usually displayed as a text header by the browser in a special title or status window. If you do not provide a title, many browsers will display a default title. However, it is to your advantage to provide a meaningful and descriptive title for your site. For example, titles are sometimes used as the default text for links.

The **title** must appear inside the **head** element. It cannot contain any other HTML tags, such as **br**. However, you can use character entities (such as ** ** for spacing).

The closing tag is mandatory.

Core Attributes

[dir](#) [lang](#)

Attributes

None.

Code:

```
<html>
<head>
<base href="http://www.devguru.com/">
<link rel="stylesheet" type="text/css" href="include/StylesDefinitions.css">
<title> DevGuru </title>
</head>
<body>
...
</body>
</html>
```

TAG: caption

<caption> ... </caption>

Available in versions: 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<caption>** tag allows you to place a comment (caption) with a table. The caption does not appear inside the table, but rather it appears outside the table on either the top (default) or bottom. Some browsers recognize proprietary attributes that permit placing the caption on the left or right side of the table. The **caption** tag must occur immediately after the **table** tag. You can only have one caption per table. The text that composes the caption can contain HTML tags and style sheet properties.

Note that the **table** tag has the **summary** attribute which is used with non-visual media such as braille and speech to provide information about a table.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

align deprecated 4.0

The **align** attribute allows you to specify if the caption is to appear on the **top** or **bottom** of the table. The default is top. This attribute is deprecated effective with version 4.0. However, most browsers continue to recognize this attribute. You are now to use style sheets.

This example demonstrates the basic table tags:

```
Code:  
<table width="80%" border="10" cellpadding="3" cellspacing="5" align="center">  
<caption>  
<center><b>How To Code A Table</b></center>  
<br>
```

Use the caption tag to name the table and to provide useful information about the contents
</caption>

```
<tr>  
<th colspan="2">Use the <b>th</b> tag to create the header which will display in bold</th>  
</tr>
```

```
<tr>  
<td width="50%">The <b>tr</b> tag creates the row</td>  
<td width="50%">The <b>td</b> tag creates individual cells for each row</td>  
</tr>
```

```
<tr>  
<td width="50%">You should always use the closing tags</td>  
<td width="50%">You can use a variety of HTML tags inside the cell</td>  
</tr>
```

```
<tr>  
<td width="50%">The three new version 4.0 table tags are <b>tbody, tfoot,</b> and  
<b>thead</b></td>
```

```
<td width="50%">The <b>colspan</b> attribute allows you span more than one cell</td>
</tr>
</table>
```

Output:

How To Code A Table

Use the caption tag to name the table and to provide useful information about the contents

Use the th tag to create the header which will display in bold	
The tr tag creates the row	The td tag creates individual cells for each row
You should always use the closing tags	You can use a variety of HTML tags inside the cell
The three new version 4.0 table tags are tbody , tfoot , and thead	Setting colspan allows you span more than one cell

TAG: center deprecated 4.0

<center> ... </center>

Available in versions: 3.2

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<center>** tag is officially deprecated effective with HTML 4.0. Instead, you are now to use style sheets. However, this tag remains in wide use on the Internet and is recognized by all major browsers.

The **<center>** tag is used to align the selected element in the center of the window with respect to the right and left borders. This is a very easy way to center text, tables, and images.

The centering effect starts with the opening **center** tag and continues until the closing tag. Therefore, the closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

none.

Code:

```
<center>  
<b>The Guru At</b>  
<br>  
  
</center>
```

Output:



TAG: cite

<cite> ... </cite>

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<cite>** tag takes the selected block of text (a citation) and displays it in italics. Unlike the **blockquote** tag, the citation is not preceded or followed by paragraph breaks. So the citation remain inline. You may still need to use other HTML tags to effect the textual display and appearance that you desire.

It is recommended that you use the **blockquote** tag for long citations and the **cite** tag for short citations.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

none.

Code:

```
As for what was lost, I simply refer you to the photography by Eliot Porter in  
<cite>The Place No One Knew: Glen Canyon on the Colorado River.</cite>
```

Output:

As for what was lost, I simply refer you to the photography by Eliot Porter in
The Place No One Knew: Glen Canyon on the Colorado River.

TAG: code

<code> ... </code>

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<code>** tag is designed to display computer code in a special font, usually in a mono spaced font style. However, the exact appearance will be browser dependent. You may still need to use various HTML tags, such as **br** and **p**, to create a presentation and appearance to suit your needs.

An HTML code example cannot use the HTML delimiters **<** and **>**, because they will be executed by the browser as real HTML code. Rather you must substitute a **<** for the **<** and a **>** for the **>**.

Usually, the display font appearance for the **code**, **kbd**, **samp**, and **tt** tags is the same. The **listing** tag has been deprecated.

The **pre** tag is more commonly used to display code.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None.

Code:

```
<code>  
if (document.getElementById) blnDOM = true;<br>  
else if (document.layers) blnNN4 = true;<br>  
else if (document.all) blnIE4 = true;<br>  
</code>
```

Output:

```
if (document.getElementById) blnDOM = true;  
else if (document.layers) blnNN4 = true;  
else if (document.all) blnIE4 = true;
```

TAG: col

<col>

Available in versions: 4.0

Browser compatibility: Explorer 4, 5 Netscape 6

The **<col>** tag is used to assign attribute values to the individual columns within the **colgroup** elements.

The **col** tag can only be used inside a **colgroup** element. However, this tag must be omitted if you are using the **span** attribute of the **colgroup** tag.

There is no closing tag.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

align

The **align** attribute is used to set the horizontal alignment of the cell contents. The possible values are **center**, **char**, **justify**, **left**, and **right**.

char

The **char** attribute permits you to select a character that will be used to align the contents of the cells in a column. This was designed to allow the decimal points in a column of numbers to be in vertical alignment. The default value is country dependent. In the U.S., it is the first occurrence of a period or a decimal point.

charoff

The **charoff** attribute is an integer used to specify if the value designated in the **char** attribute is to be the first, second, or whatever occurrence.

span

The **span** attribute is used to set the number of columns that are associated with each column group. If the columns are dissimilar, however, use the **col** tag to create each column, rather than this attribute.

valign

The **valign** attribute is used to set the vertical alignment of the cell contents. The possible values are **baseline**, **bottom**, **middle**, and **top**.

width

The **width** attribute is used to set the width of each of the spanned columns. The value can be a percentage of the total width or an integer number of pixels.

Code:

```
<table width="90%" border="10" cellspacing="5">  
<caption>  
<center><b>Column Group Table</b></center>  
<br>
```

This table uses the `colgroup` and `col` tags

```
</caption>
<colgroup>
<col width="20%">
<col width="30%">
<col width="15%">
</colgroup>
<colgroup>
<col width="15%">
<col width="20%">
</colgroup>
<tr>
<th>First Column Header</th>
<th>Second Column Header</th>
<th>Third Column Header</th>
<th>Fourth Column Header</th>
<th>Fifth Column Header</th>
</tr>
<tr>
<td>First Column First Row</td>
<td>Second Column First Row</td>
<td>Third Column First Row</td>
<td>Fourth Column First Row</td>
<td>Fifth Column First Row</td>
</tr>
<tr>
<td>First Column Second Row</td>
<td>Second Column Second Row</td>
<td>Third Column Second Row</td>
<td>Fourth Column Second Row</td>
<td>Fifth Column Second Row</td>
</tr>
</table>
```

Output:

Column Group Table

This table uses the `colgroup` and `col` tags

First Column Header	Second Column Header	Third Column Header	Fourth Column Header	Fifth Column Header
First Column First Row	Second Column First Row	Third Column First Row	Fourth Column First Row	Fifth Column First Row
First Column Second Row	Second Column Second Row	Third Column Second Row	Fourth Column Second Row	Fifth Column Second Row



TAG: colgroup

<colgroup> ... </colgroup>

Available in versions: 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<colgroup>** tag allows you to create a column-centric table as compared to the standard HTML table which is row-centric. The columns in the **table** are assigned to column groups using this tag. You can have one or more groups of columns.

If you are not using the **span** attribute of this tag, then you can use the **col** tag to assign attribute values to the individual columns within the **colgroup** elements.

The **colgroup** tag can only be used inside a **table** element. The closing tag is optional. However, it is recommended that it always be included.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#) [onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#) [title](#)

Attributes

align

The **align** attribute is used to set the horizontal alignment of the cell contents. The possible values are **center**, **char**, **justify**, **left**, and **right**.

char

The **char** attribute permits you to select a character that will be used to align the contents of the cells in a column. This was designed to allow the decimal points in a column of numbers to be in vertical alignment. The default value is country dependent. In the U.S., it is the first occurrence of a period or a decimal point.

charoff

The **charoff** attribute is an integer used to specify if the value designated in the **char** attribute is to be the first, second, or whatever occurrence.

span

The **span** attribute is used to set the number of columns that are associated with each column group. If the columns are dissimilar, however, use the **col** tag to create each column, rather than this attribute.

valign

The **valign** attribute is used to set the vertical alignment of the cell contents. The possible values are **baseline**, **bottom**, **middle**, and **top**.

width

The **width** attribute is used to set the width of each of the spanned columns. The value can be a percentage of the total width or an integer number of pixels.

Code:

```
<table width="90%" border="10" cellspacing="5">  
<caption>
```

```
<center><b>Column Group Table</b></center>
<br>
This table uses the <b>colgroup</b> and <b>col</b> tags
</caption>
<colgroup>
<col width="20%">
<col width="30%">
<col width="15%">
</colgroup>
<colgroup>
<col width="15%">
<col width="20%">
</colgroup>
<tr>
<th>First Column Header</th>
<th>Second Column Header</th>
<th>Third Column Header</th>
<th>Fourth Column Header</th>
<th>Fifth Column Header</th>
</tr>
<tr>
<td>First Column First Row</td>
<td>Second Column First Row</td>
<td>Third Column First Row</td>
<td>Fourth Column First Row</td>
<td>Fifth Column First Row</td>
</tr>
<tr>
<td>First Column Second Row</td>
<td>Second Column Second Row</td>
<td>Third Column Second Row</td>
<td>Fourth Column Second Row</td>
<td>Fifth Column Second Row</td>
</tr>
</table>
```

Output:

Column Group Table

This table uses the **colgroup** and **col** tags

First Column Header	Second Column Header	Third Column Header	Fourth Column Header	Fifth Column Header
First Column First Row	Second Column First Row	Third Column First Row	Fourth Column First Row	Fifth Column First Row
First Column Second Row	Second Column Second Row	Third Column Second Row	Fourth Column Second Row	Fifth Column Second Row

First
Column
Second
Row

Second
Column
Second
Row

Third
Column
Second
Row

Fourth
Column
Second
Row

Fifth
Column
Second
Row

TAG: dd

<dd>

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<dd>** tag is used to create a definition list which is a type of listing where a term or phrase is followed by a definition or explanation. The **dd** tag presents the definition. It must be used with the **dt** tag which presents the term and the **dl** tag which delimits the start and stop of the definition list.

The exact appearance of a definition list can be browser dependent. Most browsers place the term on the left margin and then the definition is placed on the next line and the entire text is indented. However, if the term is short (3 characters or less) the definition may start on the same line as the term. The list is usually separated from preceding and following text by paragraph breaks.

There is no closing tag.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None.

Code:

Definition Lists:

```
<dl>  
<dt>dl tag  
<dd>definition list start and stop delimiter.  
<dt>dt tag  
<dd>presents the term or phrase to be defined.  
<dt>dd tag  
<dd>presents the definition for the term or phrase.  
</dl>
```

Output:

Definition Lists:

```
dl tag  
  definition list start and stop delimiter.  
dt tag  
  presents the term or phrase to be defined.  
dd tag  
  presents the definition for the term or phrase.
```

TAG: del

** ... **

Available in versions: 4.0

Browser compatibility: Explorer 4, 5 Netscape 6

The **** tag is used with the **ins** tag to display editorial changes involving additions and deletions to the text of a document. The **del** tag marks the text to be deleted by striking a horizontal line through the characters. The **ins** tag designates the text to be added with an underline. This can also be accomplished using style sheets.

A **del** tag cannot occur between any opening and closing **ins** tags, and vice versa.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#) [onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#) [title](#)

Attributes

cite

The **cite** attribute is used to list one reference URL related to the editorial changes.

datetime

The **datetime** attribute is used to present the date and time that the editorial changes occurred. It must be of the format **YYYY-MM-DDThh:mm:ssTz** where the required **T** denotes the break between the date, the time, and the time zone.

Code:

```
The gentlemen of the first party agrees to pay <del>$1,047,300</del> <ins>$953,150</ins> to  
the gentlemen of the second party no later than <del>August 1, 2001</del> <ins>June 1,  
2002</ins>.
```

Output:

The gentlemen of the first party agrees to pay ~~\$1,047,300~~ \$953,150 to the gentlemen of the
second party no later than ~~August 1, 2001~~ June 1, 2002.

TAG: dfn

<dfn> ... </dfn>

Available in versions: 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 6

The **<dfn>** tag applies an oblique font style to the enclosed word or phrase. It is common practice in engineering, scientific, and technical articles to specially mark the first occurrence of a term.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None.

