-startdde
This switch tells Webber to ignore the command line when appropriate. When this option is not there and a file is double-clicked in Windows Explorer or File Manager, Webber will open the file twice. To avoid this, add the switch.

A Beginners Tutorial

So you want to create a home page... This short tutorial is meant for those just starting out with publishing a page(s) for the World Wide Web.

The first thing to consider is why you are doing it! Are you eager to share information about yourself, your company or a product? Do you plan to offer product support to your customers on the Internet? Is the page just for fun? Do you or your company have a 'serious' corporate image to uphold or are you aiming for 'hip'?

These and many other factors play a part in deciding on the style of page(s) you want to present. It is beyond the scope of this tutorial to try to accommodate the style of any person or organization. It is meant to introduce you to HTML, the World Wide Web and Webber™. This tutorial will use only the HTML 2.0 dtd so that it will work with the most browsers.

Through-out the tutorial there will be links to other topics in the Webber help file. Be sure to follow those links as they offer information that will be helpful in completing the tutorial. After following a link you can use the 'back' button at the top of the help screen to return to the tutorial. As each new element is added it will be highlighted in red in the example. The cursor position will be shown as a purple | where appropriate.

There are tutorials for two categories of pages available here:

1 a basic home page

2 a basic customer response form

We recommend that you do the home page tutorial first since the form tutorial doesn't go into the same level of detail on any elements except those that directly relate to forms. At the end of each tutorial the full example page will be available for you to copy (Edit|Copy) and paste into a new untitled document. You can use this as a template for your own page if you like.

This symbol takes you ahead one level in the tutorial.

This symbol takes you back one level in the tutorial.

Abbreviation

compatibility

<abbrev></abbrev>

Usage:

The <ABBREV> element is used to markup abbreviations. It is used within 'container' tags like paragraphs, headings and preformat tags.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
</HEAD>
<BODY>
<H1>Other information about HTML.</H1>
<P>There is lots of information to be had about HTML on the Internet.
One particularly good set of documentation was written by <AU>Dr. Ian
Graham</AU> at <ABBREV>U. of T.</ABBREV></P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Cerebral Systems Development Corp. is a computer software development company located in Toronto, Ontario. Our products & services include:

- Custom software and database development for Windows or the Mac
- SGML and HTML consulting
- Webber™- HTML Editor Extraordinaire!
- Webber/Dataport™ - generate HTML files from your database.
- Web Page Authoring & Maintenance, HTML Template Design
- HTML Training - using Webber™ of course!

Contact us at admin@csdcorp.com for details on any of the above.

About HTML

HTML stands for HyperText Markup Language. Based on SGML, it is a simple markup language used to define the structure and organization of a document in order to create documents which are portable from one environment to another. An HTML document is the same format on the PC as it is on UNIX, the Mac or any other platform. Tags are used to mark each separate element in a document. For example a paragraph would be surrounded by the paragraph (P) tag. Usually there will be a start tag and an end tag for each element. Tags are enclosed by angle brackets <*tagname*> and end tags also include a forward slash <*/tagname*>.

The tags are used by browsers to format and present the document on screen (and on paper for browsers which can print). Not all browsers will interpret the tags in the same way, and many browsers now allow the user to select their own fonts, background colours, etc.

Every HTML document consists of the HTML identifier <HTML> which includes a header <HEAD> and a body <BODY>. The header contains tags which identify information about a document such as its title, author, usage etc. The body contains tags which hold the text and graphics of a document. A <PLAINTEXT> tag can also be (but is rarely) used within the HTML tag.

The simplest HTML document would look like this:

```
<HTML>
<HEAD>
<TITLE>The document title.</TITLE></HEAD>
<BODY><P>The content of a document will appear here.</P></BODY>
</HTML>
```

A Typical Rendering by a browser would look like this: (the title would probably appear in the title bar of the browser)

The content of a document will appear here.

Above

compatibility

<above></above>

Usage:

Used to draw an arrow, line or symbol above an text contained within this tag. This tag is used only within a <MATH> element.

Attributes:

Required

Optional

sym - used to specify the symbol to be used. If this attribute is not used it defaults to "line".

Example:

```
<above sym="line">x+y</above>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Acknowledgments/Bibliography

Cerebral Systems Development Corp. wished to acknowledge and thank the following individuals and organizations:

 ™

- Kathryn M. Dickson for the design of our friend Webber™ the spider.
- Infrastructures for Information Inc. for their great SGML parser
- The Internet Engineering Task Force (IETF) - for their Internet Drafts, W3 for making them readily available, and Ian Graham at the University of Toronto [http://www.utirc.utoronto.ca/home.html] - for his explanations of HTML, upon which we drew for our descriptions of some of the lesser known HTML elements and attributes.
- Our beta testers, for helping fine tune Webber™
- Our development team for their hard work and long hours on Webber™ and this help file.
- Some of the documentation contained in this help file is based on that available on the Microsoft (http://www.microsoft.com/), Netscape (http://home.netscape.com), and Sun (http://www.sun.com/) web sites. Cerebral Systems Development Corp. acknowledges these resources and does not make any claims to the information itself. It is included in this help file to enable our users to use extensions made to the HTML language by these companies.

Acronym

compatibility

<acronym></acronym>

Usage:

This tag is used to markup acronyms. It is used within 'container' tags like paragraphs, headings and preformat tags.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
</HEAD>
<BODY>
<H1>Other information about HTML.</H1>
<P>There is lots of information to be had about
<ACRONYM>HTML</ACRONYM> on the Internet.</P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Address

compatibility

<address></address>

Usage:

The address tag is used for such information as address, signature and authorship. It is often at the beginning or end of the body of a document.

Most browsers render the address element in an italics and some will also indent.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

Example:

```
<HTML><HEAD>
<TITLE>Webber Home Page</TITLE>
</HEAD>
<BODY>
<H1>Welcome to the Webber Home Page</H1>
<P>This is the home page for the hot, easy to use, HTML editor -
Webber.</P>
<P>Webber was designed as a tool for HTML authors who want to see and
control what is happening in their document at the tag level.</P>
<ADDRESS>Copyright 1995, Cerebral Systems Development
Corp.</ADDRESS></BODY>
</HTML>
```

Typical Rendering (approximate):

Welcome to the Webber Home Page

This is the home page for the hot, easy to use, HTML editor - Webber.

Webber was designed as a tool for HTML authors who want to see and control what is happening in their document at the tag level.

Alias

compatibility

<alias></alias>

Usage:

The ALIAS element is used to define an object without inserting it into the document. It is used with the valueref attribute of the PARAM element to allow an object to be passed as parameter, when initializing an object associated with another OBJECT or ALIAS element. The attributes take exactly the same meaning as for the OBJECT element. The ALIAS element is a container and requires both start and end tags. It can be placed anywhere in the document HEAD or BODY. The contents are limited to PARAM and ALIAS elements, although it is anticipated that this may be extended to cover the same content model as OBJECT at some point in the future.

(This tag is not yet supported as far as we know but has been developed by Microsoft, Netscape, Sun, Spyglass, W3, Pathfinder.)

Attributes:

Required

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

Optional

classid - Used to specify a class identifier for an object.

code - Specifies a URL referencing where to find the code which implements the object's behaviour. If this URL is insufficient to locate the intended object, when for instance, a file contains the implementations for several classes, the CLASSID may be used to supply a disambiguating class identifier.

data - Specifies a URL referencing the object's data.

type - Specifies an Internet Media Type (see RFC 1590) for the object's data. The attribute can be used to allow user agents to quickly skip media they don't support, and instead to render the contents of the INSERT element. It is also useful when loading objects off local drives as it allows the media type to be specified explicitly rather than being derived from the file extension.

Example:

```
<HTML><HEAD>
<TITLE>Demo Document</TITLE></HEAD>
<ALIAS ID="marble" CODE="http://www.acme.com/applets/marble.class">
</ALIAS></HEAD>
<BODY BACKGROUND="#marble">
<P>This document has a marble texture generated by an applet.</P>
</BODY><HTML>
```

Tags in Context

In order to maintain performance and flexibility, Webber does not use context-sensitive tagging for its drop-down tag list. However, there may be times when you are uncertain about which tags are allowed within a certain context. For example, the only tag allowed within an Unordered List (UL) is a List Item (LI) but list items can contain many other tags including unordered lists and paragraphs. You can also use the validation to ensure that a document meets the specifications of the dtd. You can check this topic on Allowed Tags as well.

To find out what tags are allowed in a given place in your document:

1 Place your cursor in the appropriate position.
2 Open the Tag Assistant from the HTML menu or the tool bar. The Tag Assistant can be left open all the time since it floats on your desk-top. It will roll and unroll automatically if you have this setting enabled in your preferences.
3 Check the 'Tags in Context' button at the bottom left of the dialogue box. If you leave the 'Tags in Context' button checked all the time you may experience some delays when you click back on the tag assistant, especially in longer documents.
4 The tag list in the Tag Assistant will now show only the tags allowed at the current cursor position.
5 To insert a tag, select one from the drop-down list and click the Insert button. The tag list will again show only tags allowed in the current cursor position. Note that if you have inserted a new tag, your cursor will now be inside that tag so the context and allowed tags will have changed. To get the context of a new location, click in your document in the new position, then click back on the Tag Assistant.
6 Clicking the Close button will close the dialogue without inserting the currently selected tag. It will not remove tags already inserted by the insert button.

Anchor

compatibility

<a>

Usage:

An anchor is either the source or the target of a hypertext link. As the source the HREF attribute gives the destination for a hypertext link. The content (text or images) contained within the anchor start and end tags is clickable. Any type of file can be referenced by an anchor. Sounds, video clips, image formats not supported in-line, and other html files are all valid destinations. Most browsers will prompt the user for action (i.e. save to disk, launch another program) if the file format of the destination is not recognized.

Anchors should only contain text or text formatting tags (i.e. emphasis.). Headings and other content tags should always be placed around an anchor rather than inside one.

As the target, the NAME attribute provides a destination for hypertext links. In the example below the first two anchors are linked to external documents while the third is a link to a section within the current document marked with the NAME attribute.

In previous versions of HTML it provided the only means for defining destination anchors within documents, but starting with HTML 3.0, you can now use any ID attribute as a destination anchor so that links can now be made to divisions, paragraphs and most other elements.

(See also Anchor Assistant and Quick-Tag Buttons)

For <FIG> elements, the anchor element serves a dual role. Non-graphical user agents interpret it as a conventional text-based hypertext link, while graphical user agents interpret the anchor's SHAPE attribute as a graphical hotzone on the figure. (See the Figure tag topic for an example.)

J This tag has javascript events associated with it. For details see the description of the Javascript event handlers.

Attributes:

Required

href - address of link destination
or
name - destination of link

Optional

rel - relationship to destination
rev - relationship of destination to this document
urn - lasting name of destination *(HTML 2.0 only)*
title - title of destination
methods - operations allowed
N **target** - Browser windows can now have names associated with them. Links in any window can refer to another window by name. When you click on the link, the document you asked for will appear in that named window. If the window is not already open, Netscape will open and

name a new window for you. New in Netscape 2.0. i.e. `<A HREF="url.html" TARGET="window_name"> Click here and open a New Window </A>`.

New Optional Attributes (HTML 3.0 only):

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 md - Used to specify a message digest or cryptographic checksum for the associated HREF.

3 shape - This attribute is used within figures to define shaped hotzones for graphical hypertext links.

Example:

```
<HTML><HEAD>
<TITLE>Webber Index</TITLE>
</HEAD>
<BODY>
<H1>Webber Index</H1>
<P><A HREF="http://www.csdcorp.com/">CSD Home Page</A></P>
<P><A HREF="Welcome.html">Webber Home Page</A></P>
<P><A HREF="#Release">Release Notes</A></P>
<H3><A NAME="Release">Release Notes</A></H3>
<P>These are the release notes for Webber.</P>
</BODY><HTML>
```

Typical Rendering (approximate):

Webber Index

CSD Home Page
Webber Home Page
Release Notes

Release Notes

These are the release notes for Webber.

Anchor Assistant

The Anchor Assistant provides a quick and easy way to add an anchor to your document. It allows you to fill in either of the two most common anchor attributes, HREF and NAME. One of these two attributes is required for an anchor. HREF is used to identify the address of the destination - the URL, while NAME is used at the destination to identify it. Values for the HREF attribute are urls and can include any of: HTTP, Gopher, FTP, WAIS, Telnet, Usenet, and Mailto.

A '#' is used to identify a link to a destination marked with an anchor where the NAME attribute was used. This can be within the same document or in another document. (See the anchor tag topic for examples and more information.)

To Use:

1 Place your cursor where you want the anchor to be inserted. If you want the anchor tag wrapped around existing text, select the text first.
2 Open the Anchor Assistant from the HTML menu or the Quick-Tag button on the tool bar.
3 Select the URL type from the Type drop-down list if it is required. It probably won't be necessary if you are using a relative anchor. (i.e. if the destination of the link is on the same server or in the same document).
4 Enter the HREF URL, or browse for a local file by clicking the browse button. Use the Path Options to tell Webber whether you want absolute, relative or no paths to be included when inserting a file that has been browsed to.
5 If you are placing an anchor as the destination of a link, you should only fill in the NAME attribute.
6 Enter the name of a target, like a frame, if applicable.
7 When finished, click the Ok button to insert the anchor or the Cancel button to leave without inserting the anchor.

Java Applet

compatibility

<APPLET></APPLET>

Usage:

The <APPLET> element has recently replaced the <APP> tag, and is used for embedding a Java Applet into the HTML page. The APPLET tag can contain <PARAM> elements, as well as headers, paragraphs, lists, form elements and other block elements.

We were unable to find documentation for this tag on the Netscape Web site. We would like to acknowledge the work of Stephen L.E. Hunt in his 'HTML Reference Library', from which we gleaned the information for this tag. We obtained his reference from `http://hjs.geol.uib.no/news/htmlib/htmlib.htm.`

Attributes:

Required

code - gives the name of the file, using a path relative to the url of the applet, that contains the applet's compiled Applet subclass.

width - gives the width, in pixels, for the applet display area.

height - gives the height, in pixels, for the applet display area.

Optional

codebase - specifies the URL of the applet. The document's URL is used as a default if this attribute is not used.

align [LEFT | RIGHT | TOP | TEXTTOP | MIDDLE | ABSMIDDLE | BASELINE | BASELINE | ABSBOTTOM] - specifies the alignment of the applet.

alt - specifies any text that should be displayed when the browser recognizes the APPLET tag but can't run the applet.

name - specifies a name for the applet instance, which allows applets on the same page to refer to each other.

vspace - specifies the number of pixels to place above and below the applet

hspace - specifies the number of pixels to place on each side of the applet

Example:

```
<HTML><HEAD>
<TITLE>Java Example</TITLE>
</HEAD>
<BODY>
<H1>Java Example</H1>
<APPLET CODEBASE="http://www.csdcorp.com/applets/myapplet"
CODE="myapplet.class" width="200" height="100" align="center" >
<PARAM NAME="text" VALUE="My Applet Viewer">
<P>If you are using a browser that supports Java, 'myapplet' would be
displayed here instead of this paragraph.</P></APPLET>
</BODY></HTML>
```

Area

compatibility

<area></area>

Usage:

The area tag is used to identify regions of a client-side image map (identified by the USEMAP attribute of the IMG tag). It is contained within a map tag. An arbitrary number of AREA tags may be specified. If two areas intersect, the one which appears first in the map definition takes precedence in the overlapping region.

This tag is new with Netscape V2.0.

Attributes:

Required

coords - gives the coordinates of the shape, using image pixels as the units. For a rectangle, the coordinates are given as "left,top,right,bottom". The rectangular region defined includes the lower-right corner specified, i.e. to specify the the entire area of a 100x100 image, the coordinates would be "0,0,99,99".

Optional

href - specifies where a click in that area should lead. Note that a relative anchor specification will be expanded using the URL of the map description as a base, rather than using the URL of the document from which the map description is referenced. If a BASE tag is present in the document containing the map description, that URL will be used as the base.

nohref - [NOHREF] - indicates that clicks in this region should perform no action.

shape= [RECT | RECTANGLE | CIRC | CIRCLE | POLY | POLYGON] - gives the shape of this area. If the SHAPE tag is omitted, SHAPE="RECT" is the default. (CIRC/CIRCLE takes three coordinates, centerx, centery, and radius; POLY/POLYGON takes three or more pairs of coordinates denoting a polygonal region.)

target - Browser windows can now have names associated with them. Links in any window can refer to another window by name. When you click on the link, the document you asked for will appear in that named window. If the window is not already open, Netscape will open and name a new window for you. New in Netscape 2.0. i.e. `<A HREF="url.html" TARGET="window_name"> Click here and open a New Window </A>`.

Example:

```
<HTML><HEAD>
<TITLE>Client Side Image Maps</TITLE>
</HEAD>
<BODY>
<H1>Client Side Image Maps</H1>
<P>For example, a button bar in a document might use a 160 pixel by 60
pixel image and appear like this: <BR>
<MAP NAME="buttonbar">
<AREA SHAPE="RECT" COORDS="10,10,49,49" HREF="about_us.html">
<AREA SHAPE="RECT" COORDS="60,10,99,49" HREF="products.html">
<AREA SHAPE="RECT" COORDS="110,10,149,49" HREF="index.html">
```

```
<AREA SHAPE="RECT" COORDS="0,0,159,59" NOHREF>
</MAP>
<IMG SRC="../images/tech/bar.gif" USEMAP="#buttonbar"></P>
</BODY><HTML>
```

Typical Rendering (approximate):

Not applicable.

Array

compatibility

<array></array>

Usage:

This tag is used for LaTeX-like arrays. It can only be used within <MATH> elements. It contains one or more <ROW>s, each containing one or more <ITEM>s.

Attributes:

Required

Optional

align - [TOP | MIDDLE | BOTTOM] - By default, arrays are vertically positioned so that preceding and following expressions are aligned with the mid point of the array.

coldef - By default the columns are centered. This attribute can be used to specify the horizontal alignment for each column with character string formed by one capital letter per column, with L for left, C for center and R to right alignment, e.g. "LLCR" for 4 columns.

ldelim - Defaulted to none, this attribute is used to specify the left delimiter, e.g. ldelim="[" for a left square bracket.

rdelim - Defaulted to none, this attribute is used to specify the right delimiter, e.g. rdelim="}" for a right curly brace.

labels [LABELS] - Causes the first row and column to be separated from the rest of the array.

Example:

```
<array ldelim="{" rdelim="}">
<row><item>a<item>b<item>c
<row><item>1<item>2<item>3
</array>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Atop

compatibility

<atop>

Usage:

Use the ATOP tag when you want to place one thing above another, without a dividing line. It is used only within a <BOX> tag (which in turn is within a <MATH> tag).