Code:

`<dfn>HTML</dfn>` is the acronym for the Hyper Text Markup Language.

Output:

HTML is the acronym for the Hyper Text Markup Language.

TAG: `dir` deprecated 4.0

`<dir> ... </dir>`

Available in versions: 2.0, 3.2

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The `<dir>` tag is officially deprecated effective with HTML 4.0. Instead, you are now to use the unordered list `ul` tag. However, this tag remains in minor use on the Internet and is still recognized by all major browsers.

The `dir` tag is used to create a directory list of computer files and programs. The concept was that the items in the list were expected to be file and program names which fit on a single line, rather than large blocks of text.

In fact, this tag is simply a specialized unordered list. The `dir` tag must be used with the `li` tag which places a bullet before each line of text just like it does for the `ul` tag. You can create a nested (indented) directory list by placing `dir` tags between another opening and closing pair of `dir` tags. In some browsers, the bullet becomes a circle for the nested list.

In most browsers, a paragraph break occurs before and after the list.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

`compact` deprecated 4.0

The `compact` attribute is deprecated effective with version 4.0. It was used to make a directory list more compact in appearance.

Code:

DevGuru Directory:

```
<dir>
<li> html_lists
<dir>
<li> html_li.html
<li> html_ol.html
<li> html_ul.html
</dir>
<li> html_events
<dir>
<li> html_onclick.html
<li> html_ondblclick.html
</dir>
</dir>
```

Output:

DevGuru Directory:

- `html_lists`
 - `html_li.html`
 - `html_ol.html`
 - `html_ul.html`
- `html_events`
 - `html_onclick.html`
 - `html_ondblclick.html`

TAG: div

<div> ... </div>

Available in versions: 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 6

The **<div>** tag is used to designate a block-line portion of an HTML document as a **div** element and to apply any of the core attributes of this tag against the contents of that element. The behavior of this tag is block-line, because this tag causes a line break (the equivalent of a **br** tag) to occur both before and after the **div** element. **div** tags, which can be nested, allow a significant degree of control and manipulation of each individual block section of your web page.

For example, you could use the **class** or **style** core attributes to apply the effects of Cascading Style Sheets, or you could use the **lang** attribute to display a foreign language text, or you could assign an **id** to each block of code for reference by a hyperlink.

In contrast, you should use the **span** tag when you want to apply attributes to an inline portion of a page. Note that the behavior of the **span** tag is not block-line, but rather inline, since the effects occur in the normal flow of the text and images (without the tag inserting any additional linefeeds or carriage returns).

With the advent of Cascading Style Sheets, both the **div** and **span** tags have gained importance and usefulness.

The ending tag is optional. However, it is recommended that it be included.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#) [onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#) [title](#)

Attributes

align deprecated 4.0

The **align** attribute is used to set the position of the contents of the **div** tag. The permitted values are **center**, **justify**, **left**, and **right**. This attribute is deprecated effective with version 4.0. However, most browsers continue to recognize this attribute. You are now to use style sheets.

Code:

```
<div style="text-align: left;" id="div_example1">
```

The **<div>** offers exciting possibilities for the developer.

```
</div>
```

```
<div style="text-align: center;" id="div_example2">
```

You should take time to learn about this tag.

```
</div>
```

```
<div style="text-align: right;" id="div_example3">
```

Also check out the **** tag.

```
</div>
```

Output:

The **div** offers exciting possibilities for the developer.

You should take time to learn about this tag.

Also check out the **span** tag.

TAG: dl

<dl> ... </dl>

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<dl>** tag delimits the start and stop of a definition list. This is a type of list where a term or phrase is followed by a definition or explanation.

This **dl** tag must be used with the **dt** tag which presents the term and the **dd** tag which presents the definition.

The exact appearance of a definition list can be browser dependent. Most browsers place the term on the left margin and then the definition is placed on the next line and the entire text is indented. However, if the term is short (three characters or less) the definition may start on the same line as the term. Also, the list is usually separated from any preceding and following text by paragraph breaks.

You can use the **ol** to create an ordered list and the **ul** tag to create an unordered list.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

compact deprecated 4.0

The **compact** attribute is used to make a definition list more compact in appearance. It is deprecated effective with version 4.0.

Code:

Definition Lists:

```
<dl>
<dt>dl tag
<dd>definition list start and stop delimiter.
<dt>dt tag
<dd>presents the term or phrase to be defined.
<dt>dd tag
<dd>presents the definition for the term or phrase.
</dl>
```

Output:

Definition Lists:

```
dl tag
  definition list start and stop delimiter.
dt tag
  presents the term or phrase to be defined.
dd tag
  presents the definition for the term or phrase.
```


TAG: DOCTYPE

<!DOCTYPE ... >

Available in versions: 2.0, 3.2, 4.0, 4.01

Browser compatibility: All

The **DOCTYPE** tag specifies the Definition Type Document (DTD) for HTML. This sets the rules for the grammar and syntax of the HTML language. While this tag is not mandatory, the W3C (World Wide Web Consortium), who set the standard for HTML 4.01, strongly recommends that you always include it in your code.

This tag can only be placed as the first line of the HTML code. It is the only tag that can precede the **HTML** tag.

- This tag is not closed.
- The exclamation mark ! is required.
- Obey the syntax and case.

Code:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 TRANSITIONAL//EM>
```

```
<html>
```

```
...
```

```
</html>
```

TAG: dt

<dt>

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<dt>** tag is used to create a definition list. This is a type of list where a term or phrase is followed by a definition or explanation. The **dt** tag presents the term to be defined. It must be used with the **dd** tag which presents the definition and the **dl** tag which delimits the start and stop of the definition list.

The exact appearance of a definition list can be browser dependent. Most browsers place the term on the left margin and then the definition is placed on the next line and the entire text is indented. However, if the term is short (3 characters or less) the definition may start on the same line as the term. Also, the list is usually separated from preceding and following text by paragraph breaks.

There is no closing tag.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None.

Code:

Definition Lists:

```
<dl>  
<dt>dl tag  
<dd>definition list start and stop delimiter.  
<dt>dt tag  
<dd>presents the term or phrase to be defined.  
<dt>dd tag  
<dd>presents the definition for the term or phrase.  
</dl>
```

Output:

Definition Lists:

```
dl tag  
  definition list start and stop delimiter.  
dt tag  
  presents the term or phrase to be defined.  
dd tag  
  presents the definition for the term or phrase.
```

TAG: em

` ... `

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The `` tag is used to provide emphasis when displaying the enclosed word or phrase. Most browsers do this by rendering the selected text in italics.

The `i` tag also renders in italics (or oblique), or you could use the style sheets **font-style** property to display in italics.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None.

Code:

Did you know that the `Guru` is over `800 years old`?

`
`

Did you know that the `Guru` is over `<i>800 years old</i>`?

Output:

Did you know that the **Guru** is over 800 years old?

Did you know that the **Guru** is over 800 years old?

TAG: fieldset

<fieldset> ... </fieldset>

Available in versions: 4.0

Browser compatibility: Explorer 4, 5 Netscape 6

The **<fieldset>** tag is used to group related **form** elements together with a caption and labels. The caption is created using the **legend** tag. One or more labels are created using the **label** tag. The browser is supposed to render the appearance of the caption and labels in a special manner.

The **fieldset**, **label**, and **legend** tags are poorly implemented by most browsers.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None.

Code:

```
<form>
<fieldset>
<legend>Please Enter Your Name</legend>
<label for="firstname">First Name</label><input type="text" id="firstname">
<br>
<label for="lastname">Last Name</label><input type="text" id="lastname">
</fieldset>
</form>
```

TAG: font deprecated 4.0

` ... `

Available in versions: 3.2

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **** tag and all associated attributes are officially deprecated effective with HTML 4.0. Instead, you are now to use style sheets. However, this tag remains in wide use on the Internet and is recognized by all major browsers.

The **font** tag is used to set the face (type), color, and size of the characters in a text.

The closing tag is mandatory.

Core Attributes

[class](#) [id](#) [lang](#) [style](#) [title](#)

Attributes

color

The **color** attribute is used to set the color of the text. It can be either a named color (such as red) or a six character hexadecimal RGB color code (such as #e26fa9).

face

The **face** attribute sets the font face (type) of the text, such as Arial or Courier. You can specify alternate faces by using a comma separated list. If a browser does not recognize the specified **face**, it will display the alternate face in the list or the default.

size

The **size** attribute is used to set the size of the text. The range is 1 to 7, and the default is 3. When you use + or - values to set the size using the **font** tag, the new size will be relative to the default value of 3, or to the value set using this attribute.

Code:

```
<font face="arial, helvetica" color="red" size="1">1</font>
<font face="arial, helvetica" color="red" size="2">2</font>
<font face="arial, helvetica" color="red" size="3">3</font>
<font face="arial, helvetica" color="red" size="4">4</font>
<font face="arial, helvetica" color="red" size="5">5</font>
<font face="arial, helvetica" color="red" size="6">6</font>
<font face="arial, helvetica" color="red" size="7">7</font>
<p>
<font face="arial, helvetica" color="red" size="5">c</font>
<font face="arial, helvetica" color="orange" size="5">c</font>
<font face="arial, helvetica" color="yellow" size="5">c</font>
<font face="arial, helvetica" color="green" size="5">c</font>
<font face="arial, helvetica" color="blue" size="5">c</font>
<font face="arial, helvetica" color="purple" size="5">c</font>
<font face="arial, helvetica" color="brown" size="5">c</font>
<p>
<font face="arial">Arial </font>
<font face="courier">Courier </font>
<font face="helvetica">helvetica </font>
```

Palatino

Symbol

Times >/font>

Output:

1 2 3 4 5 6 7

C C C C C C C C

Arial Courier helvetica Palatino Συμβολ Times

TAG: form

<form> ... </form>

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<form>** tag is used to delimit the start and stop of a form element and serves as a container for form controls (fields). Control is a technical term which refers to the various elements that can be used inside a **form** to gather information. The information gathered is referred to as the contents or parameters of the **form** and is a collection of name/value pairs.

The four tags that can be used to build a **form** are:

[button](#) [input](#) [select](#) [textarea](#)

The general concept is that the user fills out the appropriate sections of the **form** as a response to a request for information (such as a shipping address). The user then clicks the submit button. The contents of the **form** are then submitted for processing, usually to another page on the web site. However, you can also submit to the same page, or to a window or frame.

This ability to use a web site to display products and information, to gather a response via a **form** from a user, to store and manipulate the gathered information, and then to have the web site respond dynamically to the user input, is the heart of the e-commerce/e-business industry.

There are two required attributes for the **form** tag. The **action** attribute dictates where the form contents will be submitted. The **method** attribute specifies how to send the submission.

It is recommended that you should only place one **form** per page. It may prove best to divide a long **form** between two or more pages.

A **form** element should not contain other **form** tags. The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

accept

The **accept** attribute is a comma-separated list of file types that can be accepted by the server when using the **type="file"** content.

accept-charset

The **accept-charset** attribute is a comma-separated or space-separated list of the character types that the server must be able to support (process) when the contents of a **form** are submitted.

action

The **action** attribute sets the URL of the page that the contents (name/value pairs) of the **form** will be submitted to for processing.

enctype

The **enctype** attribute specifies the MIME type used to encode the contents of a **form**. The default is **application/x-www-form-urlencoded**. When using **type="file"** content, this attribute must be set to **multipart/form-data**.

method

The **method** attribute specifies which of two permitted HTTP methods will be used to pass the contents of a **form**. The more commonly used **post** method submits the contents of the **form** in a two step procedure. First the URL specified by the **action** attribute is contacted. Second, if the contact is successful, the contents are transmitted to the URL. The **get** method appends the contents to the end of the URL specified by the **action** attribute. (A question mark is used to delimit the end of the URL and the start of the contents.)

name

The **name** attribute is a string of characters that is used to label a form control (field) with a name. The name must be unique to that document and cannot be reused.

onreset

The **onreset** attribute is an event that allows the execution of JavaScript code when the user resets (clears) the **form**.

onsubmit

The **onsubmit** attribute is an event that allows the execution of JavaScript code when the user submits the **form**. For example, you could call a JavaScript function that performs client-side form verification to ensure that the **form** is filled out correctly (see code example).

target

The **target** attribute is used to submit the contents of the **form** to a specified window or frame.

Here is a sample **form** that includes client-side verification of fields.

```
<html>
<head>
<title>Form Examplem</title>
<script language="JavaScript">
// checksubmit() has to be a function, not a subroutine, because the
// onSubmit attribute in the form tag (in green text below) which calls
// this function requires a Boolean return value of true or false
// if the return is true, the form is automatically submitted without
// the need to call document.formname.submit()
// if the return is false, the form submission will be aborted and the client is sent
// back to the form to fill out the missing field
function checksubmit()
{
// check to see if name field is empty
// note that you use the name of the form and the field
if (document.formname.fullname.value == "")
{
// tell client that you need the name
alert("Please enter your full name")
// send the cursor to the fullname field
document.formname.fullname.focus()
// field is empty, so return false to abort the form submission
// the client is returned to the form
return false
}
```

```
}
// check to see if email address field is empty
if (document.formname.emailaddress.value == "")
{
    alert("Please enter your email address")
    document.formname.emailaddress.focus()
    return false
}
// if both fields are filled out, return true
// this triggers the form submission
return true
}
</script>
</head>
<body onLoad="document.formname.fullname.focus()">
If you wish to receive information about upgrades to dgCharge,
please fill out this form.
<br>
<form method="post" name="formname" action="/technologies/html/quickref/html_form.html"
    onSubmit="return checksubmit()">
Full Name (required) <input type="text" name="fullname" size="30">
<br>
EmailAddress (required) <input type="text" name="emailaddress" size="30">
<br>
Phone Number (optional) <input type="text" name="phonenumber" size="15">
<br>
<input type="reset" value="Clear">
<input type="submit" name="submitbtn" value="Submit">
</form>
</body>
</html>
```

TAG: frame

<frame ... >

Available in versions: 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<frame>** tag is used to create one frame, which is simply a window within another window. As an example, if you want three frames (windows) to appear on an HTML page, you must have three **frame** tags and the **frameset** element must set the spacing for three frames.

The **frame** tags can only occur between an opening and closing **frameset** element.

A **frame** element displays content, including forms, images, multimedia, other frames, tables, etc. It is very important to understand that the only way to assign content to a **frame** is to assign a URL using the **src** attribute. Whatever content is displayable at the referenced URL will in turn be displayed inside the **frame**.

The frames can have either a column (vertical) or a row (horizontal) configuration, but not both. The configuration must be set using either the **cols** or the **rows** attribute of the **frameset** tag, but not both. The order of the **frame** tags sets the order of the frames. If they are in a **rows** configuration, the first **frame** element is the top window, the second **frame** element is second from the top, etc. For the **cols** configuration, the order is left to right.

Note that you can nest **frameset** tags inside of **frameset** tags. Therefore, an HTML document can contain numerous frames. This opens the possibility of creating complex and imaginative page displays (however, always remember to keep the page user friendly).

Core Attributes

[class](#) [id](#) [style](#) [title](#)

Attributes

frameborder

The **frameborder** attribute is used to add or remove a border around a frame. The value of **yes** or **1** allows a border. The value of **no** or **0** forbids a border.

longdesc

The **longdesc** attribute is the URL address of an HTML document that contains a long textual description of the **frame**. This is designed to provide more information than a **title** element would display.

marginheight

The **marginheight** attribute is used to set the amount of white space in pixels that will appear along the top and bottom margins between the frame border and the contents. If you provide an unrealistic value, the browser will simply ignore this attribute.

marginwidth

The **marginwidth** attribute is used to set the amount of white space in pixels that will appear along the left and right margins between the frame border and the contents. If you provide an unrealistic value, the browser will simply ignore this attribute.

name

The **name** attribute is a string of characters that is used to label a frame with a name. The

name must be unique to that document and cannot be reused.

noresize

The **noresize** attribute prevents the user from changing the **frame** size.

scrolling

The **scrolling** attribute is used to determine if horizontal and vertical scrolling bars will appear in the **frame**. If the content is larger than the frame, this permits scrolling up and down or left and right, as needed, to see the entire contents. There are three permitted values: **yes** which always displays scroll bars; **no** which never displays scroll bars; and **auto** which displays scroll bars only when needed. If a value is not provided, the default is **auto**.

src

The **src** attribute provides the URL address of any valid HTML, ASP, or other displayable page. The contents of the referenced URL will be displayed inside the **frame**.

This example creates five frames, each with a different background color. Note that there is no **body** tag.

Code:

```
<html>
<head>
<title>frames example</title>
</head>
<frameset cols="50%, 50%">
  <noframes>
    To be viewed properly, this page requires frames.
  </noframes>
  <frameset rows="33%, 33%, *">
    <frame src="html_frames_red.html">
    <frame src="html_frames_orange.html">
    <frame src="html_frames_yellow.html">
  </frameset>
  <frameset rows="50%, 50%">
    <frame src="html_frames_green.html">
    <frame src="html_frames_blue.html">
  </frameset>
</frameset>
</html>
```

TAG: frameset

`<frameset> ... </frameset>`

Available in versions: 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<frameset>** tag serves as a container element to hold a collection of frames. This tag determines the number of frames, whether the frames are to be of a column or of a row configuration, and sets the spacing for either case. In fact, you must specify either a column or row configuration using either the **rows** or **cols** attribute (but not both).

When you are using the **frameset** tag, you absolutely must not use a **body** tag. The **frameset** tag is used in place of the **body** tag. However, you can use the **iframe** tag to create inline frames within a **body** element.

The only tags that are valid inside the **frameset** tag are the **frame**, **frameset**, **head**, and **noframe**. Note that you can nest **frameset** tags inside of **frameset** tags. Therefore, an HTML document can contain numerous frames. This opens the possibility of creating complex and imaginative page displays (however, always remember to keep the page user friendly).

The closing tag is mandatory.

Core Attributes

[class](#) [id](#) [style](#) [title](#)

Attributes

cols

The **cols** attribute is used to set the number of column frames in a **frameset** element. This is done indirectly since you never provide the actual number of columns. Rather, this attribute is a quote-enclosed, comma-separated list of values that specify the width assigned to each **frame**. The number of values in the list determines the number of frames. For example, if you have three values, then you are expected to have three frames. These values can be:

- a percentage of the window width
- an integer number of pixels
- either of the above methods plus the asterisk *

The asterisk is used to proportionally assign leftover spacing. A 3* value has three times the proportion than a single *.

Possible examples are:

Code:

<code><frameset cols="40%, 60%" ></code>	for two column frames
<code><frameset cols="20%, 50%, 30%" ></code>	for three column frames
<code><frameset cols="20%, 20%, 20%, 20%, 20%" ></code>	for five column frames
<code><frameset cols="100, 100" ></code>	for two column frames
<code><frameset cols="50, 75, 150" ></code>	for three column frames
<code><frameset cols="150, * " ></code>	for two column frames
<code><frameset cols="*, 50%, *, * " ></code>	for four column frames
<code><frameset cols="3*, *" ></code>	for two column frames
<code><frameset cols="20%, *, 2*" ></code>	for three column frames

If you use the **cols** attribute, you cannot use the **rows** attribute.

onload

The **onload** attribute is an event that allows a JavaScript code to execute when the **frame** is first loaded by the browser.

onunload

The **onunload** attribute is an event that allows a JavaScript code to execute when the **frame** is unloaded by the browser.

rows

The **rows** attribute is used to set the number of row frames in a **frameset** element. This is done indirectly since you never provide the actual number of rows. Rather, this attribute is a quote-enclosed, comma-separated list of values that specify the height assigned to each **frame**. The number of values in the list determines the number of frames. For example, if you have three values, then you are expected to have three frames. These values can be:

- a percentage of the window height
- an integer number of pixels
- either an integer of the above methods plus the asterisk *

The asterisk is used to proportionally assign leftover spacing. A 3* value has three times the proportion than a single *.

Possible examples are:

Code:

<code><frameset rows="40%, 60%" ></code>	for two row frames
<code><frameset rows="20%, 50%, 30%" ></code>	for three row frames
<code><frameset rows="20%, 20%, 20%, 20%, 20%" ></code>	for five row frames
<code><frameset rows="100, 100" ></code>	for two row frames
<code><frameset rows="50, 75, 150" ></code>	for three row frames
<code><frameset rows="150, * "></code>	for two row frames
<code><frameset rows="*, 50%, *, * "></code>	for four row frames
<code><frameset rows="3*, *" ></code>	for two row frames
<code><frameset rows="20%, *, 2*" ></code>	for three row frames

If you use the **rows** attribute, you cannot use the **cols** attribute.

This example creates five frames, each with a different background color.

Code:

```
<html>
<head>
<title>frames example</title>
</head>
<frameset cols="50%, 50%">
  <noframes>
    To be viewed properly, this page requires frames.
  </noframes>
  <frameset rows="33%, 33%, *">
    <frame src="html_frames_red.html">
    <frame src="html_frames_orange.html">
    <frame src="html_frames_yellow.html">
```

```
</frameset>
<frameset rows="50%, 50%">
  <frame src="html_frames_green.html">
  <frame src="html_frames_blue.html">
</frameset>
</frameset>
</html>
```

TAG: h1, h2, h3, h4, h5, h6

<h#> ... </h#> where # = 1, 2, 3, 4, 5, or 6

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<h#>** tags are used to create text headers for a document. They can display the header text in one of six different sizes. **h1** is the largest size and, as the integer number increases to **h6**, the text size becomes progressively smaller. The exact display appearance of the header text is browser dependent.

This tag also automatically inserts a carriage return and line feed after the closing tag (i.e., a paragraph break).

It is common practice to use this tag in place of the **font** tag or style sheets to set a font size for the text of an entire document. However, the true purpose of this tag is only to create headers and titles. It is recommended that you use style sheets to set the font properties for a text.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

align deprecated 4.0

The **align** attribute is used to set the position of the header with regard to the margins. The permitted values are **center**, **left**, and **right**. This attribute is deprecated effective with version 4.0. You are now to use style sheets.

Code:

```
<h1>DevGuru in h1</h1>  
<h2>DevGuru in h2</h2>  
<h3>DevGuru in h3</h3>  
<h4>DevGuru in h4</h4>  
<h5>DevGuru in h5</h5>  
<h6>DevGuru in h6</h6>
```

Output:

DevGuru in h1

DevGuru in h2

DevGuru in h3

DevGuru in h4

DevGuru in h5

DevGuru in h6

TAG: hr

<hr>

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<hr>** tag is used to render a horizontal rule (line). This is a very useful and commonly used tag.

The flow of the text and images is halted on the current line, the rule is rendered on the next line, and then the flow of text and images resumes on the following line. The exact amount of vertical space between the rule and any elements above and below it is browser dependent.

The four attributes allow you to adjust the height, position, shading, and width of the rule. Note that these four attributes are deprecated effective with version 4.0. However, all browsers continue to recognize these attributes. You are now to use style sheets to effect the appearance of the rule.

There is no closing tag.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#) [onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#) [title](#)

Attributes

align deprecated 4.0

The **align** attribute is used to position the rule to the **center**, **left**, or **right**. This attribute is deprecated effective with version 4.0. However, all browsers continue to recognize this attribute.

noshade deprecated 4.0

The **noshade** attribute is a Boolean value. It is not assigned a value. If the attribute is not present, the rule is shaded. If the attribute is present, the rule is not shaded. This attribute is deprecated effective with version 4.0. However, all browsers continue to recognize this attribute.

size deprecated 4.0

The **size** attribute is used to set the height of the rule in pixels. The default is 3 pixels. This attribute is deprecated effective with version 4.0. However, all browsers continue to recognize this attribute.

width deprecated 4.0

The **width** attribute is used to set the horizontal width of the rule. It can be assigned either a value in pixels or a percentage of the width of the window containing the element. This attribute is deprecated effective with version 4.0. However, all browsers continue to recognize this attribute.

Code:

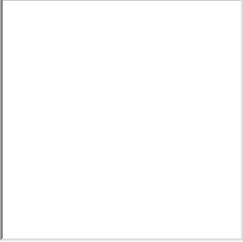
```
<hr>
```

```
<hr size="5">
```

```
<hr size="10">
```

```
<hr size="50" width="50%" align="center" noshade>  
<hr size="100" width="100" align="left">  
<hr style="color:red; height:15px; width:350px;">
```

Output:



TAG: html

<html> ... </html>

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<html>** tag informs the browser that this is an HTML encoded program. The **html** opening and closing tags delimit the start and stop of the HTML document.

The ending tag is optional. However, it is recommended that it be included.

Core Attributes

[dir](#) [lang](#)

Attributes

version deprecated 4.0

The **version** attribute defines the HTML version number. This attribute is deprecated effective with version 4.0. However, most browsers continue to recognize this attribute.

This example demonstrates the basic structure of an HTML program.

Code:

```
<html>
<head>
<title>title</title>
</head>
<body>
body content goes here...
</body>
</html>
```

Output:

body content goes here...

In general, it is not necessary (or recommended) that you designate the HTML version type. However, if you must, the following code demonstrates how:

Code:

```
<html version="-//W3C//DTD HTML 4.0//EN">
```

Rather than using the deprecated **version** attribute, some programmers place the following SGML (Standard Generalized Markup Language) command code as the first line of their HTML document.

Code:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
```

TAG: i

`<i> ... </i>`

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The `<i>` tag causes the designated text to be displayed as an italic or oblique font style. If italic or oblique is not available for a particular font, the browser may impose some other physical change to the text, such as underline.

You can use the `em` tag to render in italics, the `dfn` tag to render in oblique, or use the style sheets `font-style` property to display in italics.

The closing tag is mandatory.

Core attributes:

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None

Code:

Did you know that the `Guru` is over `<i>800 years old</i>`?

`
`

Did you know that the `Guru` is over `800 years old`?

Output:

Did you know that the **Guru** is over 800 years old?

Did you know that the **Guru** is over 800 years old?