Attributes:

Required

Optional

Example:

```
{a - b<atop>a + b}
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Author

compatibility

<au></au>

Usage:

This tag indicates the name of an author. This tag is used within 'container' tags such as headings and paragraphs.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
</HEAD>
<BODY>
<H1>Other information about HTML.</H1>
<P>There is lots of information to be had about HTML on the Internet.
One particularly good set of documentation was written by <AU>Dr. Ian
Graham</AU> at the University of Toronto.</P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Auto Tagging A Text Document

The Auto Tag feature helps you to create and tag a new HTML document from an existing text document. It is meant to give you a head start and it is expected that all documents created by this feature will require further editing and tagging. Please note that any angle brackets already existing in a document will be converted to their entity equivalent. This is useful for creating examples of HTML. If you do not want angle brackets to be converted, run the auto-tag **before** adding any tags to the document. As a precaution, If the document already contains an HTML tag, the auto-tag will not work.

Auto Tag is available from the menu by selecting HTML|Document|Auto Tag. Auto Tag creates and marks up a new untitled copy of your document - your original is not touched.

The basic elements of an HTML document are added, namely, the HTML identifier, and the HEAD, TITLE, and BODY tags. A stamp will be placed at the end of the document if Auto-Stamp for converted documents is enabled in Preferences|User Info.

Auto Tag also marks paragraphs and headings based on settings in Preferences|Editor. Paragraphs will be determined by either a hard return or by a blank line.

If your document was created as a text file with hard returns at the end of each line you should choose the 'blank line' option and ensure that there is an extra line between each paragraph. If your document was created in a program that wraps text as you type and there are only hard returns at the end of paragraphs you should choose the 'hard return' option.

Text will be marked as a heading if it is on its own line and is less than the maximum number of characters specified in the preferences. The first case will be tagged as a H1 and subsequent cases will be tagged as H2.

Big Quote

compatibility

<bq></bq>

Usage:

The <BQ> tag is used to quote information from another source. This tag has replaced HTML 2.0's <BLOCKQUOTE> in order to abbreviate the tag name and allow the source of the quote to be credited.

Attributes:

Required

Optional

id - ID - An SGML identifier, NAME token, which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

clear - The CLEAR attribute allows you to move down unconditionally or as long as there is enough room.

nowrap [NOWRAP] - The NOWRAP attribute is used to prevent the browser from automatically wrapping lines.

Example:

```
<HTML><HEAD>
<TITLE>Webber Home Page</TITLE>
</HEAD>
<BODY>
<H1>Welcome to the Webber Home Page</H1>
<BQ>
<P>Webber was designed as a tool for HTML authors who want to see and
control what is happening in their document at the tag level.</P>
<CREDIT>Webber documentation, 1995.</CREDIT>
</BQ>
</BODY></HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we cannot display a typical rendering at this time.

BT

compatibility

<bt></bt>

Usage:

This tag is used, within a <MATH> tag, to change the default font used to render variables and constants. Numbers, operators, delimiters and other symbols are unaffected. <B> renders its contents in bold, while <T> renders its contents in an upright font rather than an italic font. The BT tag combines the two effects into one tag.

Attributes:

Required

Optional

class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<BT>Variable 1</BT>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Background Sound

compatibility

<BGSOUND SRC="sound.wav">

Usage:

The BGSOUND tag allows you to create pages with background sounds or "soundtracks." Sounds can either be samples (.wav or .au format) or MIDI (.mid format).

Attributes:

Required

SRC - Specifies the address of a sound to be displayed.

Optional

LOOP [*n* | INFINITE] - Specifies how many times a sound will loop when activated. If LOOP="-1", or if LOOP="INFINITE" is specified, it will loop indefinitely.

Example:

```
<HTML><HEAD>
<TITLE>Background Sounds in Internet Explorer</TITLE>
</HEAD>
<BODY>
<H1>Background Sounds in Internet Explorer</H1>
<BGSOUND SRC="boing.wav" LOOP="5">
<P>You will hear a boinging noise five times in a row.</P>
</BODY>
</HTML>
```

Banner

compatibility

<banner></banner>

Usage:

This tag is used to define a non-scrolling area at the top of a document to hold corporate logos, disclaimers etc. If it is to be used, the banner tag must appear as the first element in the body. You can also use the <LINK> tag to use an external document as a banner.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Webber Home Page</TITLE>
</HEAD>
<BODY><BANNER>
<H1><IMG SRC="spider.gif">Webber Features</H1>
</BANNER>
<P>Webber has lots of great features.</P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Bar

compatibility

<bar></bar>

Usage:

As an alternative to using the <ABOVE> tag, the BAR tag draws a line above the term enclosed by the element. It is used only within the <MATH> tag.

Attributes:

Required

Optional

Example:

```
<BAR>a + b</BAR>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Base

compatibility

<base>

Usage:

The base tag is used to identify the url of the document itself. The base tag should be used when a document contains urls of other documents which are 'relative' to the current document. In this case other documents will be treated relative to the url mentioned in the base tag. This is useful if the document is moved (i.e. to your local machine.) anchors which use relative urls will continue to work as long as the base tag is used.

Attributes:

Required

href - contains the url of the document

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

N **target** - Browser windows can now have names associated with them. Links in any window can refer to another window by name. Target allows you to pick a default named target window for every link in a document that does not have an explicit TARGET attribute. It's format is: `<BASE TARGET="default_target">` This is new with Netscape 2.0.

Example:

```
<html><head><title>Example</title>
<base href="http://www.csdcorp.com/webber/info.htm">
</head>
<body></body>
</html>
```

Typical Rendering (approximate):

Not applicable.

Basefont

compatibility

<basefont>

Usage:

The <basefont> tag is used to set the size of the BASEFONT that all relative <FONT> changes are based on.

Most new browsers allow the user to customize font sizes for tags on their local machine. In HTML 3.0, font sizing is handled by the <SMALL> and <BIG> tags. When used in combination with their CLASS attribute, it will be possible to assign various levels of font sizes using a style sheet. It seems unlikely at this point that the <basefont> tag will be included in the HTML specification.

Attributes:

Required

size - Defaults to 3, and has a valid range of 1-7.

Optional

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
</HEAD>
<BODY>
<BASEFONT SIZE=4>
<H1>Other information about HTML.</H1>
<P>There is lots of information to be had about <FONT
SIZE=+2>HTML</FONT> on the Internet. </P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Other Information about HTML.

There is lots of information to be had about **HTML** on the Internet.

Below

compatibility

<below></below>

Usage:

This tag is used in <MATH> expressions only. It is used to draw a line, arrow, or curly bracket *below* the expression enclosed by it.

Attributes:

Required

Optional

sym - used to specify the symbol to be used. If this attribute is not used it defaults to "line".

Example:

```
<below sym="line">x+y</below>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Big

compatibility

<big></big>

Usage:

This tag specifies that the enclosed text should be displayed, if practical, using a bigger font than the current font.

Netscape has started supporting this tag with V2.0 of their browser.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
</HEAD>
<BODY>
<H1><BIG>O</BIG>ther information about HTML.</H1>
<P>There is lots of information to be had about HTML on the
Internet.</P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Blink

compatibility

<blink></blink>

Usage:

This tag specifies that the enclosed text should be flashed intermittently.

Attributes:

Required

Optional

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
</HEAD>
<BODY>
<H1>Other information about HTML.</H1>
<P>There is lots of information to be had about HTML on the <BLINK>
World Wide Web.</BLINK></P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Sorry - can't represent flashing text here.

Blockquote

compatibility

<blockquote></blockquote>

Usage:

The blockquote tag is used to quote information from another source. Some browsers render the blockquote element in italics and some will use a left and right indent as well as extra vertical space above and below the tagged text.

This tag is being deprecated in HTML 3.0 and replaced with <BQ> to abbreviate the tag name and allow the source of the quote to be credited.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

3 nowrap [NOWRAP] - Used to prevent the browser from automatically wrapping lines.

Example:

```
<HTML><HEAD>
<TITLE>Webber Home Page</TITLE>
</HEAD>
<BODY>
<H1>Welcome to the Webber Home Page</H1>
<P>The documentation says:
<BLOCKQUOTE>Webber was designed as a tool for HTML authors who want to
see and control what is happening in their document at the tag
level.</BLOCKQUOTE>
You just have to try it now!</P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Welcome to the Webber Home Page

The documentation says:

Webber was designed as a tool for HTML authors who want to see and control what is happening in their document at the tag level.

You just have to try it now!

Body

compatibility

<body></body>

Usage:

The body contains tags which hold the text and graphics of a document. All text within the body must be contained by 'container' tags like headings and paragraphs.

In HTML 3.0 the body tag is optional and need only be used if you want to use one of the attributes of the tag.

J This tag has javascript events associated with it. For details see the description of the Javascript event handlers.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 N E background - This can be used to specify a URI for an image tile to cover the document background. This provides a way of giving a group of documents a distinctive appearance.

N E **bgcolor** - This attribute to BODY is not currently in the proposed HTML 3.0 spec. It allows you to change the color of the background without having to specify a separate image that requires another network access to load. The format is: <BODY BGCOLOR="#rrggbb">Document here</BODY> where "#rrggbb" is a hexadecimal red-green-blue triplet used to specify the background color.

N E **text** - Used to control the color of all the normal text in the document that is not specially colored to indicate a link. The format of TEXT is the same as that of BGCOLOR.

N E **link** - lets you control the coloring of link text. The default coloring is: LINK=blue.

N E **vlink** - lets you control the coloring of a visited link. The default coloring VLINK=purple.

N **alink** - lets you control the coloring of active link. The default coloring of ALINK=red. The format for these attributes is the same as that for BGCOLOR and TEXT. <BODY

```
LINK="#rrggbb" VLINK="#rrggbb" ALINK="#rrggbb">Document here</BODY>
```

3 **bgproperties** [FIXED] - Specifies a watermark, which is a background picture that does not scroll.

E **leftmargin** - Specifies the left margin for the entire body of the page and overrides the default margin. The units are in pixels.

E **topmargin** - Specifies the margin for the top of the page and overrides the default margin. The units are in pixels.

Example:

```
<html>
<head>
<title>This is the title of a very simple document.</title>
</head>
<body background="tileimage.gif"><P>This is the body of the very
simple document.</P></body>
</html>
```

Typical Rendering (approximate):

If it is supported by the browser, the image specified will appear as a background behind the text of the page.

Bold

compatibility

****

Usage:

This tag is used for placing emphasis on a word or phrase. Browsers should render it in bold.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Glossary</TITLE>
</HEAD>
<BODY>
<H1>Glossary</H1>
<P>This is the <B>glossary</B> where you can find definitions for
common phrases.</P>
</BODY><HTML>
```

Typical Rendering (approximate):

Glossary

This is the **glossary** where you can find definitions for common phrases.

Box

compatibility

<box></box>

Usage:

This is a <MATH> tag only. It is used for hidden brackets, stretchy delimiters, and placing one expression over another (e.g. numerators and denominators).

The characters { and } are used as abbreviations for the start and end tags for the box tag. Use the SGML entities { and } respectively when you need to use these characters literally.

Attributes:

Required

Optional

size [NORMAL | MEDIUM | LARGE | HUGE] - Used to obtain oversized delimiters. Normal is the default. The full tag name for the BOX element is needed above in order to specify a value for the SIZE attribute.

Example:

```
<BOX>1<over>x + y</BOX> or {1<over>x + y}
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Bracket Matcher

The bracket matching feature assists you to find mismatched angle brackets in your document. The HTML dtd's specify that missing angle brackets are allowed if a program can determine that one should have existed. Unfortunately many browsers do not handle this feature well and although your document will be considered to be valid HTML, a browser may display it incorrectly. This tool is useful for those times when the validator says it is valid but your browser can't display it properly.

To check for mismatched angle brackets:

1 Select Search|Bracket Matcher from the menu.

2 This operation can take several minutes, especially for long documents. If a mismatched angle bracket is located, the area where the error occurs will be highlighted and the message "mismatched angle bracket!" will be displayed in the status line.

3 If there are no mismatched angle brackets, the message "Brackets matched" will be displayed in the status line.

Calling Webber Through DDE

All of the calls use the DDE Execute Macro interface with the following server and topic. Naturally the server (Webber) has to be running before any of these commands will work.

Server: **Webber**
Topic: **WebberCommand**

All commands are given as a single string argument with the parameters separated by spaces. So in 'C' the Open macro would be given as follows:
`"Open D:\\docs\\index.htm"`
(Note that the double slashes are necessary in 'C' to escape the backslash which is used as an escape character, they are not necessary in all languages).

The macros are not case sensitive and are described as follows:

Macro: **OpenNew**
Description: Causes Webber to open a new blank document.

Macro: **Open filepath**
Description: Causes Webber to open the given file into a document.

Macro: **SaveAs filepath**
Description: Causes Webber to save the current document to the given file.

Macro: **Save**
Description: Causes Webber to save the current document (if it does not yet have a file the user will be prompted).

Macro: **Close**
Description: Causes Webber to close the current document. This does not prompt the user to save unsaved changes so be sure you're ready to close the document before issuing this call.

Macro: **Copy**
Description: Copies the complete contents of the current document to the clipboard.

Macro: **Paste**
Description: Pastes the contents of the clipboard into the current document.

Macro: **Exit**
Description: Causes Webber to terminate.

Caption

compatibility

<caption></caption>

Usage:

This tag is used to label a table or figure. Use the align attribute to specify the position of the caption relative to the table/figure.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 E **align** - [TOP | BOTTOM | LEFT | RIGHT] - Used to position the caption relative to the table or figure it labels.

E **align** - [CENTER] - Used to position the caption relative to the table or figure it labels.

Example:

```
<HTML><HEAD>
<TITLE>Webber (TM)</TITLE>
</HEAD>
<BODY>
<H1>Welcome to the Webber Home Page</H1>
<FIG SRC="spider.gif">
<CAPTION><I>Webber</I> - HTML Editor Extraordinaire!</CAPTION>
<P>Our small arachnid friend tips his hat to you!
<CREDIT>Kathryn M. Dickson</CREDIT> </FIG>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Center

compatibility

** **

Usage:

All lines of text between the begin and end of CENTER are centered between the current left and right margins.

In the recommended dtd of HTML 2.0, it is illegal for anything other than 'container' tags to appear in the body. The validator will require the center tag to be used within one of these other tags. Therefore it will not be legal to put a center tag at the beginning of a document with the end tag at the end of the document in order to center the entire page. You must select 'Netscape' rather than 'Netscape - (R)' as your dtd in order to do this.

In HTML 3.0, centering is handled by alignment attributes on the text container tags such as paragraphs and headings. These attributes offer other alignment alternatives as well. It seems unlikely at this point that the tag will be included in the HTML specification.

Attributes:

Required

Optional

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
</HEAD>
<BODY>
<BASEFONT SIZE=4>
<H1><CENTER>Other information about HTML.</CENTER></H1>
<P>There is lots of information to be had about HTML on the Internet.
</P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Other Information about HTML.

There is lots of information to be had about **HTML** on the Internet.

Check Box Assistant

Check Boxes are input fields that allow the user to 'check' a field by clicking on it with their mouse in a form. The Check Box Assistant is meant to help you place a check box in a form with the needed attributes set. The Assistant is available from the HTML|Control|Check Box... menu and from the Form Assistant.

The attributes usually used with a check box are **name**, **value** and **checked**. The name is returned when the form is submitted as the first in the name/value pair. The value is returned as the second in the name/value pair. The return is made only if the field is checked. i.e. A check box with a name of "Operating System" and a value of "Windows" will return : "Operating System=Windows" but only if the check box is checked. The checked attribute is used to set the default state of the button when the form is loaded.

To Use:

1 Open the Check Box Assistant from the Form Assistant Dialogue by choosing Text Area from the drop-down list or from the menu by selecting HTML|Controls|Check Box... If you are using the latter method be sure to first position your cursor where you want the check box inserted.
2 Enter the name of the check box field. If you have auto-label enabled in Preferences|Forms, the label will fill in with the name value. You can edit this if you want.
3 Enter the value you want returned when the check box is checked.
4 Check the Default to Checked button if you want the check box to be checked when the form is loaded.
5 Edit the label text if you want. If you do not have the auto-label feature enabled, you will not be able to enter into this field.
6 When you are finished, click the Ok button to insert the text field into the form (if you are using the Form Assistant) or your document. Click the Cancel button to leave the Check Box Assistant without inserting the field.

Webber™ Order Form (Cheque)

Please print (File|Print Topic), fill out and mail this form to:

CSD Corp.
499 Rushton Road
Toronto, Ontario Canada M6C 2Y4
+1 416.651.7685 (Fax)

_____
Name
_____
Organization
_____
Street Address
_____ _____
City **Prov./State**
_____
Postal Code/Zip Code
_____
Country

_____ _____
Phone **Fax**
_____ **(Check your selection):**
_____
E-mail Address
Method of Payment
___ **Canadian Cheque** ___ **US Cheque**
___ **Credit Card Cheque** ___ **International Bank Draft in US$**
#_____ Quantity
$_____ Unit Total ($30 US / $40 Can per Unit)
Shipping Method (Check your selection)
[Prices are per unit]:
_____ **Canada - Air ($4 Can)** _____ **Canada - Courier ($15 Can)**
_____ **US - Air ($4 US)** _____ **US - Courier ($20 US)**
_____ **Other - Air ($6 US)** _____ **Other - Courier ($60 US)**
_____ **Compuserve E-mail ($3 US / $4 Can)**
_____ **Internet - MIME ($3 US / $4 Can)** _____ **Internet - UUEncoded ($3 US / $4 Can)**
$_____ Shipping Total

$_____ Sub-Total (Unit Total + Shipping Total)
$_____ 7% GST (Canadian Res. Only)
$_____ 8 % PST (Ontario Res. Only)
$_____ Total

- **Note that we will confirm your order by e-mail only. Be sure to give us your e-mail address.**
- **Please allow sufficient time for handling and delivery. (2-4 weeks by mail, 2-3 days by Courier)**
- **Contact us for information on site licenses at admin@csdcorp.com or fax +1 416.651.7685.**
- **Use this form for orders paid for by cheque only. Credit card orders use the Credit Order Form and fax or mail to PsL, not CSD Corp. Phone credit card orders will be taken by PsL.**

Choose

compatibility

<choose>

Usage:

This tag places one expression above another (like <ATOP>) *and* encloses them in round brackets. This tag is used in the <BOX> tag which is used within the <MATH> tag.

Attributes:

Required

Optional

Example:

```
<{n+1<choose>k}
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Citation

compatibility

<cite></cite>

Usage:

The cite tag is used to indicate a citation such as the title of a book. Usually, it is rendered in italics.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Glossary</TITLE>
</HEAD>
<BODY>
<H1>Glossary</H1>
<HR>
<P>This is the glossary for <CITE>Webber</CITE>.</P>
</BODY></HTML>
```

Typical Rendering (approximate):

Glossary

This is the glossary for *Webber*.

Code

compatibility

<code></code>

Usage:

The code tag is used to indicate a short example of code. It is usually rendered in a monospaced font. It differs from the preformat tag in that it is used in-line rather than for larger blocks of text.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Code Example</TITLE>
</HEAD>
<BODY>
<H1>Code Example</H1>
<HR>
<P>You can set the colour of the text in the document like this:
<CODE>Destination.SetTextColor(RGB(192,192,192));</CODE></P>
</BODY></HTML>
```

Typical Rendering (approximate):

Code Example

You can set the colour of the text in the document like this:

```
Dest.SetTextColor(RGB(192,192,192));
```

Col

compatibility

<col>

Usage:

The COL tag is an optional element is used to specify column based defaults for table properties. It is an empty element, does not have an end tag. Several COL elements may be given in succession.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

style - specifies rendering information about the element

dir [LTR | RTL] - Refers to the direction characters are written. (Left to right, right to left) When applied to TABLE, it indicates the geometric layout of rows (i.e. row 1 is on right if DIR=RTL, but on the left if DIR=LTR) and it indicates a default base direction for any text in the table's content if no other DIR attribute applies to that text.

valign [TOP | MIDDLE | BOTTOM | BASELINE] - Used to specify the vertical alignment of material within a table cell.

align [LEFT | CENTER | RIGHT | JUSTIFY | CHAR] - Used to explicitly specify the horizontal alignment.

char - Used to specify an alignment character for use with align=char, e.g. char=".". The default character is the decimal point for the current language, as set by the LANG attribute. The CHAR attribute value is case sensitive.

charoff - Specifies the offset to the first occurrence of the alignment character on each line. In addition to standard units, the "%" sign may be used to indicate that the value specifies the alignment position as a percentage offset of the current cell, e.g. CHAROFF="30%" indicates the alignment character should be positioned 30% through the cell.

span - A positive integer value that specifies how many columns this element applies to, defaulting to one. In the absence of SPAN attributes the first COL element applies to the first column, the second COL element to the second column and so on. If the second COL element had SPAN=2, it would apply to the second and third column. SPAN=0 has a special significance and implies that the COL element spans all columns from the current column up to and including the last column.

width - Specifies the width of the columns, see standard units. If the element spans several columns then the WIDTH attribute specifies the width for each of the individual columns - not the width of the group. In addition, the "*" suffix denotes relative widths, i.e.

width=64	width in screen pixels
width=0.5*	a relative width of 0.5

Example:

```
<HTML>
```

```
<HEAD><TITLE>Example Table</TITLE>
</HEAD><BODY>
<TABLE BORDER="1" ALIGN="Center" FRAME="HSides" RULES="Rows"
CELLSPACING="10" CELLPADDING="10" COLS="3">
<COL SPAN="2" ALIGN="center" VALIGN="middle"><COL SPAN="1"
ALIGN="char" CHAR="." VALIGN="middle">
<TR><TD></TD><TD></TD><TD></TD></TR>
<TR><TD></TD><TD></TD><TD></TD></TR>
<TR><TD></TD><TD></TD><TD></TD></TR>
<TR><TD></TD><TD></TD><TD></TD></TR>
<TR><TD></TD><TD></TD><TD></TD></TR>
</TABLE>
</BODY></HTML>
```

Colgroup

compatibility

<colgroup></colgroup>

Usage:

Normally, the COL element defines a group of one or more columns. The number of columns in the group is specified by the SPAN attribute. If you also want to specify the width for each of the columns using the WIDTH attribute, then each of the columns must have the same width.

The COLGROUP element allows you to escape this limitation. It acts as a container for one or more COL elements, so you can give each of the columns in the group different widths and alignment properties. COLGROUP suppresses the normal interpretation of COL elements as column groups for any COL elements contained within it.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

style - specifies rendering information about the element

dir [LTR | RTL] - Refers to the direction characters are written. (Left to right, right to left) When applied to TABLE, it indicates the geometric layout of rows (i.e. row 1 is on right if DIR=RTL, but on the left if DIR=LTR) and it indicates a default base direction for any text in the table's content if no other DIR attribute applies to that text.

valign [TOP | MIDDLE | BOTTOM | BASELINE] - Used to specify the vertical alignment of material within a table cell.

align [LEFT | CENTER | RIGHT | JUSTIFY | CHAR] - Used to explicitly specify the horizontal alignment.

char - Used to specify an alignment character for use with align=char, e.g. char=".". The default character is the decimal point for the current language, as set by the LANG attribute. The CHAR attribute value is case sensitive.

charoff - Specifies the offset to the first occurrence of the alignment character on each line. In addition to standard units, the "%" sign may be used to indicate that the value specifies the alignment position as a percentage offset of the current cell, e.g. CHAROFF="30%" indicates the alignment character should be positioned 30% through the cell.

Example:

```
<HTML>
<HEAD><TITLE>Example Table</TITLE>
</HEAD><BODY>
<TABLE BORDER="1" ALIGN="Center" FRAME="HSides" RULES="Rows"
CELLSPACING="10" CELLPADDING="10" COLS="3">
<COLGROUP VALIGN="middle">
```

```
<COL SPAN="2" ALIGN="center">
<COL SPAN="1" ALIGN="char" CHAR=".">
</COLGROUP>
<TR><TD></TD><TD></TD><TD></TD></TR>
<TR><TD></TD><TD></TD><TD></TD></TR>
<TR><TD></TD><TD></TD><TD></TD></TR>
<TR><TD></TD><TD></TD><TD></TD></TR>
<TR><TD></TD><TD></TD><TD></TD></TR>
</TABLE>
</BODY></HTML>
```

Webber™

Getting Started
New in Version 1.4! !! NEW !!
Quick Tips
Welcome to Webber

License Agreements
Registered Version
Evaluation Version

Other Information
To Hide or Not To Hide...
How to Order Webber

Tag Reference
About HTML
Tag Reference
Document Basics

Using Webber
Webber Features & Common Tasks
Webber Menus
A Beginner's Tutorial
Glossary

CSD Corp. has taken a great deal of care to provide you with an accurate and comprehensive help file. If you find that there are errors or omissions in our description of Webber™ or HTML, please let us know at webber@csdcorp.com or visit our Web site at http://www.csdcorp.com/. We would be happy to receive your feedback.

Copying a Table

 The copy button on the Table Assistant and the Table Editor converts the contents of the editor grid, and the size and attributes settings, to HTML. This information is stored on the clipboard where it is available for pasting into the document(s) of your choice.

This gives you the flexibility of pasting the table into a document in order to preview it in the context of the whole document. You can then continue to use the Table Editor to modify the table. When you want to preview your changes, simply copy it once again and paste into your document in place of the previous one.

When you are satisfied with the table, simply close the Table Editor.

Creating a Custom Stamp File

Webber can automatically add a stamp to New HTML and Auto Tagged documents if it is enabled in Preferences|User Info. You can also add the stamp to any document by selecting HTML|Stamp from the menu. Be sure to place your cursor in the position that you want the stamp before selecting the menu item. In this way you can use the stamp as either a signature line or as a banner.

If you want the update date and/or time to be refreshed whenever the document is saved, you must use Webber's stamp feature to create your custom stamp file.

To create a stamp file that can be updated:

1 Create a new blank document by selecting File|New from the menu.
2 Open the Options|Preferences dialog to the User Info page.
3 Check the update date and/or the update time boxes. (You can also use any of the other options here as well - see Author/Last Update Stamp)
4 Ensure that the 'use stamp file' box is **not** checked.
5 Close the dialog by clicking the 'Ok' button.
6 Select HTML|Stamp from the menu.
7 Edit the file using any text and HTML that you want. ***It is very important that you do not change, remove or move any of the comments as this is how Webber finds and updates the date and time. If you are not using the updated date or time flags you may remove the comments if you wish.***
8 Save the file.
9 Open the Options|Preferences dialog to the User Info page.
10 Check the 'use stamp file' box at the bottom of the dialog. Enter the name of your new custom stamp file including the full path.
11 Enable the Auto Stamp for New HTML documents and Auto-tagged documents if you want this feature.
12 Close the dialog by clicking the 'Ok' button.

Creating an Author/Last Update Stamp

You can use Webber's auto-stamp feature to add a stamp (usually used to denote the author, copyright notification, creation date, update date, etc. for a document.) A stamp can be automatically added to New HTML and Auto Tagged documents if it is enabled in Preferences| User Info. You can also add the stamp to any document by selecting HTML|Stamp from the menu. Be sure to place your cursor in the position that you want the stamp before selecting the menu item. The update date and time will be refreshed whenever the document is saved, if you have enabled this option.

The stamp can be your own customized file, or it can contain several pieces of information which can be enabled individually. These include the author's name, e-mail address and copyright notice, as well as the date and/or time created and the date and/or time the file was modified. HTML tags are valid content so that automatic hyperlinks can be created. You can create your own customized file in a way that will enable Webber to update the time and date modified automatically.

To use a file as your stamp:

1 Check the 'use stamp file' box and enter the path to, and the name of your file in the space provided. Customize the file using Webber's features if you want the stamp to be updated automatically.

To use the default stamp:

To use the user info in the preferences dialogue for your stamp, ensure that the 'use stamp file' box is **not** checked.

If you have enabled the Updated Date or Time flags it is very important that you do not change, remove or move any of the comments as this is how Webber finds and updates these items. If you are not using the updated date or time flags you may remove the comments if you wish.

If you use the default stamp, it will look like this in your document:

```
<HR><ADDRESS><!-- Webber Auto-Stamp. Do not remove this comment --
>Page maintained by authorname, <A
HREF="mailto:authorname@company.com">authorname@company.com</A>.
Copyright(c) 1995 <A HREF="http://www.company.com/">companyname</A>.
Created: 05/05/95 at 12:57:42 Updated: <!-- Date Stamp -->05/05/95<!--
--> at <!-- Time Stamp -->12:57:42<!-- --></ADDRESS>
```

where the words in cyan and italics are place-holders and would be taken from Preferences| User Info.

This will be rendered something like this:

Page maintained by authorname, author@company.com. Copyright (c) 1995 companyname.
Created: 05/05/95 at 22:33:05 Updated: 05/05/95 at 22:33:05

Credit

compatibility

<credit></credit>

Usage:

This tag is used to give credit for the source of quotes and figures. It is found within <BQ> and <FIG> tags.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Webber Home Page</TITLE>
</HEAD>
<BODY>
<H1>Welcome to the Webber Home Page</H1>
<BQ>
<P>Webber was designed as a tool for HTML authors who want to see and
control what is happening in their document at the tag level.</P>
<CREDIT>Webber documentation, 1995.</CREDIT>
</BQ>
</BODY></HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Webber™ Order Form (Credit)

(Reference Product #14129)

Please print (File|Print Topic), fill out and fax or mail this form to:

PsL
P.O.Box 35705,
Houston, TX 77235-5705
FAX +1 713.524.6398
or Phone +1.800.242.4775 or +1 713.524.6394

THE ABOVE ADDRESS AND NUMBER ARE FOR CREDIT CARD ORDERS ONLY. THE AUTHOR OF THIS PROGRAM CANNOT BE REACHED AT THESE NUMBERS. FOR SUPPORT, OR GENERAL PRODUCT INFORMATION, E-MAIL Webber@csdcorp.com.

_____
Name
_____
Organization
_____
Street Address
_____ _____
City **Prov./State**
_____
Postal Code/Zip Code
_____
Country

_____ _____
Phone **Fax**
_____ **(Check your selection):**
_____
E-mail Address
Method of Payment
___ **Visa** ___ **Mastercard**
___ **American Express** ___ **Discover**
Credit Card Number : _____
Expiry Date : _____
Name on Card :
_____
Signature :
_____
#_____ Quantity
$_____ Unit Total ($30 US per Unit)
Shipping Method (Check your selection)
[Prices are per unit]:
_____ **Canada - Air ($3 US)** _____ **Canada - Courier ($15 US)**
_____ **US - Air ($4 US)** _____ **US - Courier ($20 US)**
_____ **Other - Air ($6 US)** _____ **Other - Courier ($60 US)**
_____ **Compuserve E-mail ($3 US)**
_____ **Internet - MIME ($3 US)** _____ **Internet - UUEncoded ($3 US)**
$_____ Shipping Total

$_____ Sub-Total (Unit Total + Shipping Total)
$_____ 7% GST (Canadian Res. Only)

$_____ 8 % PST (Ontario Res. Only)

$_____ Total

- **Note that we will confirm your order by e-mail only. Be sure to give us your e-mail address.**
- **Please allow sufficient time for handling and delivery. (2-4 weeks by mail, 2-3 days by Courier)**
- **Contact us for information on site licenses at admin@csdcorp.com or fax +1 416.651.7685.**
- **Use this form for orders paid for by credit card only and mailed or faxed to PsL. For Cheque orders use the Cheque Order Form and mail to CSD Corp., not PsL.**

Custom Bits

"Custom Bits" are user-definable 'bits' of text, tag combinations, entities etc. For example you can define a centered heading (i.e. <H1 ALIGN="center"></H1>) and place it on the tagbar, or assign a shortcut key combination to it. You could also create a custom stamp, or you could add a newly developed tag.

Several example bits have been included with Webber such as the comment tag, <!-- -->, some common entities, and customized tags.

To Use:

1 Select Options|Custom Bits... from the menu.

2 Click the Add button

3 Enter the name of your custom bit as you would like it to appear on the tagbar. Bit names should be 8 characters or less if you plan to assign them to the tagbar. More characters will result in truncation on the button.

4 Enter a description for the custom bit - this will be used for the pop-up hints on the tagbar.

5 Enter the start of the custom bit. This would be the start tag in the case of a customized tag, or the entire entity name, etc. for bits that do not have end tags. You can paste into the edit box in order to add a lot of text or tags.

6 Enter the end of the custom bit (if applicable).

7 If you would like to define a keyboard shortcut for your new bit, place your cursor in the Shortcut edit box. Press the key combination that you would like to use (i.e. ctrl+shift+1). Some key combinations are reserved for Webber or Windows use, if you select a combination that is not available you will get a message telling you so. The check boxes for the alt, shift and ctrl keys are grayed out because you can not modify them directly - they will be set automatically based on the keys you press.

8 Add a new bit by clicking the add button, or click the Ok button to exit and save your custom bits.

9 To assign a custom bit to the tagbar, select Options|Edit Tagbar.

Custom keyboard shortcuts

Shortcut keyboard combinations can be assigned to both custom bits and to tags. The function keys, or the ctrl, shift and alt keys can be used in combination with the alphanumeric keys to create user-definable shortcut key combinations. Some key combinations are reserved for Webber or Windows use, if you select a combination that is not available you will get a message telling you so.

To Use:

1 To assign shortcut keys to custom bits, select Options|Custom Bits... from the menu. To assign shortcuts to tags, select Options|Preferences... from the menu and click the tag tab.

2 Select the custom bit or tag for the shortcut.

3 Place your cursor in the Shortcut edit box at the bottom of the dialog. Press the key combination that you would like to use (i.e. ctrl+shift+1). The check boxes for the alt, shift and ctrl keys are grayed out because you can not modify them directly - they will be set automatically based on the keys you press.

4 Select a new bit or tag to set another shortcut, or click the Ok button to exit and save your changes.

Ddot

compatibility

<ddot></ddot>

Usage:

This tag places a double dot (..) above the term enclosed. It is used in the <MATH> tag as an alternative to <ABOVE>.

Attributes:

Required

Optional

Example:

```
<DDOT>X</DDOT>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Definition

compatibility

<dfn></dfn>

Usage:

This tag indicates the defining instance of a term. It is used inside a 'container' tag such as a paragraph.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Webber Home Page</TITLE>
</HEAD>
<BODY>
<H1>Welcome to the Webber Home Page</H1>
<P>Webber was designed as a tool for HTML <DFN>(Hypertext Mark-up
Language)</DFN> authors who want to see and control what is happening
in their document at the tag level.</P>
</BODY></HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Definition Description

compatibility

<dd></dd>

Usage:

The definition term is used only in definition lists. It contains the definition of the term and is usually indented below it. It can contain multiple lines of text. It follows the <DT> tag which contains the term being defined.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

Example:

```
<HTML><HEAD>
<TITLE>Index</TITLE>
</HEAD>
<BODY>
<H1>Glossary</H1>
<DL>
<DT>Definition List</DT>
<DD>Used to present a list of terms (a single line) and their
definitions (multiple lines).</DD>
<DT>Unordered List</DT>
<DD>Contains multiple lined list items which usually result in left
aligned, bulleted paragraphs. </DD>
</DL>
</BODY><HTML>
```

Typical Rendering (approximate):

Glossary

Definition List

Used to present a list of terms (a single line) and their definitions (multiple lines).

Unordered List

Contains multiple lined list items which usually result in left aligned, bulleted paragraphs.

Definition List

compatibility

<dl></dl>

Usage:

HTML supports several types of lists which may be nested within themselves or each other. The definition list (sometimes referred to as the glossary list) differs from the other types of lists in that it has two item elements, the term and the definition. It is used to present a list of terms (a single line) and their definitions (multiple lines).

In HTML 3.0 an optional list header, <LH> tag, has been added. If used the list header comes first and is followed by a definition term.

Attributes:

Required

Optional

compact - [COMPACT] - suggests that the list be rendered with a minimum of white space, not all browsers make use of this attribute.

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

Example:

```
<HTML><HEAD>
<TITLE>Index</TITLE>
</HEAD>
<BODY>
<H1>Glossary</H1>
<DL>
<DT>Definition List</DT>
<DD>Used to present a list of terms (a single line) and their
definitions (multiple lines).</DD>
<DT>Unordered List</DT>
<DD>Contains multiple lined list items which usually result in left
aligned, bulleted paragraphs. </DD>
</DL>
</BODY><HTML>
```

Typical Rendering (approximate):

Glossary

Definition List
: Used to present a list of terms (a single line) and their definitions (multiple lines).

Unordered List
: Contains multiple lined list items which usually result in left aligned, bulleted paragraphs.

Deleted

compatibility

<del></del>

Usage:

This tag is used to mark deleted text, for instance in legal documents. It is used within a 'container' tag such as a paragraph.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
</HEAD>
<BODY>
<H1>Other information about HTML.</H1>
<P>There is lots of information to be had about HTML on the
<DEL>Internet</DEL> <INS>World Wide Web</INS>.</P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Dictionaries

Dictionaries must be installed in the \windows\system directory. They can be used alone or in concert with others. Dictionaries available are: US English (standard), UK English, French, German and Italian.

For the latest information on other language dictionaries see the web pages at csdcorp.com. (http://www.csdcorp.com/). Optional dictionaries will be available for download at this location.

To select dictionaries:

1 Click the 'Dictionaries' button on the spell checker.

2 Dictionaries being used are shown on the left (at least one dictionary must be open). The american english dictionary is used by default. Available dictionaries are shown on the right. If there are no dictionaries shown on the right, there are not any installed in your windows system directory. You may need to download a dictionary from our Web site.

3 To open an additional dictionary, select the dictionary from the list on the right and click the < button to add it to the list of open dictionaries on the left. Use the << to open all available dictionaries.

4 To close an open dictionary, select the dictionary from the list on the left and click the > button to remove it from the list of open dictionaries on the left. Use the >> to remove all open dictionaries. You must have at least one dictionary open at all times so you will need to reopen one if you use the remove all option.

5 Close the dialogue by clicking the 'Close' button when you are finished.

Directory Listing

compatibility

<dir></dir>

Usage:

HTML supports several types of lists which may be nested within themselves or each other. The directory listing is similar (with some browsers identical) to the menu list and unordered list. It contains list items which usually result in left aligned, bulleted columns. Some browsers will try to optimize and place multiple columns beside each other in 24 character wide slots. Therefore it is best to keep the content of the list items less than 24 characters.

This tag has been deprecated in HTML 3.0 and will be obsolete in the future. It is recommended that you use an unordered list instead.

Attributes:

Required

Optional

compact - [COMPACT] - suggests that the list be rendered with a minimum of white space, not all browsers make use of this attribute.

Example:

```
<HTML><HEAD>
<TITLE>Index</TITLE>
</HEAD>
<BODY>
<H1>Index</H1>
<DIR>
<LI>Ch. 1</LI>
<LI>Ch. 2</LI>
<LI>Ch. 3</LI>
<LI>Ch. 4</LI>
</DIR>
</BODY><HTML>
```

Typical Rendering (approximate):

Index

- Ch. 1
- Ch. 2
- Ch. 3
- Ch. 4

or

Index

Division

compatibility

** **

Usage:

This tag is used to divide the content of a document into hierarchical containers. It is located within the <BODY> tag and can be used with the CLASS attribute to represent different kinds of container, e.g. chapter, section, abstract, appendix.

Netscape has started supporting the align attribute of this tag with V2.0 of their browser.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

align [LEFT | CENTER | RIGHT] - Used to specify the horizontal alignment of paragraphs within the division.

nowrap [NOWRAP] - Used to prevent the browser from automatically wrapping lines.

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
</HEAD>
<BODY>
<DIV CLASS="SECTION.ABSTRACT"><H1>Other information about HTML.</H1>
<P>There is lots of information to be had about
<ACRONYM>HTML</ACRONYM> on the Internet.</P></DIV>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Not applicable as a rendering other than the alignment of paragraphs if specified.