TAG: iframe

`<iframe> ... </iframe>`

Available in versions: 4.0

Browser compatibility: Explorer 4, 5 Netscape 6

The **<iframe>** tag is used to insert an inline frame into the body of an HTML document. This is a very convenient method for inserting an HTML page inside another HTML page. Unlike the **frameset**, **frame**, and **noframes** tags, the **iframe** tag can only be used inside a **body** element. You must not place the **iframe** tag inside a **frameset** element.

An **iframe** element displays content, including forms, images, multimedia, other frames, tables, etc. It is very important to understand that the only way to assign content to an **iframe** is to assign a URL using the **src** attribute. Whatever content is displayable at the referenced URL will in turn be displayed inside the **iframe**.

You can place a default textual message between the opening and closing **iframes** tags. This will only display if the browser does not recognize inline frames.

The closing tag is mandatory.

Core Attributes

[class](#) [id](#) [style](#) [title](#)

Attributes

align

The **align** attribute is used to place the frame inline with adjacent text. The values **bottom**, **middle**, and **top** align the frame with the bottom, middle, and top of the text. The values **right** and **left** place the text on the right or left boundary and allow any text to flow around the frame.

frameborder

The **frameborder** attribute is used to add or remove a border around the frame. The values of **yes** or **1** allow a border. The values of **no** or **0** forbid a border.

height

The **height** attribute sets the vertical extent of the **iframe** in either pixels or as a percentage of the overall screen height.

longdesc

The **longdesc** attribute is the URL address of an HTML document that contains a long textual description of the **frame**. This is designed to provide more information than a **title** element would display.

marginheight

The **marginheight** attribute is used to set the amount of white space in pixels that will appear along the top and bottom margins between the frame border and the contents. If you provide an unrealistic value, the browser will simply ignore this attribute.

marginwidth

The **marginwidth** attribute is used to set the amount of white space in pixels that will appear along the left and right margins between the frame border and the contents. If you provide an unrealistic value, the browser will simply ignore this attribute.

name

The **name** attribute is a string of characters that is used to label a frame with a name. The name must be unique to that document and cannot be reused.

scrolling

The **scrolling** attribute is used to determine if horizontal and vertical scrolling bars will appear in the **iframe**. If the content is larger than the frame, this permits scrolling up and down or left and right, as needed, to see the entire contents. There are three permitted values: **yes** which always displays scroll bars; **no** which never displays scroll bars; and **auto** which displays scroll bars only when needed. If a value is not provided, the default is **auto**.

src

The **src** attribute provides the URL address of any valid HTML, ASP, or other displayable page. The contents of the referenced URL will be displayed inside the **frame**.

width

The **width** attribute sets the horizontal extent of the **iframe** in either pixels or as a percentage of the overall screen width.

Code:

```
<iframe width="50%" height="77" src="html_frames_yellow_two.html" frameborder="yes"
scrolling="yes">
Sorry, you need inline frames to fully see this page.
</iframe>
```

Output:

Sorry, you need inline frames to fully see this page.

TAG: img

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **** tag is used to insert a graphic or photographic image directly into the flow of text and other images. No line breaks or carriage returns are automatically inserted before or after the image. You can use attributes and style sheets to effect an appearance and presentation of images and text that suits your needs.

This tag can be placed in an **a** element to provide a clickable image for a hyperlink. A border will be automatically drawn around all four sides of an image used as a link. You can remove the border using style sheets or the **border** attribute.

The exact display of an image is very browser dependent.

There is no closing tag.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

align deprecated 4.0

The **align** attribute is used to control the position of the image with respect to the text and other elements. The permitted values are **absbottom**, **absmiddle**, **baseline**, **bottom**, **left**, **middle**, **right**, **texttop**, and **top**. This attribute is deprecated effective with version 4.0. However, most browsers continue to recognize this attribute.

alt

The **alt** attribute provides a text message that will be displayed (in place of the image) on browsers that cannot display a graphic image or picture.

border

The **border** attribute places a border around all four sides of the image. The thickness of the border is set in pixels. If you are using an image as a hyperlink and desire no border, set this attribute to zero.

height

The **height** attribute is used to set the vertical extent of the image in pixels.

hspace

The **hspace** attribute is used to add additional blank space in pixels to each side of the image.

ismap

The **ismap** attribute is a Boolean value that, when present, signifies that the image is being used as a mouse-clickable server-side image map.

longdesc

The **longdesc** attribute is the URL of a source that contains a long description of the image.

This is useful for server-side image maps and nonvisual browsers.

name

The **name** attribute sets the name of the image. (This is required when using JavaScript.)

src

The mandatory **src** attribute is the URL of the location of the image file. The URL must include the file name and can include a relative or absolute path.

usemap

This **usemap** attribute specifies the name of an image map that you wish to associate with the element. This indicates that the image is being used as a mouse-clickable client-side image map.

vspace

The **vspace** attribute is used to add additional blank space in pixels both above and below the image.

width

The **width** attribute is used to set the horizontal extent of the image in pixels.

Code:

```

```

Output:



TAG: ins

`<ins> ... </ins>`

Available in versions: 4.0

Browser compatibility: Explorer 4, 5 Netscape 6

The **<ins>** tag is used with the **del** tag to display editorial changes involving additions and deletions to the text of a document. The **del** tag marks the text to be deleted by striking a horizontal line through the characters. The **ins** tag designates the text to be added with an underline. This can also be accomplished using style sheets.

An **ins** tag cannot occur between any opening and closing **del** tags, and vice versa.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

cite

The **cite** attribute is used to list one reference URL related to the editorial changes.

datetime

The **datetime** attribute is used to present the date and time that the editorial changes occurred. It must be of the format **YYYY-MM-DDThh:mm:ssTZA** where the required **T** denotes the break between the date, the time, and the time zone.

Code:

```
The gentlemen of the first party agrees to pay <del>$1,047,300</del> <ins>$953,150</ins> to  
the gentlemen of the second party no later than <del>August 1, 2001</del> <ins>June 1,  
2002</ins>.
```

Output:

The gentlemen of the first party agrees to pay ~~\$1,047,300~~ \$953,150 to the gentlemen of the
second party no later than ~~August 1, 2001~~ June 1, 2002.

TAG: kbd

<kbd> ... </kbd>

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<kbd>** tag is used to signify that the indicated text is to be typed by the user on the keyboard. For example, this can be used in training manuals.

The exact appearance will be browser dependent. However, most browsers render the indicated text in a mono spaced font.

Usually, the display font appearance for the **code**, **kbd**, **samp**, and **tt** tags is the same. The **listing** tag has been deprecated.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None.

Code:

Please type `<kbd>Request.Write strVar</kbd>` in the third window.

Output:

Please type `Request.Write strVar` in the third window.

TAG: label

<label> ... </label>

Available in versions: 4.0

Browser compatibility: Explorer 4, 5 Netscape 6

The **<label>** tag is used to associate text labels with a specific element (control) inside a **form** (such as an **input** element). This is done by having the values of the **for** attribute of the **label** tag and the **id** attribute of the form element be the same. Note that you can attach more than one label to the same element. The browser is supposed to render the labels in a special manner to accentuate their appearance.

This tag is one of three tags implemented in 4.0 that help set the appearance of a **form**. The other two are **fieldset** and **legend**.

However, these three tags are poorly implemented by most browsers.

A **label** element should not contain other **label** tags. The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#) [onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#) [title](#)

Attributes

accesskey

The **accesskey** attribute allows you to designate a key on the keyboard that when pressed, along with the alt or meta key, will bring focus to the form element associated with the **label**. This attribute is poorly supported.

for

The **for** attribute is used to relate the **label** tag to a specific form control. It must have the exact same value as the **id** attribute of the form control. However, if the form control is placed inline between opening and closing **label** tags, the **for** and **id** attributes are not needed since the form control and the **label** are implicitly joined.

onblur

The **onblur** attribute is an event that allows a JavaScript code to execute when an element loses focus (for example, the mouse was clicked onto another element or a tab navigation directed the cursor elsewhere).

onfocus

The **onfocus** attribute is an event that allows a JavaScript code to execute when an element comes into focus (for example, the mouse was clicked onto the element or a tab navigation brought the cursor to the element).

Code:

```
<form>
<fieldset>
<legend>Please Enter Your Name</legend>
<label for="firstname">First Name</label><input type="text" id="firstname">
<br>
```

```
<label for="lastname">Last Name</label><input type="text" id="lastname">
</fieldset>
</form>
```

TAG: legend

<legend> ... </legend>

Available in versions: 4.0

Browser compatibility: Explorer 4, 5 Netscape 6

The **<legend>** tag is used to create a caption for the **form** elements associated with a **fieldset** group. Therefore, this tag can only be used inside the **fieldset** element. The browser is supposed to render the **legend** caption in a special manner to accentuate the appearance.

This tag is one of three tags implemented in 4.0 that help set the appearance of a **form**. The other two are **fieldset** and **label**.

However, these three tags are poorly implemented by most browsers.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

accesskey

The **accesskey** attribute allows you to designate a key on the keyboard that when pressed, along with the alt or meta key, will bring focus to the form element associated with the **legend**.

align **deprecated 4.0**

The **align** attribute is used to place the **legend** in the **fieldset**. The permitted values are **bottom**, **left**, **right**, and **top**. This attribute is deprecated effective with version 4.0. You are now to use style sheets.

Code:

```
<form>
<fieldset>
<legend>Please Enter Your Name</legend>
<label for="firstname">First Name</label><input type="text" id="firstname">
<br>
<label for="lastname">Last Name</label><input type="text" id="lastname">
</fieldset>
</form>
```

TAG: li

** ... **

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **** tag is used to list an item in a list. This tag is required for both the ordered list **ol** tag and the unordered list **ul** tag. (It is also required for two deprecated tags, the **dir** directory list and the **menu** list.)

For ordered and unordered lists, there are essentially no restrictions as to the text that may be placed in a list, such as length. (However, directory and menu lists, by definition, are listings of short items.)

In an unordered list, each item is preceded by a bullet, such as •. In an ordered list, each item is labeled with a number, letter, or Roman numeral that increments with each following item.

The ending tag is optional. However, it is recommended that it be included.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

type deprecated 4.0

The **type** attribute determines the characters used to order the list. For an ordered list, the permitted values are: **1** for Arabic numbers; **I** or **i** for upper or lower case Roman numerals; and **A** or **a** for upper or lower case alphabetic characters. For an unordered list, the permitted values are a **circle**, **disc**, and **square**. It is deprecated effective with version 4.0.

value deprecated 4.0

The **value** attribute is used to set the order value that precedes a specific item in an ordered list. It is deprecated effective with version 4.0.

This is an unordered list.

Code:

Three Major Types Of Lists:

```
<ul>
<li> Definition List
<ul type="square">
<li> dd tag
<li> dl tag
<li> dt tag
</ul>
<li> Ordered List
<ul type="circle">
<li> ol tag
<li> li tag
</ul>
<li> Unordered List
```

```
<ul type="disc">
<li> ul tag
<li> li tag
</ul>
</ul>
```

Output:

Three Major Types Of Lists:

- Definition List
 - dd tag
 - dl tag
 - dt tag
- Ordered List
 - ol tag
 - li tag
- Unordered List
 - ul tag
 - li tag

This is an ordered list.

Code:

Three Major Types Of Lists:

```
<ol>
<li> Definition List
<ol type="A" start="5">
<li> dd tag
<li> dl tag
<li> dt tag
</ol>
<li> Ordered List
<ol type="i">
<li> ol tag
<li> li tag
</ol>
<li> Unordered List
<ol type="1">
<li> ul tag
<li> li tag
</ol>
</ol>
```

Output:

Three Major Types Of Lists:

1. Definition List
 - E. dd tag
 - F. dl tag
 - G. dt tag
2. Ordered List
 - i. ol tag
 - ii. li tag
3. Unordered List

1. ul tag
2. li tag

TAG: map

<map> ... </map>

Available in versions: 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<map>** tag is used to create a client-side image map which is divided into two or more mouse-sensitive regions. For example, when you click onto a specific region of the image map, a hyperlink could send you to a target page or a pop-up window could display some information.

The **map** element essentially serves as a container for the code that creates and implements the image map. For example, this code could be a series of **area** tags that are used to set image coordinates and to assign a hyperlink to each region of the image map. Also, since the image map is mouse-sensitive, you could also use any of the onmouse core attributes to capture a mouse event and execute a JavaScript code.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

name

The **name** attribute is a string of characters that is used to label the **map** with a name. The name must be unique to that document and cannot be reused. The value for this attribute is case-sensitive.

The **area** and **map** tags are used on the **DevGuru** site to create image mapped links for the indexes to the various Quick References. For example, this is the code used for the ASP Quick Reference.

Code:

```
  
<br>  
<map name="methodproperty">  
<area shape="rect" coords="192,0,248,26" href="/technologies/asp/quickref/properties.html">  
<area shape="rect" coords="133,0,191,26" href="/technologies/asp/quickref/collections.html">  
<area shape="rect" coords="90,0,134,22" href="/technologies/asp/quickref/events.html">  
<area shape="rect" coords="43,0,89,22" href="/technologies/asp/quickref/methods.html">  
</map>
```

TAG: menu deprecated 4.0

`<menu> ... </menu>`

Available in versions: 2.0, 3.2

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The `<menu>` tag is officially deprecated effective with HTML 4.0. Instead, you are now to use the unordered list `ul` tag. However, this tag remains in minor use on the Internet and is still recognized by all major browsers.

The `menu` tag is used to create an unordered list. The concept was that the items in the list were expected to be single words or very short phrases which fit on a single line, rather than large blocks of text.

In fact, this tag is simply a specialized unordered list. The `menu` tag must be used with the `li` tag which places a bullet (a circle, disc, or square) before each line of text just like it does for the `ul` tag. You can create a nested (indented) directory list by placing `menu` tags between another opening and closing pair of `menu` tags. In some browsers, the bullet becomes a circle for the nested list.

In most browsers, a paragraph break occurs before and after the list.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None.

Code:

HTML categories:

```
<menu>
```

```
<li> lists:
```

```
<menu>
```

```
<li> li
```

```
<li> ol
```

```
<li> ul
```

```
</menu>
```

```
<li> events:
```

```
<menu>
```

```
<li> onclick
```

```
<li> ondblclick
```

```
</menu>
```

```
</dir>
```

Output:

HTML categories:

- lists:

- li
- ol
- ul
- events:
 - onclick
 - ondblclick

TAG: noframes

<noframes> ... </noframes>

Available in versions: 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<noframes>** tag is used to display an alternative message on browsers that do not recognize frames. Usually, this message advises the user that frames are needed to see the contents of this HTML document. If the browser does recognize frames, then the contents of the **noframes** element are not displayed.

The **noframes** tag must be placed immediately following the first use of the **frameset** tag. The ending tag is optional. However, it is recommended that it be included.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None.

This example creates five frames, each with a different background color.

Code:

```
<html>
<head>
<title>frames example</title>
</head>
<frameset cols="50%, 50%">
  <noframes>
    To be viewed properly, this page requires frames.
  </noframes>
  <frameset rows="33%, 33%, *">
    <frame src="html_frames_red.html">
    <frame src="html_frames_orange.html">
    <frame src="html_frames_yellow.html">
  </frameset>
  <frameset rows="50%, 50%">
    <frame src="html_frames_green.html">
    <frame src="html_frames_blue.html">
  </frameset>
</frameset>
</html>
```

TAG: noscript

<noscript> ... </noscript>

Available in versions: 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<noscript>** tag is used to display alternative content for browsers that do not recognize the **script** tag, or for those occasions where the user has purposely disabled **script**.

This tag is inconsistently implemented.

The closing tag is required.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None.

Code:

```
<noscript>
```

```
Script is disabled.
```

```
</noscript>
```

Output:

TAG: object

<object> ... </object>

Available in versions: 4.0

Browser compatibility: Explorer 4, 5

The **<object>** tag is used to insert an object (such as ActiveX components, applets, image maps, media players, and plug-ins) into an HTML document and to provide all of the necessary information to implement and run the object. For example, you can give the location and type of the executable code of the object, and the location and type of any associated data files. You have the option to declare and instantiate the object at the same time, or declare the object and instantiate it, one or more times, elsewhere in the program.

This tag can be placed in the **head** and the **body** element. If it is placed in the **head**, the object cannot be used to render (display) content onto the page. (No tag in the **head** is permitted to render content.)

You can insert text between the opening and closing tags that will be displayed as the default message if the object cannot be displayed by the browser.

The **param** tag is used to pass any parameters (run-time values) required to implement the object. This tag must appear immediately after the opening **object** tag, but before any other content.

Note that the **applet** tag was deprecated effective with version 4.0. The idea was that you were to use the **object** tag (and the **classid** attribute) to insert a Java applet into a web page. However, the **applet** is used extensively by large numbers of users throughout the Internet. The howls of protest guaranteed that the **applet** will be with us for a long time.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

align deprecated 4.0

The **align** attribute is used to position the object with respect to any surrounding text, images, or other objects. This attribute is deprecated effective with version 4.0. However, most browsers continue to recognize this attribute. You are now to use style sheets.

archive

The **archive** attribute is a comma-separated list of URLs of object resources, including resources specified by the **classid** and **data** attributes, that must be preloaded into the local user cache before the object can be displayed. If the URLs are relative, you must provide the base URL using the **codebase** attribute.

classid

The **classid** attribute is used to provide the URL location of the object. If the URL is relative, you must provide a base URL using the **codebase** attribute. When using this attribute, you may also wish to specify the program code type of the object by using the **codetype** attribute.

codebase

The **codebase** attribute is used to specify the base URL of the directory where the object is stored. This allows the use of relative URLs in other attributes. If the directory is different from where the HTML code is stored, you must use this attribute.

codetype

The **codetype** attribute is used to specify the program code type (MIME type) of the object. The default is to allow the browser to determine the program code type. The data files code type is specified by using the **type** attribute.

data

The **data** attribute is used to specify the URL location of the data files associated with the object (such as images). If the URL is relative, you must provide a base URL using the **codebase** attribute.

declare

The **declare** attribute is a Boolean value that, if present, signifies that the object is a declaration only and prevents the browser from downloading and executing the object. (When you declare an object, you should also assign a value to the **id** attribute. This **id** can be referenced at a later time when you instantiate the object.)

height

The **height** attribute is used to set the vertical extent of the object in pixels.

hspace

The **hspace** attribute is used to add additional blank space in pixels to each side of the object.

name

The **name** attribute is used to assign a unique string of characters as the name of the object (which allows the object to be referenced).

standby

The **standby** attribute is used to display a message while the object is loading. This is a wise precaution for slow-loading objects and should be considered a courtesy to the user.

tabindex

The **tabindex** attribute specifies an integer that defines the rank in the tabbing order for the specified element when you use the keyboard to navigate (tab through) a page. This attribute is poorly supported.

type

The **clear** attribute is used to specify the data files code type (MIME type) of the object. The default is to allow the browser to determine the data file code type. The program code type is specified by using the **codetype** attribute.

usemap

The **usemap** attribute is used to indicate that the object is a client-side image map.

vspace

The **vspace** attribute is used to add additional blank space in pixels both above and below the object.

width

The **width** attribute is used to set the horizontal extent of the object in pixels.

This example displays the Microsoft MSCAL calendar.

Code:

```
<object classid="clsid:8E27C92B-1264-101C-8A2F-040224009C02" ID="calFormDate"  
width="372" height="200">  
Sorry, your browser does not display the Microsoft MSCAL Calendar  
</object>
```

TAG: ol

** ... **

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **** tag is used to delimit the start and stop of an ordered list. An ordered list is a collection of items (typically related somehow) that need to be listed in particular order. For example, an ordered list could be an index, table of contents, or a set of instructions. The default is to list each item in numeric order (starting with the number 1). However, you can also specify Roman numerals or alphabetic characters.

Most browsers separate the list from any preceding and following text by paragraph breaks. You can nest ordered lists and the nested lists will also be in ordered value.

You must use the **li** tag to display an item in the list.

You can use the **ul** tag to create an unordered list and the **dl** tag to create a definition list. (The **dir** and **menu** list tags are deprecated.)

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

compact deprecated 4.0

The **compact** attribute is used to make an ordered list more compact in appearance. This attribute is poorly implemented. It is deprecated effective with version 4.0.

start deprecated 4.0

The **start** attribute is used to set the ordering value that the list will start with (the default is 1). You can also start with any letter in the alphabet or any Roman numeral. It is deprecated effective with version 4.0.

type deprecated 4.0

The **type** attribute determines the characters used to order the list. The permitted values are: **1** for Arabic numbers; **I** or **i** for upper or lower case Roman numerals; and **A** or **a** for upper or lower case alphabetic characters. It is deprecated effective with version 4.0.

Code:

Three Major Types Of Lists:

```
<ol>
<li> Definition List
<ol type="A" start="5">
<li> dd tag
<li> dl tag
<li> dt tag
</ol>
<li> Ordered List
```

```
<ol type="i">
<li> ol tag
<li> li tag
</ol>
<li> Unordered List
<ol type="1">
<li> ul tag
<li> li tag
</ol>
</ol>
```

Output:

Three Major Types Of Lists:

1. Definition List
 - E. dd tag
 - F. dl tag
 - G. dt tag
2. Ordered List
 - i. ol tag
 - ii. li tag
3. Unordered List
 1. ul tag
 2. li tag

TAG: optgroup

<optgroup> ... </optgroup>

Available in versions: 4.0

Browser compatibility: Netscape 6

The **<optgroup>** tag is used to group together related items in a pull-down form control. An item can only be added to the group by using the **option** tag. Both the **optgroup** and **option** tags must be used inside the opening and closing **select** tags.

It is expected that future versions of HTML may allow this tag to create nested pull-down lists.

This tag is currently very poorly supported by browsers.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

disabled

The **disabled** attribute is used to disable the caption group. The only permitted value is disabled.

label

The **label** attribute is used to specify a short caption.

Code:

```
<select>
<optgroup label="Mexican tropical fish">
<option>Molly</option>
<option>Platy</option>
<option>Sword Tail</option>
</optgroup>
<optgroup label="Goldfish">
<option>Comet</option>
<option>Common</option>
<option>Sarasa</option>
</optgroup>
</select>
```

TAG: option

`<option> ... </option>`

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<option>** tag is used to insert one item in a pull-down form control. You may repeat this tag an unlimited number of times in a pull-down list. (For example, you could list all fifty states.) By default, when an item is chosen, the value that is added to the contents of the **form** is the text that appears immediately after the opening **option** tag. However, you can specify a different value by using the **value** attribute.

This tag must be placed between the opening and closing **select** tags. The order of the **option** tags determines the default order of the list. However, you can use the **selected** attribute to specify one item always to appear at the top of the list.

In general, a pull-down list should be treated as a menu. It is recommended that the text for each item in the list be short, such as one or two words. A long text can be far too unwieldy to read and will probably annoy most users.

The ending tag is optional. However, it is recommended that it be included.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes disabled

The **disabled** attribute is a Boolean value that, if present, prevents the form control (field) from functioning. In some browsers, the control (field) will appear to be greyed out.

label

The **label** attribute is used to create a short text version of the text that appears after the opening **option** tag.

selected

The **selected** attribute is used to specify which one item in the pull-down will appear at the top of the list. Some browsers highlight the selected item. The default is that the list is in the same order as the **option** tags and there is no highlighting

value

The **value** attribute is used when you need to set a different value for the item in the pull-down that what is listed in the **option** tag.

Code:

```
<select size="2">  
<option selected>Angel fish</option>  
<option>Cardinal tetra</option>  
<option>Cat fish</option>  
<option>Discus</option>  
<option>Hatchet fish</option>  
<option>Neon tetra</option>  
</select>
```


TAG: p

<p> ... </p>

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<p>** tag is used to signify the beginning of a paragraph and is analogous to having both a line break and a carriage return occur. The flow of the display of the text and any images is halted on the current line, an entire line is skipped (or a certain amount of white vertical space), and then the flow resumes starting at the left margin (by default).

The closing tag is optional. When another **p** is encountered, it has the effect of an implied ending **/p** tag. Therefore, closing **/p** tags are rarely used. However, it is important to understand that the **p** tag marks the start of the paragraph and that this tag serves as a container for the text and elements that follow. Depending on the complexity of the layout of your HTML document, you may need to use the closing tag to maintain the page appearance you desire.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

align deprecated 4.0

The **align** attribute is used to determine where and how the paragraph will be placed on the page. The permitted values are **center**, **justify**, **left**, and **right**. This attribute is deprecated effective with version 4.0. You are now to use style sheets.

Code:

```
My favorite fish!  
<p align="center">  
The pseudotrophus "Acei" cichlid came from Lake Malawi in Africa...  
<p align="right">  
...while the Showa Koi in my pond came from Japan.  
</p>  
<p>  
Do you have favorites?
```

Output:

My favorite fish!

The pseudotropheus "Acei" cichlid came from Lake Malawi in Africa...

...while the Showa Koi in my pond came from Japan.

Do you have favorites?

TAG: param

<param>

Available in versions: 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 6

The **<param>** tag is used to set the name/value pair that provides a parameter required by a Java object in an **applet** tag or by some object in an **object** tag. Each **param** tag can provide only one parameter. However, there can be any number of parameters (from zero to many). These parameters, including the name and acceptable range of values (defaults, etc.), are established by the author of the applet or object. For example, they could set the color of the applet.

This tag is placed immediately after the opening **applet** or **object** tag. There is no closing tag.

Core Attributes

id

Attributes

name

The **name** attribute is the name portion of the name/value pair. You must provide a value for this attribute. The exact value (spelling, etc.) of the **name** will be determined by the author of the applet or object.

type

The **type** attribute defines the MIME media type of the value when the **valuetype** attribute is set to **ref**.

value

The **value** attribute provides the value required by the parameter for the name/value pair. You must provide a value for this parameter.

valuetype

The **valuetype** attribute defines the type of the value. There are three permitted values **data**, **object**, and **ref**. The default is **data**. The **ref** is the URL source that contains the value.