Document Basics

The HTML language is often misunderstood. As a sub-set of SGML, HTML was not intended to be a desk-top publishing or layout format. Its purpose is to mark-up a document based on its content. This means that most of the HTML tags denote the structure of the content such as headings, paragraphs, lists and images. The actual presentation of the content is determined by individual browsers. While there is some character level formatting possible for emphasis, HTML is quite limited in what it can do for the layout of a document. The extensions being made to HTML by browser vendors, allow a document author more control over the look of the document but only for the portion of Web users who are using the latest version of their browsers. Keep in mind that if you use the extensions or elements from the draft HTML 3.0 spec., not everyone looking at your pages will see them in all their glory.

If you are authoring pages for others it is important to make them aware of some of the limitations of HTML *before* they spend a lot of time planning how they want their pages to look. Here are a few things for you and them to keep in mind.

- vertical and horizontal spacing is determined by the HTML, not by spaces, tabs, carriage returns and extra lines that you add to the raw document. In fact, the browsers will ignore all of these things unless they are located within a preformat tag.
- tables are becoming more commonly supported. To align columns of information it is also possible to use the preformat tag. This tag allows you to align columns using spaces and displays the contents of the tag in a monospace font. To use the tag accurately you will need to change the font of the document to a fixed font.
- it is currently not possible to create pages with newspaper-like columns except by using a table as the underlying structure for the page.
- all text and images will be left-aligned unless you create the page using extensions or HTML 3.0 tags ***and*** the person looking at the document is using a browser that can handle these new elements.
- only one line of text can be placed beside images - when the text wraps it will wrap below an image, not around it. Again there are extensions and HTML 3.0 tags which allow text to flow around images but you need a browser that supports these tags in order to view it.
- while you can achieve many layout effects like centering and right-alignment using extensions and/or HTML 3.0 elements, people browsing the Web require a browser that supports these features in order to see your page as you meant it to look. In fact, the use of these less commonly supported features may cause unpredictable results with some browsers. Carefully consider who your intended audience is before creating pages that might look terrible or blow up for some users!
- even if you usually use a browser that can handle these new features it is a good idea to have one that can't for the purposes of testing your pages to see what they might look like.

Document Head

compatibility

<head></head>

Usage:

The head contains tags which identify information about a document such as it's title, usage etc. The head must contain a TITLE but all other head elements (META, BASE, ISINDEX, NEXTID, LINK) are optional.

Attributes:

Required

Optional

Example:

```
<html>
<head>
<title>The title of a very simple document.</title>
</head>
<body><p>This is the text of the very simple document.<p></body>
</html>
```

Typical Rendering (approximate):

Not applicable.

Dot

compatibility

<dot></dot>

Usage:

This tag places a single dot (.) above the term enclosed. It is used in the <MATH> tag as an alternative to <ABOVE>.

Attributes:

Required

Optional

Example:

```
<DOT>X</DOT>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Drag & Drop Text

Webber supports drag and drop for text and tags within a document. You can turn support for drag and drop on (or off) in Preferences|Editor. Drag and drop allows you to select a piece of text (this can include a tag or part of a tag) and drag it to another location.

To use:

1 Select the text you want to move.
2 Click on the selected text.
3 Hold the right mouse button down and move the cursor to the destination.
4 Release the mouse button.

Options:

Pressing the Ctrl key while performing the above action will copy the selected text rather than moving it.

Edit Menu

Edit|Undo
Multiple level undo - reverses changes made to tags and text in reverse order. 'Undo' does one action per click. Large operations such as pasting over a selection of text are undone in one action.
Keyboard Equivalent: Alt + E, U
Shortcut: Ctrl Z
Button?: Yes
Right Click?: Yes

Edit|Redo
Multiple level redo - redoes whatever changes have just been undone. 'Redo' does one action per click.
Keyboard Equivalent: Alt + E, R
Shortcut: None
Button?: No
Right Click?: No

Edit|Cut
Cuts the selected text and tags to the clipboard.
Keyboard Equivalent: Alt + E, T
Shortcut: Ctrl X
Button?: Yes
Right Click?: Yes

Edit|Copy
Copies the selected text and tags to the clipboard.
Keyboard Equivalent: Alt + E, C
Shortcut: Ctrl C
Button?: Yes
Right Click?: Yes

Edit|Paste
Pastes the selected text and tags from the clipboard.
Keyboard Equivalent: Alt + E, P
Shortcut: Ctrl V
Button?: Yes
Right Click?: Yes

Edit|Paste From...
Opens the file open dialog and allow the user to select a text file. Pastes the contents of the selected text file into the current document at the cursor position.
Keyboard Equivalent: Alt + E, F
Shortcut: None
Button?: No
Right Click?: No

Edit|Select All
Selects all tags and text in the current document.

Keyboard Equivalent: Alt + E, S
Shortcut: Ctrl A
Button?: No
Right Click?: No

Edit|Tag Select
Selects the text and tag from the beginning of a start tag to the end of its end tag. **Note: this feature does not work if a tag does not have an end tag. The selection will be made to the end of the document instead.**
Keyboard Equivalent: Alt + E, G
Shortcut: None
Button?: No
Right Click?: Yes

Edit|Last Tag
Repeats the most recent tag inserted by the drop-down list.
Keyboard Equivalent: Alt + E, L
Shortcut: Ctrl + L
Button?: No
Right Click?: No

Editing the Floating Tagbar

The tagbar is a customizable, floating/dockable tool bar on which you can place any number of tags on any number of tabbed 'pages'. Pausing your mouse pointer over the tag buttons will display the colloquial name for the tag, or, in the case of custom bits - the description that was defined for the bit. You can display/hide the tagbar by selecting Options|Show Tagbar from the menu.

To Use:

1 Select Options|Edit Tagbar from the menu.
2 Add tags to a tab by selecting the desired tab, selecting the desired tag and clicking the > button. Multiple tags can be selected by pressing the ctrl key and clicking on the desired tags. The >> button will add all tags to the same tab.
3 Remove tags from a tab by selecting the desired tab, selecting the desired tag and clicking the < button. Multiple tags can be selected by pressing the ctrl key and clicking on the desired tags. The << button will remove all tags from a tab.
4 Add new tabs by clicking the Add... button at the top right of the dialog. Enter the name of the new tab and click Ok. ***Note:*** *giving tabs short names will ensure that the maximum number of tabs is able to show at one time.*
5 Edit the name of a tab by selecting the desired tab, and clicking the Edit... button at the top right of the dialog. Enter the new name and click Ok.
6 Delete a tab by selecting the tab and clicking the Delete button at the top right of the dialog.
7 Add custom bits to the tagbar by selecting the 'custom bits' radio button. All custom bits defined will appear on the list and can be added to any tab on the tagbar in the same way that tags are.
8 To accept the tagbar setup, click the Ok button.

Embed

compatibility

<embed>

Usage:

The EMBED tag allows you to put documents directly into an HTML page. The EMBED tag will allow you to embed documents of any type. Your user only needs to have an application which can view the data installed correctly on their machine.

If a width and height are specified, the embedded object is scaled to fit the available space.

Embedded objects can be activated by double clicking them in the Netscape window.

Attributes:

Required

src - Specifies the source URL.

Optional

height - The object's image will be scaled to fit the specified height and width.

width - The object's image will be scaled to fit the specified height and width.

Example:

```
<HTML><HEAD>
<TITLE>Test Embed Document</TITLE></HEAD>
<body><P><EMBED SRC="images/embed.bmp"></P>
</BODY></HTML>
```

Emphasis

compatibility

****

Usage:

This tag is used for placing emphasis on a word or phrase. Browsers often render it in italics.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Glossary</TITLE>
</HEAD>
<BODY>
<H1>Glossary</H1>
<P>This is the <EM>glossary</EM> where you can find definitions for
common phrases.</P>
</BODY><HTML>
```

Typical Rendering (approximate):

Glossary

This is the *glossary* where you can find definitions for common phrases.

Entity Assistant

The Entity Assistant allows you to visually select the character that you want to use and insert the correct entity for the character. (Not all characters can be displayed in standard windows fonts - these characters are represented by a box and can be selected from the drop-down list.) The Entity Assistant can also automatically convert special characters for you when you click the 'auto' button. This does not work for angle brackets, < >, and works for ampersands, &, only if it is separated from other letters by a space. (i.e. Tom & Jane will be converted while Tom&Jane would not be.)

Entities are names used to represent either single characters or strings of characters. In HTML the most common use of entities is to represent characters that are not consistently supported across all viewing platforms. Entities are designated by the ampersand (&) and the 'name' of the character followed by a semi-colon. Entities should always be used in place of special characters such as the angle brackets, < (`<`), > (`>`) and the ampersand, & (`&`) They should also be used in place of any characters for which you have to use special key combinations on your keyboard to achieve, such as letters with accents, Ç (`Ç`), é (`é`), ê (`ê`), etc.

To use:

1 Place your cursor in the position that you want the entity inserted.
2 Open the Entity Assistant from the HTML menu.
3 If the character entity you require is available in a standard windows font it will be represented in the grid by the character. If it is not it will be represented by a blank box.
4 Select the entity you require by clicking on the character in the grid or by selecting the entity name from the drop-down list at the bottom of the dialog.
5 Click the 'Insert' button to insert the entity into your document. The entity assistant is a floating dialog that enables you to insert an entity, reposition your cursor and insert more entities.
6 When finished, click the Close button.

Example

compatibility

<xmp></xmp>

Usage:

The example and listing tags are obsolete and have been replaced by the preformat tag. They allow text to be displayed with spacing etc. exactly as it was typed. Browsers will usually use a monospaced font to render the content of these tags.

This tag has been deprecated in HTML 2.0 and is already obsolete in HTML 3.0. It is recommended that you use the preformat tag instead.

Attributes:

Required

Optional

Example:

```
<HTML><HEAD>
<TITLE>Webber Reference</TITLE>
</HEAD>
<BODY>
<H1>Webber Reference</H1>
<HR>
<XMP>A example tag retains any
spacing      and
carriage returns that you type. You should, however
use the preformat tag instead.</XMP>
</BODY></HTML>
```

Typical Rendering (approximate):

Webber Reference

```
An example tag retains any
spacing      and
carriage returns that you type. You should, however
use the preformat tag instead.
```

E When used in front of an attribute, this symbol means that the attribute is used for the current tag as a Internet Explorer Extension. It is supported by Internet Explorer 2.0 and later. If it appears in conjunction with the HTML 3.0 symbol it means that the attribute has been incorporated into the HTML 3.0 dtd.

Internet Explorer extensions are not fully supported by all browsers. Check your browser documentation or test your document with your browser to see if the attribute you are using is supported. Also keep in mind that there are many browsers on the market and significant numbers of your audience may be using browsers which don't support this feature.

Explorer Ext.
Please note that while Explorer extensions are supported by Webber and our validation feature, they are not yet part of the HTML specification. While it is possible that many or all of these extensions may be adopted as part of the standard, there is no guarantee that this will happen. If you use these extensions you should be aware that they may undergo changes. Documents using these extensions are not yet officially compliant and may never be.

Explorer Extensions

The extensions documented here are based on documentation on the Microsoft web site for their Internet Explorer browser. For the complete list of tags supported by Explorer, see the Quick Tag Reference. Starting with V2.0 of Internet Explorer, you can configure your system so that a new menu item appears on the 'Edit' menu of Explorer. The menu item is 'Current Page' and when selected will open the source for the current document into your editor. To configure Webber as your editor of choice, please see this document on DDE.

Tags:
AREA (Area for Client Side Image Map)
BGSOUND (Background Sound)
MARQUEE (Marquee)
MAP (Client Side Image Map)
OBJECT (ActiveX Object)
SCRIPT (Script)

Attributes:
BODY (Body)
CAPTION (Caption)
FONT (Font)
IMG (Image)
TABLE (Table)

Plus Support for Some Netscape Extensions:
APPLET (Java Applet)
BASEFONT (Basefont)
CENTER (Center)
FRAME (Frames)
FRAMESET (Frameset)
HR (Horizontal Rule)
ISINDEX (Document is a searchable index)
LI (List Item)
META (Meta-Information)
NOBR (No Break)
NOFRAME (Noframe)
OL (Ordered List)
STRIKE (Strike Through)
WBR (Word Break)

If this symbol appears alone it tells you that this tag is a ***Internet Explorer Extension***. Not all attributes of this tag may apply - check the attribute list for this tag for more information.

If this symbol appears with an HTML 3 symbol (3), it means that Microsoft has added support for this HTML 3 element to their browser. If it appears with a Netscape Extension symbol (N), it means that Microsoft has added support for this Netscape tag to their browser.

Internet Explorer Extensions are not fully supported by all browsers. Check your browser documentation or test your document with your browser to see if the tag you are using is supported. Keep in mind that if you use unsupported features there may be a significant number of people viewing your document who use other browsers which do not support these extensions.

Figure

compatibility

<fig></fig>

Usage:

Figures are used for large graphical images whereas the <IMG> tag is meant for small graphics such as icons. The figure tag allows for a text description of the figure for use by non-graphical browsers. The figure tag contains optional <OVERLAY>s followed by an optional <CAPTION>, then the text description, and finally an optional <CREDIT>.

Attributes:

Required

src - the url of the image.

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

md - Used to specify a message digest or cryptographic checksum for the associated graphic.

align [BLEEDLEFT | LEFT | CENTER | RIGHT | BLEEDRIGHT | JUSTIFY] - Used to specify the horizontal alignment of the figure. The default is CENTER.

noflow [NOFLOW] - The presence of this attribute disables text flow around the figure.

width - Used to specify the desired width in pixels or en units (according to the value of the UNITS attribute). User agents may scale the figure image to match this width.

height - Used to specify the desired height in pixels or en units (according to the value of the UNITS attribute). User agents may scale the figure image to match this height.

units [EN | PIXELS] - Specifies the choice of units for width and height. Pixels are the default.

imagemap - Used to specify a URI for processing image clicks and drags.

Example:

```
<HTML><HEAD>
<TITLE>Webber (TM)</TITLE>
</HEAD>
<BODY>
<H1>Welcome to the Webber Home Page</H1>
<FIG SRC="spider.gif">
<CAPTION><I>Webber</I> - HTML Editor Extraordinaire!</CAPTION>
<P>Our small arachnid friend tips his hat to you!
<CREDIT>Kathryn M. Dickson</CREDIT> </FIG>
</BODY>
</HTML>
```

Example of Using Anchors to define Hotzones:

```
<FIG SRC="menu.gif">
<H1>Welcome to Webber's Home</H1>
```

```
<P>Select between:
<UL>
<LI><A HREF="intro.htm" SHAPE="rect 30,200,60,16">Introduction</A>
<LI><A HREF="relnotes.htm" SHAPE="rect 100,200,50,16">Release
Notes</A>
<LI><A HREF="techsupt.htm" SHAPE="rect 160,200,30,16">Tech Support</A>
<LI><A HREF="news.htm" SHAPE="rect 200,200,50,16">News Releases</A>
</UL>
</FIG>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

File Drag & Drop

Webber can open files by drag and drop. One file or many files can be opened by dropping them onto Webber in its open state or minimized state. If Webber is not running (as on a launch bar) and files are dropped onto it, Webber will launch first and then open the dropped files.

File Menu

File|New
Opens a new untitled document.
Keyboard Equivalent: Alt + F, N
Shortcut: Ctrl N
Button?: Yes

File|New HTML
Opens a new untitled document and inserts the basic elements required for a valid HTML document. If you have set the preference to do so, the auto-stamp will also be inserted. See Creating a new HTML document.
Keyboard Equivalent: Alt + F, H
Shortcut: Ctrl H
Button?: Right-click on New button

File|Open...
Opens a dialogue box allowing you to select a file (or multiple files) to open. Multiple files can be selected with standard Windows behaviour - i.e. holding down the ctrl key while clicking on multiple file names in sequence. The files you have opened most recently are available from the drop-down list in the dialogue.
Keyboard Equivalent: Alt + F, O
Shortcut: Ctrl O
Button?: Yes

File|Open Template...
Opens a dialogue box allowing you to select a predefined template to open. Create new templates by saving any HTML file to the /tmplts subdirectory in your Webber program directory. The file will be opened as an untitled document to prevent it from being accidentally overwritten when you save your document.
Keyboard Equivalent: Alt + F, T
Shortcut: No
Button?: Right-click on open button

File|Save
Saves the active file (i.e. the file whose window has the focus) to the name and directory indicated in the title bar of its window. If the file does not have a name yet, you will be prompted to enter one.
Keyboard Equivalent: Alt + F, S
Shortcut: Ctrl S
Button?: Yes

File|Save As...
Saves the active file to the name and directory of your choice. You will be prompted for this information with the File|Save dialogue box.
Keyboard Equivalent: Alt + F, A
Shortcut: None
Button?: No

File|Close

Closes the active file.
Keyboard Equivalent: Alt + F, C
Shortcut: Ctrl P
Button?: No

File|Print
Prints the active file to the default printer. Beginning with version 1.2 the font used for printing will be the one you are using on the screen (or the printer equivalent of the screen font).
Keyboard Equivalent: Alt + F, P
Shortcut: Ctrl P
Button?: No

File|Print Setup...
Opens the printer setup dialogue box which allows you to change the page orientation, printer, etc.
Keyboard Equivalent: Alt + F, R
Shortcut: None
Button?: No

File|Exit
Exits the application after prompting you to save any unsaved work.
Keyboard Equivalent: Alt + F , X
Shortcut: None
Button?: No

File|Quick Open List
The four most recently opened files are stored on this list so that you can reopen them quickly. Simply click on the file that you want to open.
Keyboard Equivalent: None
Shortcut: None
Button?: No

Font

compatibility

****

Usage:

The font tag is used to set the size of the font. Valid values range from 1-7. The default FONT size is 3. The value given to size can optionally have a '+' or '-' character in front of it to specify that it is relative to the <BASEFONT>.

In HTML 3.0, font sizing is handled by the <SMALL> and <BIG> tags. When used in combination with their CLASS attribute, various levels of font sizes will be possible. It seems unlikely at this point that the <font> tag will be included in the HTML specification. Netscape has added support for these tags with V2.0 of their browser.

Attributes:

Required

N E **size** - Defaults to 3, and has a valid range of 1-7.

Optional

N E **color** - sets the color of the surrounded text.

E **face** - Sets the font. A list of font names can be specified. If the first font is available on the system, it will be used, otherwise the second will be tried, and so on. If none are available, a default font will be used. i.e. <FONT FACE="Arial,Lucida Sans,Times Roman"> This text will be in either Arial, Lucida Sans, or Times Roman, depending on which fonts you have installed on your system.</FONT>

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
</HEAD>
<BODY>
<BASEFONT SIZE=4>
<H1>Other information about HTML.</H1>
<P>There is lots of information to be had about <FONT SIZE=+2
COLOR="#00FF00">HTML</FONT> on the Internet. </P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Other Information about HTML.

There is lots of information to be had about **HTML** on the Internet.

Font Selection

Fonts can be set only for a document and are not currently available at the character level. (This feature will be implemented in a future version of Webber.)

Your preferred font can be set as a preference. This font will be used for all documents when they are opened.

To change the font of a specific document select Options|Font from the menu. This method of font selection will only affect the active window until it is closed.

Footnote

compatibility

<fn></fn>

Usage:

This tag identifies a footnote. When given an ID, an anchor can be used from within the document to link to the footnote. When supported by browsers, the footnote will rendered as a pop-up note. When the anchor is clicked on within the text of the document.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Hamlet - the Play</TITLE>
</HEAD>
<BODY>
<H1>Quotes from Hamlet</H1>
<P>Hamlet: <Q>Get thee to a nunnery. Why wouldst thou be a breeder of
sinners? I am myself <a href="#fn2">indifferent honest</a> ...
</Q></P>
<FN id=fn3><P><i>indifferent honest</i> - moderately virtuous</P></FN>
</BODY></HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Form

compatibility

<form></form>

Usage:

A form is a template for collecting information from a user of the Internet. The form element can contain input, selection, and text area, tags, along with document structuring elements. Forms elements can be mixed in with document structuring elements. For example, a <PRE> element may contain a <FORM> element, or a <FORM> element may contain lists which contain input elements. This gives considerable flexibility in designing the layout of forms.

J This tag has javascript events associated with it. For details see the description of the Javascript event handlers.

More: See Form Assistant.

Attributes:

Required

action - specifies the action URI for the form. This attribute defaults to the base URI of the document

Optional

method - [GET | POST] - specifies the method of accessing the action URI.

enctype - This attribute specifies the MIME content type to be used to encode the form contents. It defaults to the string: "application/x-www-form-urlencoded".

3 **script -** used to give a URI for a script. The scripting language and the interface with the user agent is not part of the HTML 3.0 specification.

N **target** - Browser windows can now have names associated with them. Links in any window can refer to another window by name. When you click on the link, the document you asked for will appear in that named window. If the window is not already open, Netscape will open and name a new window for you. New in Netscape 2.0. i.e. `<A HREF="url.html" TARGET="window_name"> Click here and open a New Window </A>`.

Example:

```
<HTML>
<HEAD>
<TITLE></TITLE>
</HEAD>
<BODY><FORM ACTION="http://www.somewhere.com/cgi-bin/mailscript"
METHOD="POST">
</FORM></BODY>
</HTML>
```

Typical Rendering (approximate):

Not applicable on it's own.

Form Assistant

Forms are probably the most complicated tags to place into a document. There are many tags with many attributes. The Form Assistant was designed to help you easily and quickly create and preview forms.

The Form Assistant, (as well as all other Assistants), is a 'one-way' tool. Once you have added the form to your document you must edit the tags directly to make changes to them. If your browser type supports it (see Preferences|HTML), Webber will create and load a copy of your form for viewing with your default browser when you click the preview button. In this way you can make changes and corrections to your form before inserting it into your document.

The Form Assistant is actually a series of assistants which work together to create a form. The Form Field Assistants which are opened as you select field types for your form, are also available individually from the HTML menu for use with existing forms.

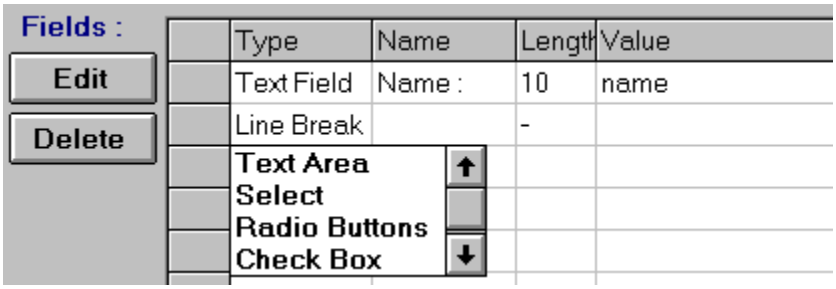
The Form Assistant allows you to select field types, add labels and insert line breaks. Optionally, you can enable the Auto-Label feature in Preferences|Forms. The Auto-Label feature uses the values you specify for your fields as labels as well as values. This eliminates the need to type in labels manually.

To Use:

1 Place your cursor in the position that you want the form to be inserted.

2 Open the Form Assistant by selecting HTML|Form from the menu.

3 Fill in the action attribute by select the url type from the drop-down list and filling in the url in the space provided. Also select the method attribute from the drop-down list. Forms require server scripts in order to know what to do with the information entered into them. If you are uncertain what values to give the action, method and encoding type attributes, check with your service provider or web server administrator for assistance.



4 Double-click on an empty cell in the Type column for the fields. Select the field type, line break or label from the drop-down list.

If you select a line break, no further action is required since it does not have any attributes.

If you select a label, a dialogue asking you for the text of the label will appear. A label is not an HTML tag - it is optional content that appears in your form to identify fields to the user.

You can have the Form Assistant generate the labels for you by enabling the Auto-Label feature in the Preferences|Forms. The position of the labels can also be set there.

If you select a field type, one of the form field assistants will pop up. The available field types and buttons in a form are:

Text Edit Fields
Radio Buttons
Check Boxes
Selection Lists
Multi-line Text Areas
Reset Buttons
Submit Buttons

5 When you have selected the fields you want, you can preview the result by clicking the test button. This will load your browser and display your form.

6 You can change the order of the fields by clicking on the first column of the field type table and dragging a row up or down with your mouse. You can edit any of the fields by clicking on them and clicking the edit button. Fields can be removed by clicking the delete button after clicking on the field to be removed.

7 When you are happy with the result, click the Ok button to insert the form into your document. If you want to abandon the form click the Cancel button to leave the Form Assistant without inserting the form.

Form Controls

The Form Field Assistants are used to insert fields into a form. These Form Field Assistants are called from the Form Assistant when building a form. They are also available from the HTML menu if you want to add a control to an existing form.

The Assistants were designed to help you place form fields in a form without having to remember which attributes are used. The Assistants prompt you for the important information required to make the fields work. Text labels can be generated for the fields automatically if Auto-Label is enabled in Preferences|Forms. The position of the labels is also set there. You can disable this feature for each field type individually if you want.

Text Field Assistant
Radio Button Assistant
Check Box Assistant
Selection List Assistant
Multi-line Text Area Assistant
Reset Button Assistant
Submit Button Assistant

Form Input

compatibility

<input>

Usage:

The input tag represents a form field for user input. These can appear in a wide variety of types - see the attribute list below.

(See also: Form Assistant and Form Field Assistants)

File uploads are supported with Netscape 2.0. (TYPE=FILE) - see below.

J This tag can have javascript events associated with it, but only with certain input types are set. For details see the description of the Javascript event handlers.

Attributes:

Required

Optional

type - [TEXT | PASSWORD | CHECKBOX | RADIO | SUBMIT | RESET | IMAGE | HIDDEN] - indicates type of the field. Defaults to 'TEXT'. (For new HTML 3.0 types see below.)

name - name for the form field corresponding to this element or group of elements. Returned when the form is submitted as the first in the name/value pair. Required for all but submit and reset types.

value - The initial value of the field. Required for radio buttons & checkboxes.

src - A URI specifying an image resource. For use only with 'TYPE=IMAGE', (or 'TYPE=SCRIBBLE', 'TYPE=RESET', and 'TYPE=SUBMIT' in HTML 3.0)

checked - [CHECKED] - indicates that the initial state of a checkbox or radio button is selected. For use only with TYPE=CHECKBOX or TYPE=RADIO.

size - specifies the amount of display space allocated to this input field according to its type.

maxlength - constrains the number of characters that can be entered into a text input field. If the value of MAXLENGTH is greater the the value of the SIZE attribute, the field should scroll appropriately. The default number of characters is unlimited.

align - [TOP | MIDDLE | BOTTOM] - vertical alignment of the image. For use only with 'TYPE=IMAGE'. (See new alignment option with HTML 3.0 below.)

New Attributes (HTML 3.0 only) :

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 N **type** - [RANGE | FILE | SCRIBBLE] - Netscape V2.0 also supports FILE.

- **align** - [LEFT | RIGHT] - With ALIGN=LEFT, the graphic will float down and over to the current left margin, and subsequent text will wrap around the right hand side of the graphic. Likewise for ALIGN=RIGHT, the graphic aligns with the current right margin and, and text wraps around the left.
- md - Used to specify a message digest or cryptographic checksum for an associated graphic (for use with TYPE=IMAGE only).
- **disabled** [DISABLED] - Will result in the field being rendered as usual except that it can't be modified by users. It may 'look' disabled i.e. be grayed out. This is for read-only fields.
- **error** - This attribute specifies an error message explaining why the field's current value is incorrect.
- **min** - Specifies the lower limit for RANGE fields.
- **max** - Specifies the upper limit for RANGE fields.
- **accept** - Specifies a list of acceptable MIME types for file fields.

Example:

```
<HTML>
<HEAD>
<TITLE>Example Form</TITLE>
</HEAD>
<BODY><FORM ACTION="http://www.somewhere.com/cgi-bin/mailscript"
METHOD="POST">
<P><BR>My password is: <INPUT TYPE="PASSWORD" SIZE="10">
<BR>Types of computers that I use:<BR><INPUT TYPE="CHECKBOX"
NAME="Windows" CHECKED="CHECKED"> a PC with Windows
<BR><INPUT TYPE="CHECKBOX" NAME="Mac"> a Mac
<BR><INPUT TYPE="CHECKBOX" NAME="Unix"> Unix
<BR>I prefer to communicate by:
<BR>E-mail <INPUT TYPE="RADIO" NAME="Reply to" VALUE="E-mail"
CHECKED="CHECKED">
Snail Mail <INPUT TYPE="RADIO" NAME="Reply to" VALUE="Snail Mail">
<BR>Fax <INPUT TYPE="RADIO" NAME="Reply to" VALUE="Fax">
Phone <INPUT TYPE="RADIO" NAME="Reply to" VALUE="Phone">
<BR><INPUT TYPE="SUBMIT" VALUE="Send info" NAME="Send info"><INPUT
TYPE="RESET" VALUE="Clear Form"></P></FORM></BODY></HTML>
```

Typical Rendering (approximate):

Tell us about yourself:
My e-mail address is: webber@csdcorp.com
My password is: ********
Types of computers that I use:
☒ a PC with Windows
☐ a Mac
☐ Unix
I prefer to communicate by:
E-mail ◉ Snail Mail ○
Fax ○ Phone ○
Send info | Clear Form

Frame Assistant

The Frame Assistant allows you to visually layout frames, and assign an HTML file to them, as well as setting other basic attributes. You can build both simple and complex, nested frames very easily.

To Use:

1 Position your cursor where you would like the frame layout to be placed.

2 Select HTML|Frames... from the menu.

3 The tool bar in the Frame Assistant has three buttons. Clicking the first divides the active area vertically. Clicking the second divides the active area horizontally. The third button will delete the active frame. A frame is active if the background is white.

4 To add a frame, click one of the buttons. Set the size of the frame by dragging the frame borders, or use the size entry box. For vertical frames the size field displays the width of the frames, while with horizontal frames it displays the height.

5 The frame assistant will automatically give the frames identifiers which you can modify if you choose. The SRC attribute is required and frames will not display in your browser if you have not specified a file as the SRC.

6 You can also set other common attributes such as no user-resizing, no scroll bars, and no border (Internet Explorer only).

7 When you have completed laying out your frames, click the Ok button to insert the set of frame tags. The frame assistant will insert the FRAMESET, FRAME, and NOFRAME tags. The NOFRAME tag is meant to hold a BODY tag etc. that should be displayed if the browser does not support frames.

i.e.

```
<HTML><HEAD>
<TITLE>Frame Example</TITLE>
</HEAD>
<FRAMESET Cols="10%,90%">
<FRAME NAME="Frame1" SRC="one.htm" FRAMEBORDER="No">
<FRAMESET Rows="33%,67%">
<FRAME NAME="Frame2" SRC="two.htm" NORESIZE>
<FRAME NAME="Frame3" SRC="three.htm" SCROLLING="No">
</FRAMESET>
```

```
<NOFRAME><BODY>
<P>If you are seeing this message, your browser does not support frames.
Please see our <A HREF="alt.htm">alternative pages</A>.</P>
</BODY>
</NOFRAME></FRAMESET>
</HTML>
```

Frames

compatibility

<frame>

Usage:

A Frame Document has a basic structure very much like your normal HTML document, except the BODY container is replaced by a FRAMESET container which describes the sub-HTML documents, or Frames, that will make up the page.

This tag defines a single frame in a frameset. The FRAME tag is not a container so it has no matching end tag.

You can easily create frames by using the Frame Assistant.

Attributes:

Required

Optional

src - Takes as its value the URL of the document to be displayed in this particular frame. If there is no SRC attribute, the frame will be displayed as a blank space the size the frame would have been.

name - Used to assign a name to a frame so it can be targeted by links in other documents (These are usually from other frames in the same document.) By default all windows are unnamed. Names must begin with an alphanumeric character. Named frames can have their window contents targeted with the TARGET attribute.

marginwidth - Used when the document author wants some control of the margins for this frame. If specified, the value for MARGINWIDTH is in pixels. Margins can not be less than one - so that frame objects will not touch frame edges - and can not be specified so that there is no space for the document contents. By default, all frames default to letting the browser decide on an appropriate margin width.

marginheight Just like MARGINWIDTH above, except it controls the upper and lower margins instead of the left and right margins.

scrolling [yes | no | auto] - Used to describe if the frame should have a scrollbar or not. Yes results in scrollbars always being visible on that frame. No results in scrollbars never being visible. Auto instructs the browser to decide whether scrollbars are needed, and place them where necessary. Default value is auto.

noresize - Has no value. It is a flag that indicates that the frame is not resizable by the user. Users typically resize frames by dragging a frame edge to a new position. Note that if any frame adjacent to an edge is not resizable, that entire edge will be restricted from moving. This will effect the resizability of other frames. By default all frames are resizable.

Example:

```
<HTML><HEAD><TITLE>Test Frame Document</TITLE></HEAD>
<FRAMESET COLS="50%,50%">
  <FRAMESET ROWS="50%,50%">
    <FRAME SRC="cell.html">
    <FRAME SRC="cell.html">
  </FRAMESET>
```

```
  <FRAMESET ROWS="33%,33%,33%">
    <FRAME SRC="cell.html">
    <FRAME SRC="cell.html">
    <FRAME SRC="cell.html">
  </FRAMESET>
</FRAMESET></HTML>
```

Frameset

compatibility

<frameset></frameset>

Usage:

Frame syntax is similar in scope and complexity to that used by tables, and has been designed to be quickly processed by Internet client layout engines.
The FRAMESET is the main container for a Frame. A frame document has no BODY, and no tags that would normally be placed in the BODY can appear before the FRAMESET tag, or the FRAMESET will be ignored. The FRAMESET tag has a matching end tag, and within the FRAMESET you can only have other nested FRAMESET tags, FRAME tags, or the NOFRAMES tag.

The FRAMESET tag can be nested inside other FRAMESET tags. In this case the complete subframe is placed in the space that would be used for the corresponding frame if this had been a FRAME tag instead of a nested FRAMESET.

You can easily create frames by using the Frame Assistant.

J This tag has javascript events associated with it. For details see the description of the Javascript event handlers.

Attributes:

Required

Optional

rows - Takes as its value a comma separated list of values. These values can be absolute pixel values, percentage values between 1 and 100, or relative scaling values. The number of rows is implicit in the number of elements in the list. Since the total height of all the rows must equal the height of the window, row heights might be normalized to achieve this. A missing ROWS attribute is interpreted as a single row arbitrarily sized to fit.

Syntax of value list.:

n - A simple numeric value is assumed to be a fixed size in pixels. This is the most dangerous type of value to use since the size of the viewer's window can and does vary substantially. If fixed pixel values are used, it will almost certainly be necessary to mix them with one or more of the relative size values described below. Otherwise the client engine will likely override your specified pixel value to ensure that the total proportions of the frame are 100% of the width and height of the user's window.

n% - This is a simple percentage value between 1 and 100. If the total is greater than 100 all percentages are scaled down. If the total is less than 100, and relative-sized frames exist, extra space will be given to them. If there are no relative-sized frames, all percentages will be scaled up to match a total of 100%.

n* - The value on this field is optional. A single '*' character is a "relative-sized" frame and is interpreted as a request to give the frame all remaining space. If there exist multiple relative-sized frames, the remaining space is divided evenly among them. If there is a value in front of the '*', that frame gets that much more relative space. "2*,*" would give 2/3 of the space to the first frame, and 1/3 to the second.

Example for 3 rows, the first and the last being smaller than the center row: <FRAMESET ROWS="20%,60%,20%">

Example for 3 rows, the first and the last being fixed height, with the remaining space assigned to the middle row: <FRAMESET ROWS="100,*,100">

cols - takes as its value a comma separated list of values that is of the same syntax as the list described above for the ROWS attribute.

Example:

```
<HTML>
<HEAD></HEAD>
<FRAMESET ROWS="20%,60%,20%">
</FRAMESET>
</HTML>
```

Getting Help on a Tag

Webber offers extensive help on HTML tags and their attributes.

You can get context sensitive help on any tag by placing your cursor between the left and right angle brackets of the tag name, right-clicking and selecting Tag Help from the pop-up menu. Tag Help is also available from the Help menu and from a button on the Tag Assistant. You can also find out what tags are allowed within the context of your document and your location within the document by using the Tags in Context button in the Tag Assistant.

HTML Tags used in Version 2.0, and Version 3.0 (draft) of the HTML specification , as well as Netscape and Internet Explorer Extensions, are covered by this help file. Each tag topic shows the common usage, the attributes, and an example where possible.

dtd Quick Reference (2.0 + 3.0)
Alphabetical Tag List (2.0)
Tags By Usage (2.0)
Tags By Usage (3.0)
Netscape Extensions
Internet Explorer Extensions

Glossary

T

- **tag**
- **text input**
- **typed**

U

- **URI**
- **url**

HTML 3 Table Model

The information presented here has been distilled from the W3C Working Draft of October, 27 '95. The original draft of the HTML 3 specification was posted in March 1995. Since then work has continued on various aspects of HTML 3, with this table model being one aspect.

The table model was changed to try to accomplish several goals:

- to allow alignment of cell contents on characters such as a decimal
- to give more flexibility in specifying table frames
- to enable the browsers to incrementally display large tables as information is received from the server
- to support scrollable tables with fixed headers, plus better support for breaking tables across pages for printing
- to allow for optional column based defaults for alignment properties
- to provide backwards compatibility with Netscape tables
- to simplify the import of tables based on the CALS table model

In order to meet these goals, several attributes have been added to the TABLE element and a few removed. As well, there have been several new elements introduced including: COLGROUP, COL, THEAD, TFOOT, and TBODY.

HTML 3.0

Please note that while the draft version of HTML 3.0 is supported by Webber and our validation feature, it has not yet been fully adopted. While we continue to update our support of the proposed specification as it changes, if you use this support you should be aware that it may undergo changes. Documents using this support are not yet officially compliant and may never be.

HTML 3.2

In the spring of 1996 HTML 3.2 was introduced by the W3 organization working in conjunction with other industry players such as Microsoft and Netscape. HTML 3.2 is meant to encompass the tags that are commonly used by the major browsers. It goes a long way toward supporting many of the Microsoft and Netscape extensions but there are still some extensions supported by specific browsers that are not included in HTML 3.2 (such as frames). For the complete list of tags in HTML 3.2 see the Quick Tag Reference.

a
address
applet
area
b
base
big
blockquote
body
br
caption
center
cite
code
dd
dfn
dir
div
dl
dt
em
font
form
h1
h2
h3
h4
h5
h6
head
hr
html
i
img
input
isindex
kbd
li
link
listing
map
menu
meta
ol
option

p
param
plaintext
pre
samp
script
select
small
strike
strong
style
sub
sup
table
td
textarea
textflow
th
title
tr
tt
ul
var
xmp

HTML Identifier

compatibility

<html></html>

Usage:

The HTML Identifier identifies that the document contains HTML elements, it is required for all HTML documents. It usually contains only the <head> and <body> elements. A third element <plaintext> is optional in HTML 2.0 but rarely used. The plain text element is obsolete in HTML 3.0.

The HTML 3.0 dtd specifies that the HTML tag, the HEAD tag and the BODY tag are all optional. The smallest valid HTML 3.0 document contains only a title.

Attributes:

Required

Optional

version - Used to help avoid future compatibility problems, gives the version number of the specification to which the document conforms. This is fixed by the HTML 3.0 DTD as the string "-//W3O//DTD W3 HTML 3.0//EN"

3 urn - Specifies the universal resource name for the document.

3 role - An optional space separated list of SGML NAME tokens that define the role this document plays, e.g. table of contents.

Example:

```
<html>
<head>
<title>This is the title of a very simple document.</title>
</head>
<body><p>This is the text of the very simple document.</p></body>
</html>
```

Typical Rendering (approximate):

Not applicable.

HTML Menu

HTML|Tag...
Opens the Tag Assistant dialogue box which shows a list of all tags plus their attributes.
Keyboard Equivalent: Alt + T, T
Button?: Yes
Right Click?: No

HTML|Entity...
Opens the Entity Assistant dialogue box which allows you to select an entity and insert into the current document.
Keyboard Equivalent: Alt + T, E
Button?: Yes
Right Click?: No

HTML|Anchor...
Opens the Anchor Assistant dialogue box which allows you to quickly add an anchor to your document and set the common attributes.
Keyboard Equivalent: Alt + T, A
Button?: Yes
Right Click?: No

HTML|Image...
Opens the Image Assistant dialogue box which allows you to quickly add a graphic to your document and to set its common attributes.
Keyboard Equivalent: Alt + T, I
Button?: Yes
Right Click?: No

HTML|List...
Opens the List Assistant dialogue box which allows you to quickly add an unordered, ordered or definition list, or just multiple list items. Specify the type of list, the number of list items, enter a list header (HTML 3 only) and click Ok.
Keyboard Equivalent: Alt + T, L
Button?: Yes
Right Click?: No

HTML|Form...
Opens the Form Assistant dialogue box which allows you to layout and preview a form and then insert it into your document.
Keyboard Equivalent: Alt + T, F
Button?: No
Right Click?: No

HTML|Table
Opens the Table Assistant dialogue box which allows you to specify the number of rows and columns, and set other table attributes before copying your table to the clipboard so that you can paste it into you document. You can also use the Table Editor to edit and layout your table, and join cells to create row and column spanning.
Keyboard Equivalent: Alt + T, B

Button?: Yes
Right Click?: No
Shortcut: Ctrl + T

HTML|Insert -->
Allows you to insert the header (docinfo) or a stamp (or a custom stamp file) into the current document at the cursor position. The header info will be inserted into New HTML documents automatically if this option is enabled in the preferences.

The information included in the default stamp is specified in Preferences|User Info.. If auto-stamp is checked then this information will be added to New HTML documents and to Auto Tagged documents. The date and time updated will be refreshed automatically whenever a document containing them is saved.
Keyboard Equivalent: Alt + T, I +D for docinfo, +S for stamp.
Button?: No
Right Click?: No

HTML|Controls -->
Shows a drop-down list of form controls which can be added to your document independently of the form tag. This option allows you to add fields to an existing form.
Keyboard Equivalent: Alt + T, C
Button?: No
Right Click?: No

HTML|Document -->
Shows a drop-down list of actions which can be taken on the current document. These include: Previewing the document with your browser; Validating the document's HTML; Auto-tagging a plain text document; and Stripping the HTML tags from an HTML document.
Keyboard Equivalent: Alt + T, D
Button?: Preview & Validate - Yes, Auto-tag, Strip Tags - No
Right Click?: Validate - Yes

HTML Tag Reference

2
3
E
N

The HTML specification is in an ever changing state. The popularity of the World Wide Web is keeping the standards-making groups and browser and editor vendors on our toes. Currently most browsers on the market support HTML 2.0 very well. If you want to be sure that you are creating valid HTML documents you should use only the HTML 2.0 support of Webber. You can select the specification you want to use by right-clicking on the dtd button on the status line. Most of these tags are also included in the HTML 3.0 dtd and will be marked with the symbols shown above.

2 HTML 2.0 Tags (Alphabetical)
2 HTML 2.0 Tags (By Usage)

A ³ (superscript 3) following a tag name following a tag name on a drop-down list, indicates that the tag is a HTML 3.0 tag. Two stars (**) following a tag name mean that it is an extension tag by Netscape or Microsoft. Tags available in HTML 2.0 do not have any special indicators. To quickly compare HTML 2.0 to HTML 3.0, use the following topic.

dtd Quick Reference

The HTML 3.0 specification is still in its draft phase. It continues to change monthly. The information in this help file is based on the draft dated March 28, 1995 with the exception of the TABLE tags which are based on the draft table specification of October 27, 1995, and the insert tag from the W3C working draft specification of December 21, 1995. We will continue to monitor the progress of the drafts and will make changes to the help file with new versions of Webber. We will also try post important revisions on our Web site at: http//www.csdcorp.com/ in a timely manner.

Many of the tags and attributes that are new to the HTML 3.0 dtd are not yet widely supported with the exception of the table tag and alignment attributes on headings and paragraphs. If you choose to use any of these features you should be aware that the spec will change and tags you use may not be included in the final version. With that said, there are some tags and attributes that are already supported by browsers - these features will probably continue being supported but there is no guarantee. Most new browsers already support tables and alignment attributes on images, paragraphs, headings etc.

3 HTML 3.0 Tags (By Usage)

In the spring of 1996 HTML 3.2 was introduced by the W3 organization working in conjunction with other industry players such as Microsoft and Netscape. HTML 3.2 is meant to encompass the tags that are commonly used by the major browsers. It goes a long way toward supporting many of the Microsoft and Netscape extensions but there are still some extensions supported by specific browsers that are not included in HTML 3.2 (such as frames).

HTML 3.2 Tags (Alphabetical)

Netscape Extensions are additions made to the HTML spec by Netscape Communications. While some of their work has been incorporated into the HTML 3.0 dtd, some has not. Where possible we have tried to indicate the status of these extensions with regards to the HTML 3.0 dtd. Even if these tags and attributes are not included in the official HTML spec., they will be

supported by Netscape and some other browsers. However, you should keep in mind that there is probably a significant portion of the Web surfing audience who will be using browsers that do not support some or all of these features. You need to decide whether you want 100% compatibility.

Netscape Extensions

Microsoft's Internet Explorer extensions are additions made to the HTML spec by Microsoft. Although they are supported by Webber and our validation feature, they are not yet part of the HTML specification. As far as we know they are supported only by the Microsoft Browser which also supports many of the Netscape Extensions. While it is possible that many or all of these extensions may be adopted as part of the standard, there is no guarantee that this will happen. If you use these extensions you should be aware that they may undergo changes. Documents using these extensions are not yet officially compliant and may never be.

Internet Explorer Extensions

HTML Tag Reference Index

A ³ (superscript 3) following a tag name on the drop-down list in the tag assistant, indicates that the tag is a HTML 3.0 tag. Two stars (**) following a tag name mean that it is an extension tag by Netscape or Microsoft. Tags available in HTML 2.0 do not have any special indicators.

Don't know which DTD to use? Here is some more detail on what they are and what state they are in. Not all tags mentioned in this documentation are currently supported by browsers. This help file contains information on all tags contained in the specifications or introduced by other parties. Although each tag has a 'compatibility' chart associated with it, it is possible that some of the information may become outdated - the web changes very quickly. We recommend that you try the tags in several of the latest browsers to determine the level of support available.

2
3
N
3 - When these symbols are used next to attributes they have special meanings, click on each to find out more.

When positioned in front of an **attribute**, this symbol tells you that the attribute is applicable to the current tag *only* in the ***HTML 2.0 dtd***, not the HTML 3.0 dtd.

This symbols tells you that this tag is found in the ***HTML 2.0 dtd***. Not all attributes may apply to this dtd - check the attribute list for this tag for more information.

HTML 2.0 is the only dtd that is fully supported by most browsers at this time.

When positioned in front of an **attribute**, this symbol tells you that the attribute is applicable to the current tag only in the ***HTML 3.0 dtd***, not the HTML 2.0 dtd. This symbol will be used for attributes of tags that are found in both dtd's. It is considered redundant for those tags which are found only in HTML 3.0.

The HTML 3.0 dtd is not yet fully supported by most browsers. Some elements of this dtd, such as tables, have been adopted by new browsers. Check your browser documentation or test your document with your browser to see if the tag you are using is supported.

The HTML 3.0 tag descriptions in this help file are based on the Internet *draft* dated March 28, 1995. Note that this is only a draft dtd and changes to it are expected. Changes to the draft that affect this help file will be posted on our web site at http://www.csdcorp.com/.

When positioned beneath the tag name, this symbol tells you that the tag is found in the ***HTML 3.0 dtd***. Not all attributes listed for this tag may apply to this dtd - check the attribute list for this tag for more information.

The HTML 3.0 dtd is not yet fully supported by most browsers. Some elements of this dtd, such as tables, have been adopted by new browsers. Check your browser documentation or test your document with your browser to see if the tag you are using is supported.

The HTML 3.0 tag descriptions in this help file are based on the Internet *draft* dated March 28, 1995. Note that this is only a draft dtd and changes to it are expected. Changes to the draft that affect this help file will be posted on our web site at http://www.csdcorp.com/.

Hat

compatibility

<hat></hat>

Usage:

This tag places a 'hat' accent (^) above the term enclosed. It is used in the <MATH> tag as an alternative to <ABOVE>.

Attributes:

Required

Optional

Example:

```
<HAT>X</HAT>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Header (META) information

Header information (META tags) can be generated automatically based on specified settings and/or several predefined 'macros' - perfect for expiry dates, author name, version number, keywords etc. For example:

```
<META HTTP-EQUIV="Expires" CONTENT="Mon, 03 Jun 1996 10:47:53
GMT-05:00">
<META HTTP-EQUIV="Keywords" CONTENT="Webber V 1.4, CSD Corp., HTML
editor">
<META NAME="Generator" CONTENT="Webber V1.4">
<META NAME="Copyright" CONTENT="1996 Cerebral Systems Development
Corp.">
```

To Configure:

1 Select Options|Header Info... from the menu.

2 You can use either the HTTP-EQUIV attribute, or the NAME attribute to identify the type of META tag.

3 To use HTTP-EQUIV, select from the list of options in that column. Meta tags which have an HTTP-EQUIV in them are used by either the http server or client. For example, Netscape Navigator and Internet Explorer use 'refresh' to specify a 'client pull' (see the meta tag for details). Some servers and clients will use 'expires' to determine whether a document should be cached or loaded from the server. Be sure that you have set up your time zone correctly in the preferences if you are going to use a date.

4 To use NAME, type the name into the space provided.

5 The CONTENT attribute is required. Type the appropriate value into the space provided, or use the drop-down list of macros.

6 Click the add button to add a new meta tag, or click the Ok button to save the changes and exit the dialog.

To Use:

1 You can specify that you would like the header information to be included in all New HTML documents by enabling the option on the HTML tab of the Preferences dialog.

2 You can also insert the header information on demand by selecting HTML|Insert|Docinfo from the menu.

Heading - Level 1

compatibility

<h1></h1>

Usage:

Headings are used to identify the beginning of a new section within a document. There are six levels of headings. Browsers usually render a Level 1 heading in a large bold font and decrease the size of the font with each level of heading.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

3 align [LEFT | CENTER | RIGHT | JUSTIFY] - Used to specify the horizontal alignment of the heading. The default is left.

3 seqnum - This attribute is used to set the sequence number associated with the header level of the current element to a given number, e.g. SEQNUM=10.

3 skip - Increments the sequence number before rendering the element. It is used when headers have been left out of the sequence.

3 dingbat - Used to specify an iconic image to appear preceding the header.

3 src - Used to specify an image to appear preceding the header.

3 md - Specifies a message digest used when you want to be sure that a linked graphic, specified by the SRC attribute, is the same one intended, and hasn't been changed in any way.

3 nowrap [NOWRAP] - Used to prevent the browser from automatically wrapping lines.

Example:

```
<HTML><HEAD>
<TITLE>Webber Home Page</TITLE>
</HEAD>
<BODY>
```

```
<H1 ALIGN="CENTER">Welcome to the Webber Home Page</H1>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Welcome to the Webber Home Page

Heading - Level 2

compatibility

<h2></h2>

Usage:

Headings are used to identify the beginning of a new section within a document. There are six levels of headings. Browsers usually render a Level 1 heading in a large bold font and decrease the size of the font with each level of heading.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

3 align [LEFT | CENTER | RIGHT | JUSTIFY] - Used to specify the horizontal alignment of the heading. The default is left.

3 seqnum - This attribute is used to set the sequence number associated with the header level of the current element to a given number, e.g. SEQNUM=10.

3 skip - Increments the sequence number before rendering the element. It is used when headers have been left out of the sequence.

3 dingbat - Used to specify an iconic image to appear preceding the header.

3 src - Used to specify an image to appear preceding the header.

3 md - Specifies a message digest used when you want to be sure that a linked graphic, specified by the SRC attribute, is the same one intended, and hasn't been changed in any way.

3 nowrap [NOWRAP] - Used to prevent the browser from automatically wrapping lines.

Example:

```
<HTML><HEAD>
<TITLE>Webber Home Page</TITLE>
</HEAD>
<BODY>
```

```
<H2 ALIGN="RIGHT">Welcome to the Webber Home Page</H2>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Welcome to the Webber Home Page

Heading - Level 3

compatibility

<h3></h3>

Usage:

Headings are used to identify the beginning of a new section within a document. There are six levels of headings. Browsers usually render a Level 1 heading in a large bold font and decrease the size of the font with each level of heading.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

3 align [LEFT | CENTER | RIGHT | JUSTIFY] - Used to specify the horizontal alignment of the heading. The default is left.

3 seqnum - This attribute is used to set the sequence number associated with the header level of the current element to a given number, e.g. SEQNUM=10.

3 skip - Increments the sequence number before rendering the element. It is used when headers have been left out of the sequence.

3 dingbat - Used to specify an iconic image to appear preceding the header.

3 src - Used to specify an image to appear preceding the header.

3 md - Specifies a message digest used when you want to be sure that a linked graphic, specified by the SRC attribute, is the same one intended, and hasn't been changed in any way.

3 nowrap [NOWRAP] - Used to prevent the browser from automatically wrapping lines.

Example:

```
<HTML><HEAD>
<TITLE>Webber Home Page</TITLE>
</HEAD>
<BODY>
```

```
<H3 ALIGN="CENTER" NOWRAP="NOWRAP">Welcome to the Webber Home
Page</H3>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Welcome to the Webber Home Page

Heading - Level 4

compatibility

<h4></h4>

Usage:

Headings are used to identify the beginning of a new section within a document. There are six levels of headings. Browsers usually render a Level 1 heading in a large bold font and decrease the size of the font with each level of heading.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

3 align [LEFT | CENTER | RIGHT | JUSTIFY] - Used to specify the horizontal alignment of the heading. The default is left.

3 seqnum - This attribute is used to set the sequence number associated with the header level of the current element to a given number, e.g. SEQNUM=10.

3 skip - Increments the sequence number before rendering the element. It is used when headers have been left out of the sequence.

3 dingbat - Used to specify an iconic image to appear preceding the header.

3 src - Used to specify an image to appear preceding the header.

3 md - Specifies a message digest used when you want to be sure that a linked graphic, specified by the SRC attribute, is the same one intended, and hasn't been changed in any way.

3 nowrap [NOWRAP] - Used to prevent the browser from automatically wrapping lines.

Example:

```
<HTML><HEAD>
<TITLE>Webber Home Page</TITLE>
</HEAD>
<BODY>
```

```
<H4 SRC="bullet.gif">Welcome to the Webber Home Page</H4>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Welcome to the Webber Home Page

Heading - Level 5

compatibility

<h5></h5>

Usage:

Headings are used to identify the beginning of a new section within a document. There are six levels of headings. Browsers usually render a Level 1 heading in a large bold font and decrease the size of the font with each level of heading.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

3 align [LEFT | CENTER | RIGHT | JUSTIFY] - Used to specify the horizontal alignment of the heading. The default is left.

3 seqnum - This attribute is used to set the sequence number associated with the header level of the current element to a given number, e.g. SEQNUM=10.

3 skip - Increments the sequence number before rendering the element. It is used when headers have been left out of the sequence.

3 dingbat - Used to specify an iconic image to appear preceding the header.

3 src - Used to specify an image to appear preceding the header.

3 md - Specifies a message digest used when you want to be sure that a linked graphic, specified by the SRC attribute, is the same one intended, and hasn't been changed in any way.

3 nowrap [NOWRAP] - Used to prevent the browser from automatically wrapping lines.

Example:

```
<HTML><HEAD>
<TITLE>Webber Home Page</TITLE>
</HEAD>
<BODY>
```

```
<H5>Welcome to the Webber Home Page</H5>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Welcome to the Webber Home Page

Heading - Level 6

compatibility

<h6></h6>

Usage:

Headings are used to identify the beginning of a new section within a document. There are six levels of headings. Browsers usually render a Level 1 heading in a large bold font and decrease the size of the font with each level of heading.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

3 align [LEFT | CENTER | RIGHT | JUSTIFY] - Used to specify the horizontal alignment of the heading. The default is left.

3 seqnum - This attribute is used to set the sequence number associated with the header level of the current element to a given number, e.g. SEQNUM=10.

3 skip - Increments the sequence number before rendering the element. It is used when headers have been left out of the sequence.

3 dingbat - Used to specify an iconic image to appear preceding the header.

3 src - Used to specify an image to appear preceding the header.

3 md - Specifies a message digest used when you want to be sure that a linked graphic, specified by the SRC attribute, is the same one intended, and hasn't been changed in any way.

3 nowrap [NOWRAP] - Used to prevent the browser from automatically wrapping lines.

Example:

```
<HTML><HEAD>
<TITLE>Webber Home Page</TITLE>
</HEAD>
<BODY>
```

```
<H6>Welcome to the Webber Home Page</H6>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Welcome to the Webber Home Page

Help Menu

Help|Contents
Opens this help file at its contents.
Keyboard Equivalent: Alt + H, C
Shortcut: F1
Button?: Yes
Right Click?: No

Help|Search for Help On...
Opens the search dialogue box for this help file.
Keyboard Equivalent: Alt + H, S
Shortcut: None
Button?: No
Right Click?: No

Help|Tag Help
Opens this help file at the help topic for a selected tag. Tags may be selected by double-clicking on them.
Keyboard Equivalent: Alt + H, T
Shortcut: None
Button?: No
Right Click?: Yes

Help|Beginner's Tutorial
Opens this help file at a simple tutorial for creating a basic home page and a basic form using Webber.
Keyboard Equivalent: Alt + H, U
Shortcut: None
Button?: No
Right Click?: No

Help|How to Use Help
Opens the Windows *Help on Help* file.
Keyboard Equivalent: Alt + H, H
Shortcut: None
Button?: No
Right Click?: No

Help|About...
Opens the *About Webber* information box.
Keyboard Equivalent: Alt + H, A
Shortcut: None
Button?: No
Right Click?: No

Hints

Webber provides help for menu options and buttons in the form of hints. Menu item hints appear in the status line as you highlight an item with your mouse. Buttons hints appear below the button in bubbles when you pause your mouse pointer over the button. No clicking is necessary.

Horizontal Rule

compatibility

<hr>

Usage:

The horizontal rule tag tells the browser to place a horizontal line in the document. The appearance of the horizontal line will vary depending on the browser and the colours used. It is an empty tag.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

3 src - Used to specify a custom image for the rule.

3 md - Specifies a message digest used when you want to be sure that a linked graphic, specified by the SRC attribute, is the same one intended, and hasn't been changed in any way.

N E **size** - Used to specify the thickness of the horizontal rule.

N E **width** - Used to specify the width of the rule in pixels or as a percent of the page width. The default width is that of the page.