Code:

```
<applet codebase="clock/classes" code="JavaClock.class" width="150" height="150">  
<param name="timezone" value="eastern">  
</applet>
```

TAG: pre

`<pre> ... </pre>`

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The `<pre>` tag is used to display pre-formatted text. The output will mimic exactly how the text is rendered inside the `pre` element including white space, tabs, and line breaks. This allows you to maintain the appearance of data in rows and columns, or have extra white spaces in text such as poems. Perhaps one of the most common uses of this tag is to display computer code and output.

The browser will display the text in a mono spaced font. However, you can use style sheets, including the `style` attribute, to effect the appearance of the text to suit your desires.

The following tags may not appear inside a `pre` element: **big**, **img**, **object**, **small**, **sub**, and **sup**. Therefore, images (which require an `img` tag) cannot appear in a `pre`.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#) [title](#)

Attributes

width deprecated 4.0

The **width** attribute is used to set the maximum number of characters that can be displayed on one line. This attribute is deprecated effective with version 4.0. However, most browsers continue to recognize this attribute.

The example is shown twice, with and without the use of style sheets. (Note that HTML does not show the extra white spaces that are present in the code.)

Code without style:

```
<pre>
function CheckFrames()
{
var strPage, strURL
strPage = location.pathname + location.search
strURL = location.protocol + "://" + location.hostname +
"/index.asp?page=" + escape(strPage)

if ((window.name != "body") || (window.parent.name != "frameset") ||
(window.parent.parent.parent != window.parent.parent))
{
window.top.location.replace(strURL)
}
}
}
</pre>
```

Code with style:

```
<pre style="font-family:arial; color:red; font-size:12px;">
function CheckFrames()
{
```

```
var strPage, strURL
strPage = location.pathname + location.search
strURL = location.protocol + "://" + location.hostname +
"/index.asp?page=" + escape(strPage)

if ((window.name != "body") || (window.parent.name != "frameset") ||
(window.parent.parent.parent != window.parent.parent))
{
window.top.location.replace(strURL)
}
}
</pre>
```

Output without style:

```
function CheckFrames()
{
    var strPage, strURL
    strPage = location.pathname + location.search
    strURL = location.protocol + "://" + location.hostname +
        "/index.asp?page=" + escape(strPage)

    if ((window.name != "body") || (window.parent.name != "frameset") ||
        (window.parent.parent.parent != window.parent.parent))
    {
        window.top.location.replace(strURL)
    }
}
```

Output with style:

```
function CheckFrames()
{
    var strPage, strURL
    strPage = location.pathname + location.search
    strURL = location.protocol + "://" + location.hostname +
        "/index.asp?page=" + escape(strPage)

    if ((window.name != "body") || (window.parent.name != "frameset") ||
        (window.parent.parent.parent != window.parent.parent))
    {
        window.top.location.replace(strURL)
    }
}
```

TAG: q

<q> ... </q>

Available in versions: 4.0

Browser compatibility: Netscape 6

The **<q>** tag takes the selected block of text and displays it as a quote which is enclosed by a pair of double quotes. Unlike the **blockquote** tag, the **q** quote is not preceded or followed by paragraph breaks. So the quote remains inline. You may still need to use other HTML tags to effect the textual display and appearance that you desire.

Since this tag is poorly supported, you should simply use quotation marks.

It is recommended that you use the **blockquote** tag for long citations and the **cite** tag for short citations.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

cite

The **cite** attribute is used to provide a reference to the URL that is the source of the quote.

Few browsers recognize this tag.

Code:

```
<q>His enthusiasm was only exceeded by his lack of enthusiasm!</q>
```

Output:

His enthusiasm was only exceeded by his lack of enthusiasm!

TAG: s deprecated 4.0

<s> ... </s>

Available in versions: 3.2

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<s>** tag is officially deprecated effective with HTML 4.0. Instead, you are now to use the style sheets **text-decoration** property. However, this tag remains in minor use on the Internet and is still recognized by all major browsers.

The **s** tag causes the designated text to be displayed with a struck through horizontal line. The **strike** tag performs the exact same function, however, it has also been deprecated.

The closing tag is mandatory.

Core attributes:

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None

Code:

I have `<s>one hundred twenty four</s>`, oops, one hundred twenty three, Neon Tetras in my 45 gallon aquarium.

Output:

I have ~~one hundred twenty four~~, oops, one hundred twenty three, Neon Tetras in my 45 gallon aquarium.

TAG: samp

<samp> ... </samp>

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<samp>** tag is designed to place emphasis on a short string of special characters. The designated text is displayed in a teletype or mono space font face. The actual appearance will be determined by the browser.

Usually, the display font appearance for the **code**, **kbd**, **samp**, and **tt** tags is the same. The **listing** tag has been deprecated.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None.

Code:

```
<samp>DevGuru</samp>
```

Output:

DevGuru

TAG: small

<small> ... </small>

Available in versions: 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<small>** tag causes the designated text to be displayed in a smaller font size. The actual amount of change will be determined by the browser. If the text is already at the minimum size, this attribute will have no effect.

This can also be done using the style sheets **font-size** property.

The closing tag is mandatory.

Core attributes:

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None

Code:

```
<big>THINK</big> very <small>small</small>
```

Output:

THINK very small

TAG: span

** ... **

Available in versions: 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **** tag is used to designate an inline portion of an HTML document as a **span** element and to apply any of the core attributes of this tag against the contents of that element. The behavior of this tag is inline because the effects occur in the normal flow of the text and images (without the tag inserting any additional linefeeds or carriage returns). **span** tags, which can be nested, allow a significant degree of control and manipulation of a localized part of your web page.

For example, you could use the **class** or **style** core attributes to apply the effects of Cascading Style Sheets or you could use the **lang** attribute to display a foreign language text.

In contrast, you should use the **div** tag when you want to apply attributes to a block of code. Note that the behavior of the **div** tag is not inline, but rather is block-line, since this tag causes a line break to occur both before and after the **div** element.

With the advent of Cascading Style Sheets, both the **div** and **span** tags have gained importance and usefulness.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#) [onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#) [title](#)

Attributes

None.

As an example, all of the code samples on the **DevGuru** site are in blue colored text. This is done by enclosing the code sample inside a pair of **span** tag elements and designating a **class** attribute of "CODE". In turn, the "CODE" class has been assigned a text color of blue in the style sheets file (a .css file) that is located in the Include/ directory on the **DevGuru** web site. Thus, only one file is needed to define the appearance of the entire site.

Code:

```
<span class="CODE">
```

```
This line is blue.
```

```
</span>
```

Output:

This line is blue.

However, if you do not wish to create a .css file, you can still get the exact same results by using the **style** attribute for the **span** tag and the style sheet **color** property.

Code:

```
<span style="color: blue;">
```

This line is blue.

Output:
This line is blue.

TAG: ~~strike~~ deprecated 4.0

<strike> ... </strike>

Available in versions: 3.2

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<strike>** tag is officially deprecated effective with HTML 4.0. Instead, you are now to use the style sheets **text-decoration** property. However, this tag remains in minor use on the Internet and is still recognized by all major browsers.

The **strike** tag causes the designated text to be displayed with a struck through horizontal line. The **s** tag performs the exact same function, however, it also is deprecated.

The closing tag is mandatory.

Core attributes:

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None

Code:

I have ~~one hundred twenty four~~, oops, one hundred twenty three, Neon Tetras in my 45 gallon aquarium.

Output:

I have ~~one hundred twenty four~~, oops, one hundred twenty three, Neon Tetras in my 45 gallon aquarium.

TAG: strong

** ... **

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **** tag is used to provide a strong emphasis when displaying the enclosed word or phrase. Most browsers do this by rendering the selected text in bold which makes the characters thicker and slightly bigger in appearance than the surrounding text. Of course, you can also do this using the **b** tag.

If you wish to have less emphasis, you can use the **em** tag which usually renders in italics, rather than in bold.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None.

Code:

Did you know that the **Guru** is over **800** years old?

**
**

Did you know that the **Guru** is over **800** years old?

Output:

Did you know that the **Guru** is over **800** years old?

Did you know that the **Guru** is over 800 years old?

TAG: sub

_{...}

Available in versions: 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<sub>** tag is used to insert subscript into a text. This allows the correct display of chemical equations, footnotes, math formulas, and scientific notation on a web page.

Most browsers reduce the text size by half for the subscript (and superscript), but use the same font face as used for the rest of the text. However, you can use style sheets and the **style** attribute to effect the appearance of the text to suit your desires.

The similar **sup** tag is used to create superscripts.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None.

Code:
2H₂+O₂ -> 2H₂O

Output:
2H₂+O₂ -> 2H₂O

TAG: sup

^{...}

Available in versions: 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<sup>** tag is used to insert superscript into a text. This allows the correct display of footnotes, math formulas, and scientific notation on a web page. Most browsers reduce the text size by half for the superscript (and subscript), but use the same font face as used for the rest of the text.

However, you can use style sheets and the **style** attribute to effect the appearance of the text to suit your desires.

The similar **sub** tag is used to create subscripts.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None.

Code:

```
E = MC<sup>2</sup>
```

Output:

E = MC²

TAG: table

<table> ... </table>

Available in versions: 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<table>** tag is used to designate that this element is a table. A table is a structural presentation of data or information using rows and columns.

The default is that the flow of any text and images is stopped in the HTML document, the **table** element is inserted starting on the next line, and after the end of the **table** element is reached, the flow of text and images resumes on the following line. However, the use of attributes and style sheet properties will allow you to embed the **table** element within the text and images.

The insides of the **table** are constructed using the **tr**, **th**, **td**, and the **caption** tags. There are also three new table tags that were released with version 4.0. They are **tbody**, **tfoot**, and **thead**.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#) [onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#) [title](#)

Attributes

align deprecated 4.0

The **align** attribute is used to align a table in the **center**, along the **left** margin, or the **right** margin of the display window. This permits the table to be embedded within the flow of the text and images. This attribute was deprecated effective with version 4.0. However, it remains in wide use and is recognized by most browsers. You are now to use the style sheets **float** or **text-align** property.

bgcolor deprecated 4.0

The **bgcolor** attribute is used to set the background color of a table. It can be either a named color (such as red) or a six character hexadecimal RGB color code (such as #e26fa9). This will be treated as the default color, since individual **tr** and **td** cells can also have their color set. This attribute was deprecated effective with version 4.0. However, it remains in wide use and is recognized by most browsers. You are now to use style sheets.

border

The **border** attribute informs the browser to draw lines around the entire table and all of the rows and cells. The lines are shaded to give an embossed appearance. You declare the thickness of the border line in pixels as an integer. The default is no border (zero pixels).

cellpadding

The **cellpadding** attribute sets the amount of white space to place between a cell wall and the contents of a cell. You declare the amount in pixels as an integer. The default is one pixel.

cellspacing

The **cellspacing** attribute sets the amount of white space to place between each adjacent cell and between each cell and the outer border of the table. You declare the amount in pixels as

an integer. The default is two pixels.

frame

The **frame** attribute is used to determine which of the four lines that make up the outside border of a table will be visible. The permitted values are **above**, **below**, **border**, **box**, **hsides**, **lhs**, **rhs**, **void**, and **vsides**.

rules

The **rules** attribute is used to control whether the border lines will appear around cells inside the table. You can specify **rows** only to have borders between rows, **cols** for only between columns, and **none** for no internal borders.

summary

The **summary** attribute is used with non-visual media such as braille and speech to provide information about a table.

width

The **width** attribute is used to set the width of a table. It can be declared either as an integer number of pixels or as a percentage of the width of the display window.

This example demonstrates the basic table tags:

```
Code:
<table width="80%" border="10" cellpadding="3" cellspacing="5" align="center">
```

```
<caption>
<center><b>How To Code A Table</b></center>
<br>
```

```
Use the caption tag to name the table and to provide useful information about the contents
</caption>
```

```
<tr>
<th colspan="2">Use the <b>th</b> tag to create the header which will display in bold</th>
</tr>
```

```
<tr>
<td width="50%">The <b>tr</b> tag creates the row</td>
<td width="50%">The <b>td</b> tag creates individual cells for each row</td>
</tr>
```

```
<tr>
<td width="50%">You should always use the closing tags</td>
<td width="50%">You can use a variety of HTML tags inside the cell</td>
</tr>
```

```
<tr>
<td width="50%">The three new version 4.0 table tags are <b>tbody, tfoot,</b> and
<b>thead</b></td>
```

```
<td width="50%">The <b>colspan</b> attribute allows you span more than one cell</td>
</tr>
```

```
</table>
```

Output:

How To Code A Table

Use the caption tag to name the table and to provide useful information about the contents

Use the **th** tag to create the header which will display in bold

The **tr** tag creates the row

The **td** tag creates individual cells for each row

You should always use the closing tags

You can use a variety of HTML tags inside the cell

The three new version 4.0 table tags are **tbody**, **tfoot**, and **thead**

Setting **colspan** allows you span more than one cell

TAG: tbody

`<tbody> ... </tbody>`

Available in versions: 4.0

Browser compatibility: Explorer 4, 5 Netscape 6

The **<tbody>** tag is one of three new version 4.0 tags that divide a **table** into three distinct sections.

The **tbody** tag defines the body portion where the data is displayed, the **thead** tag defines the header section, and the **tfoot** tag defines the footnote section of the **table**. The purpose of this division is to allow scrolling through the body of a large table while both the header and footnote portions remain visible.

Unfortunately, no browser truly implements these three new tags.

The closing tag is optional. However, it is recommended that it always be included.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

align

The **align** attribute is used to set the horizontal alignment of the cell contents for all of the cells in a single row. The five permitted values are **center**, **char**, **justify**, **left**, and **right**.

char

The **char** attribute permits you to select a character that will be used to align the contents of all of the rows in a cell. This was designed to allow the decimal points in a column of numbers to be in vertical alignment. The default value is country dependent. In the U.S., it is the first occurrence of a period or a decimal point.

charoff

The **charoff** attribute is an integer used to specify if the value designated in the **char** attribute is to be the first, second, or whatever occurrence.

valign

The **valign** attribute is used to set the vertical alignment of the cell contents for all of the cells in a single row. The possible values are **baseline**, **bottom**, **middle**, and **top**

Note the the correct order for using these tags is **thead**, then **tfoot**, and last **tbody**.

Code:

```
<table border="1">  
<caption>  
<center>  
<b>TABLE</b>  
<br>  
thead, tfoot & tbody  
</center>
```

```
</caption>
<thead>
<tr><th>Header 1</th><th>Header 2</th></tr>
</thead>
<tfoot>
<tr><td colspan="2">footnote</td></tr>
</tfoot>
<tbody>
<tr><td>data cell 1</td><td>data cell 2</td></tr>
</tbody>
</table>
```

Output:

TABLE

thead, tfoot & tbody

Header 1	Header 2
footnote	
data cell 1	data cell 2

TAG: td

`<td> ... </td>`

Available in versions: 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<td>** tag is used to create cells that contain the data, information, or text that you wish to display in the **table**. You may place as many data cells in a row as you desire.

The **th** tag is used to create a header cell for the cells in a row in a **table** element.

The ending tag is optional. However, it is recommended that it be included.

The coding sequence is:

```
<tr><td> place data here </td></tr>
```

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

abbr

The **abbr** attribute is not recognized by most browsers. You use it to provide an abbreviated summary of the data cell text.

align

The **align** attribute is used to set the horizontal alignment of the cell contents. The possible values are **center**, **char**, **justify**, **left**, and **right**.

axis

The **axis** attribute is not recognized by most browsers. This attribute is used to create a list of category names which form the basis of a query. This query is executed against the cells that form the table.

bgcolor

The **bgcolor** attribute is deprecated effective with version 4.0. You are now to use style sheets. This attribute is used to set the background color for the cell. It can be either a named color (such as red) or a six character hexadecimal RGB color code (such as #e26fa9). This value overrides any color value that was set in the **table** or **tr** tags.

char

The **char** attribute permits you to select a character that will be used to align the contents of the cell. This was designed to allow the decimal points in a column of numbers to be in vertical alignment. The default value is country dependent. In the U.S., it is the first occurrence of a period or a decimal point.

charoff

The **charoff** attribute is an integer used to specify if the value designated in the **char** attribute is to be the first, second, or whatever occurrence.

colspan

The **colspan** attribute allows a header to span horizontally two or more columns (cells). This may be combined with the **rowspan** attribute to form data cells that encompass several rows and columns.

headers

The **headers** attribute is not recognized by most browsers. The value of this attribute is a list of cell names. The names are the values of the cells' **id** attributes. This attribute is used with speech capable browsers.

height

The **height** attribute is deprecated effective with version 4.0. You are now to use style sheets. You use this attribute to set a minimum height for the cell. This value is an integer number of pixels. However, you may exceed this minimum height to accommodate the cell contents. The tallest cell in the row determines the actual maximum height which will be adopted by all of the cells in that row.

nowrap

The **nowrap** attribute is deprecated effective with version 4.0. You are now to use style sheets. You use this attribute to stop word wrapping and display the cell contents in one continuous line. However, you can insert **br** and **p** tags to create line and paragraph breaks.

rowspan

The **rowspan** attribute allows a cell to extend down two or more rows. This may be combined with the **colspan** attribute to form data cells that encompass several rows and columns.

scope

The **scope** attribute is not recognized by most browsers. It is used to assign a set of data cells to one header cell. The four possible values are **col**, **colgroup**, **row**, and **rowgroup**.

valign

The **valign** attribute is used to set the vertical alignment of the cell contents. The possible values are **baseline**, **bottom**, **middle**, and **top**.

width

The **width** attribute is deprecated effective with version 4.0. You are now to use style sheets. It is used to set the horizontal width of the cell. This value is an integer number of pixels. The widest cell sets the width for the entire column.

This example demonstrates the basic table tags:

Code:
`<table width="80%" border="10" cellpadding="3" cellspacing="5" align="center">`

`<caption>`

`<center>How To Code A Table</center>`

`
`

Use the caption tag to name the table and to provide useful information about the contents

`</caption>`

`<tr>`

`<th colspan="2">Use the th tag to create the header which will display in bold</th>`

`</tr>`

`<tr>`

`<td width="50%">The tr tag creates the row</td>`

`<td width="50%">The td tag creates individual cells for each row</td>`

`</tr>`

`<tr>`

```
<td width="50%">You should always use the closing tags</td>
<td width="50%">You can use a variety of HTML tags inside the cell</td>
</tr>
<tr>
<td width="50%">The three new version 4.0 table tags are <b>tbody, tfoot,</b> and
<b>thead</b></td>
<td width="50%">The <b>colspan</b> attribute allows you span more than one cell</td>
</tr>
</table>
```

Output:

How To Code A Table

Use the caption tag to name the table and to provide useful information about the contents

Use the th tag to create the header which will display in bold	
The tr tag creates the row	The td tag creates individual cells for each row
You should always use the closing tags	You can use a variety of HTML tags inside the cell
The three new version 4.0 table tags are tbody , tfoot , and thead	Setting colspan allows you span more than one cell

TAG: tfoot

<tfoot> ... </tfoot>

Available in versions: 4.0

Browser compatibility: Explorer 4, 5 Netscape 6

The **<tfoot>** tag is one of three new version 4.0 tags that divide a **table** into three distinct sections.

The **tbody** tag defines the body portion where the data is displayed, the **thead** tag defines the header section, and the **tfoot** tag defines the footnote section of the **table**. The purpose of this division is to allow scrolling through the body of a large table while both the header and footnote portions remain visible.

Unfortunately, no browser truly implements these three new tags.

The closing tag is optional. However, it is recommended that it always be included.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

align

The **align** attribute is used to set the horizontal alignment of the cell contents for all of the cells in a single row. The five permitted values are **center**, **char**, **justify**, **left**, and **right**.

char

The **char** attribute permits you to select a character that will be used to align the contents of all of the rows in a cell. This was designed to allow the decimal points in a column of numbers to be in vertical alignment. The default value is country dependent. In the U.S., it is the first occurrence of a period or a decimal point.

charoff

The **charoff** attribute is an integer used to specify if the value designated in the **char** attribute is to be the first, second, or whatever occurrence.

valign

The **valign** attribute is used to set the vertical alignment of the cell contents for all of the cells in a single row. The possible values are **baseline**, **bottom**, **middle**, and **top**.

Note the the correct order for using these tags is **thead**, then **tfoot**, and last **tbody**.

Code:

```
<table border="1">  
<caption>  
<center>  
<b>TABLE</b>  
<br>  
thead, tfoot & tbody  
</center>
```

```
</caption>
<thead>
<tr><th>Header 1</th><th>Header 2</th></tr>
</thead>
<tfoot>
<tr><td colspan="2">footnote</td></tr>
</tfoot>
<tbody>
<tr><td>data cell 1</td><td>data cell 2</td></tr>
</tbody>
</table>
```

Output:

TABLE

thead, tfoot & tbody

Header 1	Header 2
footnote	
data cell 1	data cell 2

TAG: th

<th> ... </th>

Available in versions: 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<th>** tag is used to create a header cell for the cells in a row in a **table** element. The purpose of the header is to provide a caption, name, or information about the column of cells. The header text will be displayed in a bold font style.

The **td** tag is used to create cells that contain the data, information, or text that you wish to display in the **table**.

The closing tag is optional. However, it is recommended that it always be included.

The coding sequence is:

```
<tr><th> place header here </th></tr>
```

Core attributes:

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

abbr

The **abbr** attribute is not recognized by most browsers. You use it to provide an abbreviated summary of the header text.

align

The **align** attribute is used to set the horizontal alignment of the cell contents. The possible values are **center**, **char**, **justify**, **left**, and **right**.

axis

The **axis** attribute is not recognized by most browsers. This attribute is used to create a list of category names which form the basis of a query. This query is executed against the cells that form the table.

bgcolor

The **bgcolor** attribute is deprecated effective with version 4.0. You are now to use style sheets. This attribute is used to set the background color for the cell. It can be either a named color (such as red) or a six character hexadecimal RGB color code (such as #e26fa9). This value overrides any color value that was set in the **table** or **tr** tags.

char

The **char** attribute permits you to select a character that will be used to align the contents of the cell. This was designed to allow the decimal points in a column of numbers to be in vertical alignment. The default value is country dependent. In the U.S., it is the first occurrence of a period or a decimal point.

charoff

The **charoff** attribute is an integer used to specify if the value designated in the **char** attribute is to be the first, second, or whatever occurrence.

colspan

The **colspan** attribute allows a header to span horizontally two or more columns (cells). This may be combined with the **rowspan** attribute to form data cells that encompass several rows and columns.

headers

The **headers** attribute is not recognized by most browsers. The value of this attribute is a list of cell header names. The names are the values of the cells' **id** attributes. This attribute is used with speech capable browsers.

height

The **height** attribute is deprecated effective with version 4.0. You are now to use style sheets. You use this attribute to set a minimum height for the cell. This value is an integer number of pixels. However, you may exceed this minimum height to accommodate the cell contents. The tallest cell in the row determines the actual maximum height which will be adopted by all of the cells in that row.

nowrap

The **nowrap** attribute is deprecated effective with version 4.0. You are now to use style sheets. You use this attribute to stop word wrapping and display the cell contents in one continuous line. However, you can insert **br** and **p** tags to create line and paragraph breaks.

rowspan

The **rowspan** attribute allows a cell to extend down two or more rows. This may be combined with the **colspan** attribute to form data cells that encompass several rows and columns.

scope

The **scope** attribute is not recognized by most browsers. It is used to assign a set of data cells to one header cell. The four possible values are **col**, **colgroup**, **row**, and **rowgroup**.

valign

The **valign** attribute is used to set the vertical alignment of the cell contents. The possible values are **baseline**, **bottom**, **middle**, and **top**.

width

The **width** attribute is deprecated effective with version 4.0. You are now to use style sheets. It is used to set the horizontal width of the cell. This value is an integer number of pixels. The widest cell sets the width for the entire column.

This example demonstrates the basic table tags:

```
Code:
<table width="80%" border="10" cellpadding="3" cellspacing="5" align="center">
<caption>
<center><b>How To Code A Table</b></center>
<br>
Use the caption tag to name the table and to provide useful information about the contents
</caption>
<tr>
<th colspan="2">Use the <b>th</b> tag to create the header which will display in bold</th>
</tr>
<tr>
<td width="50%">The <b>tr</b> tag creates the row</td>
<td width="50%">The <b>td</b> tag creates individual cells for each row</td>
```

```
</tr>
<tr>
<td width="50%">You should always use the closing tags</td>
<td width="50%">You can use a variety of HTML tags inside the cell</td>
</tr>
<tr>
<td width="50%">The three new version 4.0 table tags are <b>tbody, tfoot,</b> and
<b>thead</b></td>
<td width="50%">The <b>colspan</b> attribute allows you span more than one cell</td>
</tr>
</table>
```

Output:

How To Code A Table

Use the caption tag to name the table and to provide useful information about the contents

Use the th tag to create the header which will display in bold	
The tr tag creates the row	The td tag creates individual cells for each row
You should always use the closing tags	You can use a variety of HTML tags inside the cell
The three new version 4.0 table tags are tbody , tfoot , and thead	Setting colspan allows you span more than one cell

TAG: thead

<thead> ... </thead>

Available in versions: 4.0

Browser compatibility: Explorer 4, 5 Netscape 6

The **<thead>** tag is one of three new version 4.0 tags that divide a **table** into three distinct sections.

The **tbody** tag defines the body portion where the data is displayed, the **thead** tag defines the header section, and the **tfoot** tag defines the footnote section of the **table**. The purpose of this division is to allow scrolling through the body of a large table while both the header and footnote portions remain visible and fixed in place.

Unfortunately, no browser truly implements these three new tags.