N E **align** [LEFT | RIGHT | CENTER] - Used to specify the alignment of the rule when it is less than the width of the page.

N E **noshade** [NOSHADE] - Used to disable any fancy (3D) rendering of the line.

Example:

```
<HTML><HEAD>
<TITLE>Glossary</TITLE>
</HEAD>
<BODY>
<H1>Glossary</H1>
```

```
<HR>
<P>This is the glossary for Webber.</P>
<HR>
<ADDRESS>Copyright 1995, Cerebral Systems Development
Corp.</ADDRESS></BODY></HTML>
```

Typical Rendering (approximate):

Glossary

This is the glossary for Webber.

Copyright 1995, Cerebral Systems Development Corp.

How To Use Webber

ID

An SGML identifier - NAME token - which must be unique within the scope of the current document. The ID attribute can be used as the target for hypertext links or for naming particular elements in associated style sheets. This replaces the HTML 2.0 Anchor / Name method of specifying a target. (<A NAME="target">Target</A>)

IETF
Internet Engineering Task Force - these are the folks who are working hard at coming up with a specification for HTML that meets as many peoples' requirements as possible.

Image

compatibility

<img>

Usage:

The image tag is an empty tag. It references an image by it's URI and browsers display the image where the image tag is located. Currently most browsers support only GIF (.gif) images as in-line images although some of the more popular browsers also now support JPEG (.jpg) images in-line. Other formats can be used within anchor tags and will often be displayed by programs external to the viewer.

For large figures with captions and text flow see <FIG>.

Internet Explorer 2.0 enables you to embed .avi (Audio Video Interleave) video clips in HTML pages. This is done by adding several new attributes, notably DYNSRC (Dynamic Source), to the IMG tag. Non-video-enabled browsers display still images in their place.

(See also: Image Assistant and Quick-Tag Buttons)

Attributes:

Required

src - the url of the image

Optional

alt - alternate text for display if the browser does not support graphics

align=[TOP | MIDDLE | BOTTOM] - specifies whether the text beside the image should align with the top, middle or bottom of the image. (For text flow alternatives with HTML 3.0, see below).

ismap - [ISMAP] - used to identify images which are clickable imagemaps

New Attributes (HTML 3.0 only):

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 md - Used to specify a message digest or cryptographic checksum for the associated graphic.

3 E **align** [LEFT | RIGHT] - With ALIGN=LEFT, the graphic will float down and over to the current left margin, and subsequent text will wrap around the right hand side of the graphic. Likewise for ALIGN=RIGHT, the graphic aligns with the current right margin and, and text wraps around the left. (see below for more Netscape options)

3 N E **width** - Used to specify the desired width in pixels or en units (according to the value of the UNITS attribute). Browsers may scale the image to match this width.

3 N E **height** - Used to specify the desired height in pixels or en units (according to the value of the UNITS attribute). Browsers may scale the image to match this height.

3 units [EN | PIXELS] - Specifies the choice of units for width and height. Pixels are the default.

New Attributes (Netscape & Explorer Extensions only):

N align - [TOP | MIDDLE | BOTTOM | LEFT | RIGHT | TEXTTOP | ABSMIDDLE | BASELINE | ABSBOTTOM]

N E **border** - Used to control the thickness of the border around an image displayed. Warning: setting BORDER=0 on images that are also part of anchors may confuse your users as they are used to a colored border indicating an image is an anchor.

N E **vspace** - Used to control the vertical space above and below a floating image.

N E **hspace** - Used to control the horizontal space to the left and right of the image.

N E **usemap** - indicates that it is a client-side image map. The different regions of the image are described using a MAP element. This is a new feature of Netscape V2.0.

E **controls** - If a video clip is present, a set of controls is displayed under the clip.

E **dynsrc** - Specifies the address of a video clip or VRML world to be displayed in the window. Stands for Dynamic Source. Can be used in combination with the SRC attribute. If a browser does not support DYNSRC the still image defined by SRC will be used instead.

E **loop** [*n* | INFINITE] - Specifies how many times a video clip will loop when activated. If LOOP="-1", or if LOOP="INFINITE" is specified, it will loop indefinitely.

E **start** [FILEOPEN | MOUSEOVER] - Specifies when the video clip should start playing. FILEOPEN means start playing as soon as the file is done opening. This is the default. MOUSEOVER means start playing when the user moves the mouse cursor over the animation. Both can be used together.

Example:

```
<HTML><HEAD>
<TITLE>Webber Index</TITLE>
</HEAD>
<BODY>
<H1><IMG SRC="logo.gif" ALIGN=TOP>Webber Index</H1>
<P><A HREF="http://www.csdcorp.com/">CSD Home Page</A></P>
<P><A HREF="Welcome.html">Webber Home Page</P>
<P><A HREF="#Release">Release Notes</P>
<H3><A NAME="Release">Release Notes</A></H3>
<P>These are the release notes for Webber.</P>
```

```
</BODY><HTML>
```

Typical Rendering (approximate):

Webber Index

CSD Home Page
Webber Home Page
Release Notes

Release Notes

These are the release notes for Webber.

Image Assistant

The Image Assistant provides a quick and easy way to add an image with the basic attributes. Until recently graphics had to be in GIF (.gif) format but most browsers now also support JPEG (.jpg) images. Images can be aligned to the top, middle or bottom of the text beside them. Some browsers can only display one line of text beside an image and will wrap additional text below, but newer browsers, such as Explorer and Netscape, support left and right alignment of images, which allows you to wrap multiple lines of text beside an image.

To Use:

1 Place your cursor in the position where you want the image tag to be inserted.
2 Open the image assistant by selecting it from the HTML menu or by clicking on the Quick-Tag button on the tool bar.
3 Enter the name of the image file, or browse for a local file by clicking the browse button. If the file is not a local one you will need to enter the full url for the image. Use the Path Options to tell Webber whether you want absolute, relative or no paths to be included when inserting a file that has been browsed to.
4 Enter any alternate text you want displayed. This is optional and is used by some browsers which can not display graphics. It is also used by browsers when auto-load of images is turned off. It is a good idea to use the ALT attribute so that someone reading your pages can make sense of them even if they do not have their images on.
5 Set height and width attributes if desired. The image assistant will try to find the image and automatically get the height and width for you. This only works if the image is local and the directory path that you have entered is correct for your local system.
6 Set the HSPACE, VSPACE and BORDER attributes if desired.
7 Select the alignment position for the image in relation to text that may appear beside it. (This is optional, the default alignment is bottom.)
8 When you are finished, click the Ok button to insert the tag into the document or the Cancel button to leave without inserting the tag.

Insert

compatibility

<ins></ins>

Usage:

This tag is used to mark inserted text, for instance in legal documents. It is used within a 'container' tag such as a paragraph.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
</HEAD>
<BODY>
<H1>Other information about HTML.</H1>
<P>There is lots of information to be had about HTML on the
<DEL>Internet</DEL> <INS>World Wide Web</INS>.</P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Italics

compatibility

****

Usage:

This tag is used for placing emphasis on a word or phrase. Browsers should render it in italics.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Glossary</TITLE>
</HEAD>
<BODY>
<H1>Glossary</H1>
<P>This is the <I>glossary</I> where you can find definitions for
common phrases.</P>
</BODY><HTML>
```

Typical Rendering (approximate):

Glossary

This is the *glossary* where you can find definitions for common phrases.

Item

compatibility

<item></item>

Usage:

This tag is used for items within a <ROW> of the <ARRAY> element which is used only in the <MATH> tag.

Attributes:

Required

Optional

align [LEFT | CENTER | RIGHT] - Used to override the coldef attribute on the parent array tag if it is used. By default, expressions in items are centered horizontally.

colspan - Used to provide a single item which spans several columns. The attribute value is a positive integer and defaults to one.

rowspan - Used to provide a single item which spans several rows. The attribute value is a positive integer and defaults to one. It can be used together with the COLSPAN attribute.

Example:

```
<array>
<row><item>a_11_<item>a_12_<item>&cdots;<item>a_1n_
<row><item>&vdots;<item>&vdots;<item>&ddots;<item>&vdots;
<row><item>a_n1_<item>a_n2_<item>&cdots;<item>a_nn_
</array>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Java Applet

compatibility

<APP CLASS="">

Usage:

The <APP> element is used to embed a Java applet in an HTML document. It has recently been replaced by the <APPLET> tag.

Attributes:

Required

class - Used to identify the name of the applet.

Optional

src - specifies the directory (folder) containing the classes directory that contains the applet. (If the Applet is named MyApplet, then its compiled file is named MyApplet.class.) If you don't specify this attribute, HotJava looks for the ".class" file in the classes directory under the directory that the HTML file containing the APP tag is in.

align - specifies how the applet's image should be aligned with the text that follows. The default alignment is bottom.

width - indicates the width of the applet image. HotJava uses these to lay out the page properly while it's waiting for the applet to draw itself.

height - indicates the height of the applet image. HotJava uses these to lay out the page properly while it's waiting for the applet to draw itself.

AppletSpecificAttribute - is any attribute name the applet allows. The applet writer specifies what (if any) attributes the applet accepts. aValue is a string that should be enclosed in quotation marks (but doesn't necessarily have to be). The applet determines how to interpret this string.

Example:

```
<HTML><HEAD>
<TITLE>Java Example</TITLE>
</HEAD>
<BODY>
<H1>Java Example</H1>
<APP CLASS="myapp" SRC="http://mycompnay.com/myapp>
</BODY>
</HTML>
```

JavaScript Event Handlers

This information is from the Netscape Web site. At the time of publication, only the Netscape browser has support for Javascript although Microsoft recently announced that they intended to support it. Information on JavaScript objects and properties is available on the Netscape Web site.

The following event handlers are available in JavaScript:

- onBlur
- onChange
- onClick
- onFocus
- onLoad
- onMouseOver
- onSelect
- onSubmit
- onUnload

onBlur

A blur event occurs when a text or textArea field on a form loses focus. The onBlur event handler executes JavaScript code when a blur event occurs.

Applies to
selection, text, textarea

onChange

A change event occurs when a selection, text, or textArea field loses focus and its value has been modified. The onChange event handler executes JavaScript code when a change event occurs.

Use the onChange event handler to validate data after it is modified by a user.

Applies to
selection, text, textarea

onClick

For button or radioButton, JavaScript code to run when a button is clicked. For checkbox, JavaScript code to run when user checks or unchecks an item.

Applies to
checkbox, radiobutton, link, reset, submit

Examples
For example, suppose you have created a JavaScript function called compute(). You can execute the compute() function when the user clicks a button by calling the function in the onClick event handler, as follows:

```
<INPUT TYPE="button" VALUE="Calculate" onClick="compute(this.form)">
```

In the above example, the keyword this refers to the current object; in this case, the Calculate button. The construct this.form refers to the form containing the button.

For another example, suppose you have created a JavaScript function called pickRandomURL() that lets you select a URL at random. You can use the onClick event handler of an anchor to dynamically specify a value for the HREF attribute of the anchor, as shown in the following example:

```
<A HREF=""
   onClick="this.href=pickRandomURL();"
   onMouseOver="window.status='Pick a random URL'; return true">
Go!</A>
```

In the above example, the onMouseOver event handler specifies a custom message for the Navigator status bar when the user places the mouse pointer over the Go! anchor. As this example shows, you must return true to set the window.status property in the onMouseOver event handler.

onFocus

Executed when input focus enters the field, either by tabbing in or by clicking but not selecting in the field.

Applies to
selection, text, textarea

onLoad

A load event occurs when Navigator finishes loading a window or all frames within a FRAMESET. The onLoad event handler executes JavaScript code when a load event occurs.

Use the onLoad event handler within either the <BODY> or the <FRAMESET> tag, for example, <BODY onLoad="...">.

Applies to
document, window

See also
onUnload event handler

onMouseOver

Executes when the mouse pointer is over an object. You must return true if you want to set the window.status property with the onMouseOver event handler.

Applies to
link

Examples

By default, the HREF value of an anchor displays in the status bar at the bottom of the Navigator when a user places the mouse pointer over the anchor. In the following example, the onMouseOver event handler provides the custom message "Click this if you dare."

```
<A HREF="http://home.netscape.com/"
   onMouseOver="window.status='Click this if you dare!'; return true">
Click me</A>
```

See onClick for an example of using onMouseOver when the anchor HREF attribute is set dynamically.

onSelect

A select event occurs when a user selects some of the text within a text or textArea field. The onSelect event handler executes JavaScript code when a select event occurs.

Applies to
text, textarea

onSubmit

Specifies the JavaScript code to run when a user attempts to submit a form. Return true to allow the form to be submitted; return false to prevent the form from being submitted.

Applies to
form

Examples

In the following example, the onSubmit event handler evaluates the data being submitted to test if it is legal. If the data is legal, the form is submitted; otherwise, the form is not submitted.

```
form.onSubmit=
"if badFormData(this.form) {
    return false;
} else {
    return true;
}"
```

onUnload

An unload event occurs when you exit a document. The onUnload event handler executes JavaScript code when an unload event occurs.

Use the onUnload event handler within either the <BODY> or the <FRAMESET> tags, for example, <BODY onUnload="...">.

Applies to
document, window

See also
onLoad event handler

Key-Combo Entities

Several keyboard short-cuts are available for inserting the entities of common characters into your HTML document. All entity short-cuts use the shift+ctrl keys in combination with another key. The available short-cuts are as follows:

Character	Description	Entity	Key-Combo
&	ampersand	&	ctrl + shift + 7
<	left angle bracket	<	ctrl + shift + ,
>	right angle bracket	>	ctrl + shift + .
"	double quote	"	ctrl + shift + '
©	copyright	©	ctrl + shift + c
™	trademark	™	ctrl + shift + t
®	registered trademark	®	ctrl + shift + r

Keyboard

compatibility

<kbd></kbd>

Usage:

The keyboard tag indicates that the text is something that the user types or an instruction to the user. This tag is often used for reference manuals.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Webber Reference</TITLE>
</HEAD>
<BODY>
<H1>Webber Reference</H1>
<HR>
<P>To open the help file, select <KBD>Help|Contents</KBD> from the
menu.</P>
</BODY></HTML>
```

Typical Rendering (approximate):

Webber Reference

To open the help file, select `Help|Contents` from the menu.

Keyword Index

compatibility

<isindex>

Usage:

This element should not be added by hand. This tag is usually added by a server script. It tells the browser that the document can be examined using a keyword search. Keywords can be passed to the server by adding a question mark (?), followed by a list of keywords separated by plus signs (+), to the url. (http://www.somewhere.com/cgi-bin/srchscript?authoring+tools).

Attributes:

Required

Optional

3 href - Used to point to the server handling the searching.

3 prompt - Used to specify the prompt message.

N E **action** - (ACTION="filename") Specifies the gateway program to which the string in the text box should be passed. <ISINDEX ACTION="search">

Example:

```
<html>
<head>
<title>Example</title>
<isindex>
</head>
<body><p>The document content.</p></body>
</html>
```

Typical Rendering (approximate):

This is a searchable index. Enter search keywords: |_____|

Language

compatibility

<lang></lang>

Usage:

This tag can be used to alter the language context when it is inappropriate to do this with other character-level elements. It is used within a 'container' tag such as a paragraph.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Working Committee</TITLE>
</HEAD>
<BODY>
<H1>Committee Members</H1>
<P>The members of this committee are:
<BR>Jane Smith
<BR>John Doe
<BR><LANG LANG="en.uk">Jan Green</LANG>.</P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Left

compatibility

<left></left>

Usage:

Used to place a delimiter, which grows to match the height of the expression, before an expression. This tag is used only within the <BOX> element in a <MATH> element.

Attributes:

Required

Optional

Example:

```
{(<left>1+x<over> sin x <right>)}
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

License Agreement

Webber™ Version 1.4 Unregistered (For the Registered version, see here)

Information on registration, and why you should register, installation and deletion.

YOUR INSTALLATION AND USE OF THIS SOFTWARE CONSTITUTES YOUR AGREEMENT TO THE TERMS OF THIS LICENSE. THIS IS A LEGAL AGREEMENT TO WHICH YOU ARE CONSENTING TO BE BOUND. IF YOU DO NOT AGREE TO ALL OF THE TERMS OF THIS LICENSE, DO NOT USE THE SOFTWARE - DELETE IT FROM YOUR SYSTEM.

Evaluation and Registration

Cerebral Systems Development Corporation ("CSD") grants to you a non-exclusive, non-sublicensable, license to use the evaluation version 1.4 of Webber™ (the "Software"), in binary executable form for evaluation and trial use purposes only.

This is not free software. You are hereby licensed to use the Software, for evaluation purposes only, without charge for a period of 30 days. In order to continue using the Software after the evaluation period you must register. If, after the evaluation period, you do not find that the Software meets your requirements, you must destroy all copies of the software.

Unregistered use of the Software after the 30 day evaluation period is in violation of Canadian and international copyright laws.

Distribution

You are hereby licensed to copy the Software and/or to make it available for download. All files, including this license agreement must be kept together as part of the package.

YOU ARE SPECIFICALLY PROHIBITED from charging, or requesting donations, for any such copies, however made; and from distributing the software and/or documentation with other products (commercial or otherwise) without prior written permission.

CSD MAKES NO REPRESENTATIONS ABOUT THE SUITABILITY OF THIS SOFTWARE (SHAREWARE VERSION OR REGISTERED VERSION) FOR ANY PURPOSE. THE SOFTWARE IS PROVIDED 'AS IS' WITHOUT WARRANTIES, EXPRESS OR IMPLIED, INCLUDING WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. CSD SHALL NOT BE LIABLE UNDER ANY THEORY OR ANY DAMAGES SUFFERED BY YOU OR ANY USER OF THE SOFTWARE.

The Software and all supporting files and documentation, are the sole property of Cerebral Systems Development Corporation (CSD), unless otherwise specified. CSD reserves all title ownership rights, and intellectual property rights to the Software. You agree to be bound by Canadian and international copyright laws and all other applicable laws of Canada.

Governing Law

This agreement shall be governed by the laws of the Province of Ontario, Canada.

Trademarks

Cerebral Systems Development, the Cerebral Systems Development logo, Webber, the Webber

License Agreement

Webber™ Version 1.4 Registered (For the evaluation version, see here)

YOUR INSTALLATION AND USE OF THIS SOFTWARE CONSTITUTES YOUR AGREEMENT TO THE TERMS OF THIS LICENSE. THIS IS A LEGAL AGREEMENT TO WHICH YOU ARE CONSENTING TO BE BOUND. IF YOU DO NOT AGREE TO ALL OF THE TERMS OF THIS LICENSE, DO NOT USE THE SOFTWARE - DELETE IT FROM YOUR SYSTEM.

Cerebral Systems Development Corporation ("CSD") grants to you a non-exclusive, non-sublicensable, license to use Version 1.4 of Webber™ (the "Software"), in binary executable form according to the following terms:

You are hereby licensed to:

1 Use the Software on one computer.
2 Use the Software on a second computer belonging to you, so long as the two computers are not used at the same time. (For example, your desktop and your laptop computers.)
3 Make one back-up copy of the Software.
4 Access the registered version of the Software through a network, provided that you have obtained individual licenses for the Software for each workstation that can access the Software through the network.

You are not permitted to:

1 Copy the Software or its documentation except for the purposes of back-up.
2 Sell, rent, lease, or otherwise transfer rights to the Software or its documentation.
3 Sell, distribute, upload or make available copies of the Software to others by any means for any purpose.
4 Remove any proprietary notices, labels or copyrights from the Software, its documentation or packaging.
5 Modify the Software or attempt to decipher, decompile, disassemble or reverse engineer the Software.