The closing tag is optional. However, it is recommended that it always be included.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

align

The **align** attribute is used to set the horizontal alignment of the cell contents for all of the cells in a single row. The five permitted values are **center**, **char**, **justify**, **left**, and **right**.

char

The **char** attribute permits you to select a character that will be used to align the contents of all of the rows in a cell. This was designed to allow the decimal points in a column of numbers to be in vertical alignment. The default value is country dependent. In the U.S., it is the first occurrence of a period or a decimal point.

charoff

The **charoff** attribute is an integer used to specify if the value designated in the **char** attribute is to be the first, second, or whatever occurrence.

valign

The **valign** attribute is used to set the vertical alignment of the cell contents for all of the cells in a single row. The possible values are **baseline**, **bottom**, **middle**, and **top**.

Note the the correct order for using these tags is **thead**, then **tfoot**, and last **tbody**.

Code:

```
<table border="1">  
<caption>  
<center>  
<b>TABLE</b>  
<br>  
thead, tfoot & tbody  
</center>
```

```
</caption>
<thead>
<tr><th>Header 1</th><th>Header 2</th></tr>
</thead>
<tfoot>
<tr><td colspan="2">footnote</td></tr>
</tfoot>
<tbody>
<tr><td>data cell 1</td><td>data cell 2</td></tr>
</tbody>
</table>
```

Output:

TABLE

thead, tfoot & tbody

Header 1	Header 2
footnote	
data cell 1	data cell 2

TAG: tr

`<tr> ... </tr>`

Available in versions: 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The `<tr>` tag is used to create a row in a **table** element. You can create as many rows as you wish.

This row can contain one or more cells where each cell can contain data, information, or text that you wish to display in the **table**. These cells are created by using either the **td** or **th** tags.

The closing tag is optional. However, it is recommended that it always be included. Also, **tr** tags should never appear inside of **th**, **td**, or other **tr** tags.

The coding sequence is:

```
<tr><th> place header here </th></tr>
```

And:

```
<tr><td> place data here </td></tr>
```

Core attributes:

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

align

The **align** attribute is used to set the horizontal alignment of the cell contents for all of the cells in a single row. The five permitted values are **center**, **char**, **justify**, **left**, and **right**.

bgcolor

The **bgcolor** attribute sets the background color for all of the cells in the row. It can be either a named color (such as red) or a six character hexadecimal RGB color code (such as #e26fa9). This value overrides any value that was set in the **table** tag.

char

The **char** attribute permits you to select a character that will be used to align the contents of all of the rows in a column. This was designed to allow the decimal points in a column of numbers to be in vertical alignment. The default value is country dependent. In the U.S., it is the first occurrence of a period or a decimal point.

charoff

The **charoff** attribute is an integer used to specify if the value designated in the **char** attribute is to be the first, second, or whatever occurrence.

valign

The **valign** attribute is used to set the vertical alignment of the cell contents for all of the cells in a single row. The possible values are **baseline**, **bottom**, **middle**, and **top**.

This example demonstrates the basic table tags:

Code:

```
<table width="80%" border="10" cellpadding="3" cellspacing="5" align="center">
<caption>
<center><b>How To Code A Table</b></center>
<br>
Use the caption tag to name the table and to provide useful information about the contents
</caption>
<tr>
<th colspan="2">Use the <b>th</b> tag to create the header which will display in bold</th>
</tr>
<tr>
<td width="50%">The <b>tr</b> tag creates the row</td>
<td width="50%">The <b>td</b> tag creates individual cells for each row</td>
</tr>
<tr>
<td width="50%">You should always use closing tags</td>
<td width="50%">You can use a variety of HTML tags inside the cell</td>
</tr>
<tr>
<td width="50%">The three new version 4.0 table tags are <b>tbody, tfoot,</b> and
<b>thead</b></td>
<td width="50%">The <b>colspan</b> attribute allows you span more than one cell</td>
</tr>
</table>
```

Output:

How To Code A Table

Use the caption tag to name the table and to provide useful information about the contents

Use the th tag to create the header which will display in bold	
The tr tag creates the row	The td tag creates individual cells for each row
You should always use the closing tags	You can use a variety of HTML tags inside the cell
The three new version 4.0 table tags are tbody , tfoot , and thead	Setting colspan allows you span more than one cell

TAG: tt

<tt> ... </tt>

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<tt>** tag causes the designated text to be displayed in a teletype or mono space font face. The actual appearance will be determined by the browser.

Usually, the display font appearance for the **code**, **kbd**, **samp**, and **tt** tags is the same. The **listing** tag has been deprecated.

The closing tag is mandatory.

Core attributes:

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None

Code:

```
<tt>The tt tag is named teletype.</tt>
```

Output:

```
The tt tag is named teletype.
```

TAG: `u` deprecated 4.0

`<u> ... </u>`

Available in versions: 3.2

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The `<u>` tag is officially deprecated effective with HTML 4.0. Instead, you are now to use the style sheets **text-decoration** property. However, this tag remains in moderate use on the Internet and is still recognized by all major browsers.

The `<u>` tag causes the designated text to be underlined.

The closing tag is mandatory.

Core attributes:

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None

Code:

Remember, the default for `<u>hyperlinks</u>` is to be `<u>underlined</u>`.

Output:

Remember, the default for [hyperlinks](#) is to be [underlined](#).

TAG: ul

** ... **

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **** tag is used to delimit the start and stop of an unordered list. An unordered list is a collection of items (typically related somehow) that are in no particular order. Each item in the list is preceded by a bullet with the default being a round disc: •

Most browsers separate the list from any preceding and following text by paragraph breaks. You can nest unordered lists and each nested list can have a different bullet.

You must use the **li** tag to display an item in the list.

You can use the **ol** tag to create an ordered list and the **dl** tag to create a definition list. (The **dir** and **menu** list tags are deprecated.)

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

compact deprecated 4.0

The **compact** attribute is used to make an unordered list more compact in appearance. This attribute is poorly implemented. It is deprecated effective with version 4.0.

type deprecated 4.0

The **type** attribute is used to select the type of bullet that appears before each item in the list. The permitted values are a **circle**, **disc**, and **square**. This attribute is poorly implemented. It is deprecated effective with version 4.0.

Code:

Three Major Types Of Lists:

```
<ul>
<li> Definition List
<ul type="square">
<li> dd tag
<li> dl tag
<li> dt tag
</ul>
<li> Ordered List
<ul type="circle">
<li> ol tag
<li> li tag
</ul>
<li> Unordered List
<ul type="disc">
<li> ul tag
```

```
<li> li tag
```

```
</ul>
```

```
</ul>
```

Output:

Three Major Types Of Lists:

- Definition List
 - dd tag
 - dl tag
 - dt tag
- Ordered List
 - ol tag
 - li tag
- Unordered List
 - ul tag
 - li tag

TAG: var

<var> ... </var>

Available in versions: 2.0, 3.2, 4.0

Browser compatibility: Explorer 4, 5 Netscape 4, 6

The **<var>** tag is used to signify that the designated word is a variable name in a listing of computer code. This tag is used inside **code** and **pre** tag elements. On most browsers, the word or text is rendered in an italic or oblique font style.

The closing tag is mandatory.

Core Attributes

[class](#) [dir](#) [id](#) [lang](#) [onclick](#) [ondblclick](#) [onkeydown](#) [onkeypress](#) [onkeyup](#)
[onmousedown](#) [onmousemove](#) [onmouseout](#) [onmouseover](#) [onmouseup](#) [style](#)
[title](#)

Attributes

None.

Code:

```
if (document.getElementById) <var>blnDOM</var> = true;
```

Output:

```
if (document.getElementById) blnDOM = true;
```

Or:

Code:

```
<code>  
if (document.getElementById) <var>blnDOM</var> = true;  
</code>
```

Output:

```
if (document.getElementById) blnDOM = true;
```

Hexadecimal and HTML Named Color Charts.

The chart below gives the hexadecimal numbers for the RGB codes that define the displayed colors.

#EEEEEE	#DDDDDD	#CCCCCC	#BBBBBB	#AAAAAA	#999999	
#888888	#777777	#666666	#555555	#444444	#333333	
#000000	#000033	#000066	#000099	#0000CC	#0000FF	
#330000	#330033	#330066	#330099	#3300CC	#3300FF	
#660000	#660033	#660066	#660099	#6600CC	#6600FF	
#990000	#990033	#990066	#990099	#9900CC	#9900FF	
#CC0000	#CC0033	#CC0066	#CC0099	#CC00CC	#CC00FF	
#FF0000	#FF0033	#FF0066	#FF0099	#FF00CC	#FF00FF	
#003300	#003333	#003366	#003399	#0033CC	#0033FF	
#333300	#333333	#333366	#333399	#3333CC	#3333FF	
#663300	#663333	#663366	#663399	#6633CC	#6633FF	
#993300	#993333	#993366	#993399	#9933CC	#9933FF	
#CC3300	#CC3333	#CC3366	#CC3399	#CC33CC	#CC33FF	
#FF3300	#FF3333	#FF3366	#FF3399	#FF33CC	#FF33FF	
#006600	#006633	#006666	#006699	#0066CC	#0066FF	
#336600	#336633	#336666	#336699	#3366CC	#3366FF	
#666600	#666633	#666666	#666699	#6666CC	#6666FF	
#996600	#996633	#996666	#996699	#9966CC	#9966FF	
#CC6600	#CC6633	#CC6666	#CC3399	#CC66CC	#CC66FF	
#FF6600	#FF6633	#FF6666	#FF6699	#FF66CC	#FF66FF	
#009900	#009933	#009966	#009999	#0099CC	#0099FF	
#339900	#339933	#339966	#339999	#3399CC	#3399FF	
#669900	#669933	#669966	#669999	#6699CC	#6699FF	
#999900	#999933	#999966	#999999	#9999CC	#9999FF	
#CC9900	#CC9933	#CC9966	#CC9999	#CC99CC	#CC99FF	
#FF9900	#FF9933	#FF9966	#FF9999	#FF99CC	#FF99FF	
#00CC00	#00CC33	#00CC66	#00CC99	#00CCCC	#00CCFF	
#33CC00	#33CC33	#33CC66	#33CC99	#33CCCC	#33CCFF	
#66CC00	#66CC33	#66CC66	#66CC99	#66CCCC	#66CCFF	
#99CC00	#99CC33	#99CC66	#99CC99	#99CCCC	#99CCFF	
#CCCC00	#CCCC33	#CCCC66	#CCCC99	#CCCCCC	#CCCCFF	
#FFCC00	#FFCC33	#FFCC66	#FFCC99	#FFCCCC	#FFCCFF	
#00FF00	#00FF33	#00FF66	#00FF99	#00FFCC	#00FFFF	
#33FF00	#33FF33	#33FF66	#33FF99	#33FFCC	#33FFFF	
#66FF00	#66FF33	#66FF66	#66FF99	#66FFCC	#66FFFF	
#99FF00	#99FF33	#99FF66	#99FF99	#99FFCC	#99FFFF	
#CCFF00	#CCFF33	#CCFF66	#CCFF99	#CCFFCC	#CCFFFF	
#FFFF00	#FFFF33	#FFFF66	#FFFF99	#FFFFCC	#FFFFFF	

The following chart shows the HTML named colors in alphabetical order.

Aliceblue		Antiquewhite		Aqua	
Aquamarine		Azure		Beige	
Bisque		Black		Blanchedalmond	
Blue		Blueviolet		Brown	
Burlywood		Cadetblue		Chartreuse	
Chocolate		Coral		Cornflowerblue	
Cornsilk		Crimson		Cyan	
Darkblue		Darkcyan		Darkgoldenrod	
Darkgray		Darkgreen		Darkkhaki	
Darkmagenta		Darkolivegreen		Darkorange	
Darkorchid		Darkred		Darksalmon	
Darkseagreen		Darkslateblue		Darkslategray	
Darkturquoise		Darkviolet		Deeppink	
Deepskyblue		Dimgray		Dodgerblue	
Firebrick		Floralwhite		Forestgreen	
Fuchsia		Gainsboro		Ghostwhite	
Gold		Goldenrod		Gray	
Green		Greenyellow		Honeydew	
Hotpink		Indianred		Indigo	
Ivory		Khaki		Lavender	
Lavenderblush		Lawngreen		Lemonchiffon	
Lightblue		Lightcoral		Lightcyan	
Lightgoldenrodyellow		Lightgreen		Lightgrey	
Lightpink		Lightsalmon		Lightseagreen	
Lightskyblue		Lightslategray		Lightsteelblue	
Lightyellow		Lime		Limegreen	
Linen		Magenta		Maroon	
Mediumaquamarine		Mediumblue		Mediumorchid	
Mediumpurple		Mediumseagreen		Mediumslateblue	
Mediumspringgreen		Mediumturquoise		Mediumvioletred	

Midnightblue		Mintcream		Mistyrose	
Moccasin		Navajowhite		Navy	
Oldlace		Olive		Olivedrab	
Orange		Orangered		Orchid	
Palegoldenrod		Palegreen		Paleturquoise	
Palevioletred		Papayawhip		Peachpuff	
Peru		Pink		Plum	
Powderblue		Purple		Red	
Rosybrown		Royalblue		Saddlebrown	
Salmon		Sandybrown		Seagreen	
Seashell		Sienna		Silver	
Skyblue		Slateblue		Slategray	
Snow		Springgreen		Steelblue	
Tan		Teal		Thistle	
Tomato		Turquoise		Violet	
Wheat		White		Whitesmoke	