CSD MAKES NO REPRESENTATIONS ABOUT THE SUITABILITY OF THIS SOFTWARE (SHAREWARE VERSION OR REGISTERED VERSION) FOR ANY PURPOSE. THE SOFTWARE IS PROVIDED 'AS IS' WITHOUT WARRANTIES, EXPRESS OR IMPLIED, INCLUDING WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. CSD SHALL NOT BE LIABLE UNDER ANY THEORY OR ANY DAMAGES SUFFERED BY YOU OR ANY USER OF THE SOFTWARE.

The Software and all supporting files and documentation, are the sole property of Cerebral Systems Development Corporation (CSD), unless otherwise specified. CSD reserves all title ownership rights, and intellectual property rights to the Software. You agree to be bound by Canadian and international copyright laws and all other applicable laws of Canada.

Governing Law

This agreement shall be governed by the laws of the Province of Ontario, Canada.

Trademarks

Cerebral Systems Development, the Cerebral Systems Development logo, Webber, the Webber

Line Break

compatibility

**
**

Usage:

Since HTML documents ignore carriage returns (except when inside a PRE tag), the line break tag is necessary to tell the browser to place text following it on the next line. It does not add any extra vertical white space. It is an empty tag.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 N clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

Example:

```
<HTML><HEAD>
<TITLE>Glossary</TITLE>
</HEAD>
<BODY>
<H1>Glossary</H1>
<P>A<BR>B<BR>C<BR>D<BR>E<BR>F<BR>G
<BR>H<BR>I<BR>J
<BR>etc.
</P></BODY></HTML>
```

Typical Rendering (approximate):

Glossary

A
B
C
D
E
F
G
H

I
J
etc.

Link from Document

compatibility

<link>

Usage:

The link tag is used to indicated the relationship of the document to another document. It is not often used and is not yet supported by browsers. Typical uses would be to indicate authorship, related indexes and glossaries, older or more recent versions, etc. Multiple link tags can be used. The same attributes are used as in the Anchor tag.

In HTML 3.0 the link tag is intended to be used to link to tool bar buttons, banners and style sheets. i.e. `<LINK REL="Previous">`- references the previous document in a guided tour, `<LINK REL=Banner HREF=banner.html>`, and `<LINK REL="StyleSheet" HREF="style.dsl">`. These features are not yet commonly implemented.

Attributes:

Required

href - address of link destination

Optional

rel - relationship to destination
rev - reverse relationship - of destination to this document
title - title of destination
methods - operations allowed
urn - lasting name of destination

Example:

```
<HTML><HEAD>
<TITLE>Webber Home Page</TITLE>
<LINK href="http://www.csdcorp.com/" rel="Authors of Webber"
title="CSD Corp. Home Page">
<LINK href="index.htm" rel="parent">
</HEAD>
<BODY></BODY>
</HTML>
```

Typical Rendering (approximate):

Not currently applicable - browsers may support this in the future.

List Assistant

The List Assistant allow you to quickly add bulleted, numbered or glossary style lists to your document. It can be used to insert the list tag or any number of list items on their own. Definition lists (glossary style) have two type of list items - definition terms, and definition descriptions - these will be inserted in an alternating fashion, in equal numbers. (i.e. if you select 5 list items you will get 5 DT and 5 DD.) If you are using HTML 3, you can also enter a list header and an id for the list tag.

To insert an empty list:

1 Open the list assistant by selecting HTML|List from the menu.
2 Select from unordered, ordered, or definition lists, or select 'no list tag' to insert only list items.
3 Set the number of list items to insert.
4 Check the 'compact' attribute if desired.
5 If you are using HTML 3, enter an optional list header, or an id for the list. You can also set the list items to be 'plain' (i.e. no bullets) or specify an image to be used as the bullet, if you have selected an unordered list. If you have selected an ordered list you can set the starting sequence number.
6 Click the Ok button to insert the list.
7 Type your content into your document in the appropriate place.

To insert a list around existing content:

1 Select the content that you want to make into a list.
2 Open the list assistant by selecting HTML|List from the menu.
3 Select from unordered, ordered, or definition lists, or select 'no list tag' to insert only list items.
4 The list assistant will have calculated the number of list items that are required for the selected text (based on the number of paragraphs in the selection). You can not enter fewer list items but you can insert more. Extra list items will be added to the bottom of the new list.
5 Check the 'compact' attribute if desired.
6 If you are using HTML 3, enter an optional list header, or an id for the list. You can also set the list items to be 'plain' (i.e. no bullets) or specify an image to be used as the bullet, if you have selected an unordered list. If you have selected an ordered list you can set the starting sequence number.
7 Click the Ok button to wrap the list around your content.

List Header

compatibility

<lh></lh>

Usage:

The list header allows you to place a description of a list within the list itself. It is an optional tag and must appear within a list tag before the first list item if it is to be used. While this tag is not yet well supported by browsers, it does appear to be possible to use it without breaking them. (As tested with Netscape 1.1, 1.2 and Mosaic - on the Windows platform)

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Index</TITLE>
</HEAD>
<BODY>
<UL><LH>Index</LH>
<LI>Ch. 1</LI>
<LI>Ch. 2</LI>
<LI>Ch. 3</LI>
<LI>Ch. 4</LI>
</UL>
</BODY><HTML>
```

Typical Rendering (approximate):

Index

- Ch. 1
- Ch. 2
- Ch. 3
- Ch. 4

List Item

compatibility

**
**

Usage:

The list item is used in unordered lists, ordered lists, menu lists and directory lists. Typically, it is indented with a bullet or number (depending on the type of list) before any content.

Menu lists and directory lists have been deprecated in HTML 3 and replaced by the UL.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

3 skip - Increments the sequence number before rendering the element. It is used when headers have been left out of the sequence.

3 dingbat - Used to specify an iconic image to appear preceding the list item.

3 src - Used to specify an image to appear preceding the list item.

3 md - Specifies a message digest used when you want to be sure that a linked graphic, specified by the SRC attribute, is the same one intended, and hasn't been changed in any way.

N E **type** [CIRCLE | DISC | SQUARE | A | a | i | I | 1] - Used to change the bullet type or numbering type of a list item. The allowed value will depend on the type of list you are in. (i.e. UL means you can use a circle, disk or square.) This is handled by the dingbat and src attributes in HTML 3.0.

N E **value** - Used to change the count, for that list item and all subsequent (for ordered lists only).

Example:

```
<HTML><HEAD>
<TITLE>Index</TITLE>
```

```
</HEAD>
<BODY>
<H1>Index</H1>
<UL>
<LI>Ch. 1</LI>
<LI>Ch. 2</LI>
<LI>Ch. 3</LI>
<LI>Ch. 4</LI>
</UL>
</BODY><HTML>
```

Typical Rendering (approximate):

Index

- Ch. 1
- Ch. 2
- Ch. 3
- Ch. 4

Listing

compatibility

<listing></listing>

Usage:

The listing and example tags are obsolete and have been replaced by the preformat tag. They allow text to be displayed with spacing etc. exactly as it was typed. Browsers will usually use a monospaced font to render the content of these tags.

This tag has been deprecated in HTML 2.0 and is already obsolete in HTML 3.0. It is recommended that you use the preformat tag instead.

Attributes:

Required

Optional

Example:

```
<HTML><HEAD>
<TITLE>Webber Reference</TITLE>
</HEAD>
<BODY>
<H1>Webber Reference</H1>
<HR>
<LISTING>A listing tag retains any
spacing      and
carriage returns that you type. You should, however
use the preformat tag instead.</LISTING>
</BODY></HTML>
```

Typical Rendering (approximate):

Webber Reference

```
A listing tag retains any
spacing      and
carriage returns that you type. You should, however
use the preformat tag instead.
```

Long File Name Support

Long file name support can be enabled on the Editor tab of the Preferences dialog. When enabled, you can use long file names and long directory names under the Windows 95 and Windows NT operating systems. The file name dialogs have been designed to offer similar functionality to the Windows 95 dialogs. Those accustomed to Windows 95 will notice some minor differences but these will be addressed in the full 32-bit version.

The one remaining limitation of the long file name support is in the 'browse' dialogs used with the image and anchor assistants. A maximum of 255 characters can be returned by these controls, larger numbers will be truncated. This will not be a problem for most users unless you are using absolute paths and have a deeply nested directory structure. We apologize for this temporary limitation, it will also be fixed in the 32-bit version.

Macros

Macros available for use in the header information include:

copyright	inserts the copyright notice from Options| Preferences|User Info.
date modified	inserts/updates the date that the document has been modified each time it is saved. You can specify a number of days in conjunction with this macro - i.e. [date modified]+14 would result in Webber inserting a date 14 days from the date the file was modified. Do not place spaces between the macro, the plus sign or the number. NOTE: Be sure that you have set up your time zone in the preferences.
date created	inserts the creation date of the document when the docinfo is inserted or when a New Html document is created. You can specify a number of days in conjunction with this macro - i.e. [date created] + 14 would result in Webber inserting a date 14 days from the date the file was created. Do not place spaces between the macro, the plus sign or the number. NOTE: Be sure that you have set up your time zone in the preferences.
dtd	inserts the name of the current dtd when the file is created or saved
e-mail	inserts the e-mail address from Options| Preferences|User Info.
generator	inserts the name and version of the generating program - in this case, Webber.
prompt create	results in a dialog that prompts for the specified information when a file is created or the docinfo is inserted. (Useful for keywords etc.)
prompt modify	results in a dialog that prompts for the specified information when a file is created or modified or when the docinfo is inserted. (Useful for keywords etc.)
version	inserts the version number of a new document - i.e. Version 1 - which a user can then modify when appropriate

Map

compatibility

<map></map>

Usage:

The map tag is used to identify regions of a client-side image map (identified by the USEMAP attribute of the IMG tag). It contains areas defined by the AREA tag.

This tag is new with Netscape 2.0 and Explorer 2.0.

Attributes:

Required

name - specifies the name of the map so that it can be referenced by an IMG element.

Optional

Example:

```
<HTML><HEAD>
<TITLE>Client Side Image Maps</TITLE>
</HEAD>
<BODY>
<H1>Client Side Image Maps</H1>
<P><MAP NAME="buttonbar">
<AREA SHAPE="RECT" COORDS=" 305, 45,  411, 101" HREF="business.htm">
<AREA SHAPE="RECT" COORDS=" 151, 106, 258, 142" HREF="recreatn.htm">
<AREA SHAPE="RECT" COORDS=" 18, 82, 143, 115"
HREF="townhall.htm"></MAP>
<IMG SRC="../images/tech/bar.gif" USEMAP="#buttonbar"></P>
</BODY><HTML>
```

Typical Rendering (approximate):

Not applicable.

Marquee

compatibility

<marquee></marquee>

Usage:

The MARQUEE tag enables you to create a scrolling text marquee. Marquees can be left- or right-aligned, like images, and have a variety of attributes to control them. If no attributes are specified the text will scroll from left to right.

Attributes:

Required

Optional

align [TOP | MIDDLE | BOTTOM] - Specifies that the text around the marquee should align with the top, middle, or bottom of the marquee.

behavior [SCROLL | SLIDE | ALTERNATE] - Specifies how the text should behave. SCROLL (the default) means start completely off one side, scroll all the way across and completely off, and then start again. SLIDE means start completely off one side, scroll in, and stop as soon as the text touches the other margin. ALTERNATE means bounce back and forth within the marquee.

bgcolor [#rrggbb | *colorname*] - Specifies a background color for the marquee, either as a RGB triple or using a "friendly" colorname. See the Explorer web site for info on colornames.

direction - [LEFT | RIGHT] - Specifies which direction the text should scroll. The default is LEFT, which means scrolling to the left from the right.

height - Specifies the height of the marquee, either in pixels or as a percentage of the screen height.

hspace - Specifies left and right margins for the outside of the marquee, in pixels.

loop [*n* | INFINITE] - Specifies how many times a marquee will loop when activated. If loop="-1", or if LOOP="INFINITE" is specified, it will loop indefinitely.

scrollamount - Specifies the number of pixels between each successive draw of the marquee text.

scrolldelay - Specifies the number of milliseconds between each successive draw of the marquee text.

vspace - Specifies top and bottom margins for the outside of the marquee, in pixels.

width - Sets the width of the marquee, either in pixels or as a percentage of the screen width.

Example:

```
<HTML><HEAD>
<TITLE>Marquee in Internet Explorer</TITLE>
</HEAD>
<BODY>
<H1>Marquee in Internet Explorer</H1>
<MARQUEE BEHAVIOR="SCROLL">This text will scroll all the way on and
then all the way off.</MARQUEE>
</BODY>
```

</HTML>

Math

compatibility

$$

Usage:

The MATH tag is used to include in-line math expressions. HTML math doesn't provide direct support for multi-line equations, but this can be handled by combining math with the <TABLE> tag.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<MATH>&int;_a_^b^{f(x)<over>1+x} dx</MATH>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Menu List

compatibility

<menu></menu>

Usage:

HTML supports several types of lists which may be nested within themselves or each other. The menu list is similar (with some browsers identical) to the directory list and unordered list. It contains list items which usually result in left aligned, bulleted paragraphs. It is meant to be used for smaller paragraphs than the unordered list and may be rendered with less vertical white space depending on the browser.

This tag has been deprecated in HTML 3.0 and will be obsolete in the future. It is recommended that you use an unordered list instead.

Attributes:

Required

Optional

compact - [COMPACT] - suggests that the list be rendered with a minimum of white space, not all browsers make use of this attribute.

Example:

```
<HTML><HEAD>
<TITLE>Index</TITLE>
</HEAD>
<BODY>
<H1>Index</H1>
<MENU>
<LI>Ch. 1</LI>
<LI>Ch. 2</LI>
<LI>Ch. 3</LI>
<LI>Ch. 4</LI>
</MENU>
</BODY><HTML>
```

Typical Rendering (approximate):

Index

- Ch. 1
- Ch. 2
- Ch. 3
- Ch. 4

(tagname) element not allowed at this point in (tagname) element

The tag (element) encountered was not legal within the current tag. For example an <INPUT> tag was encountered within a <BODY> without a <FORM> tag first.

Required (attribute value) attribute was not specified; may affect processing

A required attribute is missing such as the SRC attribute for the image tag. Check the tag assistant to see what attributes are required for this tag. Be sure you know which dtd you are validating against.

(tagname) end-tag implied by (tagname) end-tag; not minimizable
(tagname) end-tag implied by (tagname) start-tag; not minimizable
(tagname) end-tag implied by data; not minimizable
(tagname) end-tag implied by short start-tag (no GI); not minimizable

An end tag for a currently open start tag is missing and is required. While some end tags can be omitted (they are minimizable), many cannot. It is a good idea to always use end tags to avoid this problem. Find the position where the end tag should be and type it in then try again.

(tagname) start-tag implied by (tagname) start-tag; not minimizable
(tagname) start-tag implied by data; not minimizable
(tagname) start-tag implied by short start-tag (no GI); not minimizable

A required tag was omitted. For example <HTML><TITLE> might cause this error because the <HEAD> tag is required and missing between <HTML> and <TITLE>. A short start tag means that just the angle brackets were used (<>) with no tag name (general identifier) between them.

Possible attributes treated as data because none were defined

The tag has no attributes defined for it and some were given anyway. This can happen if the right angle bracket is left off of a tag which is followed by content. For example <H1 This is the first heading</H1>. Run the angle bracket matcher from the Search menu to find this if you don't see it.

(tagname) element ended prematurely; required (tagname) omitted
(tagname) element ended prematurely; required subelement omitted

A required tag is missing within the given tag. For example <TITLE> is required within <HEAD>. There are not many required tags in HTML so this message will not appear often.

No declaration for entity "(entity name)"; reference ignored

An entity (&name;) was found that was not defined in the currently selected dtd. Check the entity Assistant to see if you have used the correct entity name. There may be a spelling error, or the entity may not exist in the dtd.

Short end-tag (no GI) ignored: no open elements

A short end tag (</>) was encountered which does not match any open tag. 'No GI' refers to the fact that there is no 'general identifier' or name given for the tag. Try removing it if you think it might be an extra or type in the name of the tag that it is meant to end.

Out-of-context (tagname) start-tag ended (tagname) document element (and parse)

A tag was encountered in a position where it is not allowed. For example a line break is not allowed within the BODY - it must be contained within a container such as a paragraph or heading, etc. A list item must be contained within a list. Emphasis, strong etc. do not count as containers and must be 'contained' themselves.

Try positioning your cursor just after the start tag previous to this one. Then open the Tag Assistant, check the 'Tags in Context' button and check the drop-down list to see what is allowed.

Out-of-context data ended (tagname) document element (and parse

Data (text/content) was encountered in a position where it is not allowed. For example text cannot be located in the body without being wrapped inside a container such as a paragraph or heading. Emphasis, strong etc. do not count as containers and must be 'contained' themselves. In HTML 3 Recommended, you must also have a paragraph around the contents of table cells and list items.

Start tag of document element omitted; not minimizable

The <HTML> tag was left out and is required in the currently selected dtd.

Non-SGML character ignored
Non-SGML character found; should have been character reference

The document contains a character that is not allowed in HTML (or SGML) documents.

Examples of these might be 'smart' quotes and apostrophes that are inserted by other programs such as Microsoft Word. Have a close look for quotes or apostrophes - it is probably worth replacing them just in case. The other possibility is a non-printing character. These can get inserted inadvertently by pressing some key combinations (i.e. the ctrl key plus some others). This one is tough because you can't see these in most programs. Try using Windows notepad with the system font to find these. If all else fails you could try deleting the whole line and retyping it.

Length of name, number, or token exceeded NAMELEN or LITLEN limit.

The value of an attribute (such as the HREF) is longer than the maximum allowed value in the HTML specification. These are as follows (copied from the HTML 3 dtd):

```
"LITLEN   1024
 NAMELEN  72  --somewhat arbitrary; taken from internet line  length
conventions--"
```

These are specifications as defined by the IETF and should allow for plenty of length. Check for a missing quote or other error in the url.

Incorrect character in markup; markup terminated

Caused by illegal characters in the mark-up. For example, you might get this error if you forget to put the quotes around the url in an HREF. Check carefully for missing quotes around attribute values.

Short start-tag (no GI) ended (tagname) document element (and parse)

A 'short' start tag was encountered in a position where it is not allowed. 'Short' start tags are represented by just the angle brackets such as <>. 'No GI' refers to the fact that there is no 'general identifier' or name given for the tag. These were meant to be used as a typing convenience and are not recommended. They are only useful if the parser (or browser) can determine what they represent - which is not likely to be often! Enter the proper name inside the angle brackets and try again.

(tagname) end-tag ignored: doesn't end any open element (current is (tagname))

An end tag was encountered which did not have a matching start tag. You may have deleted the start tag and forgotten the end tag.

(tagname) start-tag exceeds open element limit; possible lies from (tagname) on

You have too many start tags without matching end tags. Too many is a large number that you should never hit in a normal document. There may be some other problem - recheck your document very carefully from the identified tag on for extra or missing angle brackets etc.

Start-tag omitted from (tagname) with empty content

You are missing the named start tag. Try typing it in and validating again.

Data not allowed at this point in (tagname) element

Some HTML tags do not allow data (characters or content). For example, under STRICT validation you cannot have content in a <BODY> tag, you must use a <P> or some other tag first. In HTML 3 you must also have a paragraph around the contents of table cells and list items.

No element declaration for (tagName) end-tag GI; end-tag ignored

The end tag can't be found. Probably caused by a typo or you have an end tag from another version of the dtd. For example the lits header is an HTML 3 tag and if you had the end tag </LH> in an HTML 2 document you would get this error.

(attribute Name) = "(attribute value)" attribute ignored: not defined for this element

An undefined attribute was used. Check Tag Assistant for the list of attributes for the tag. You may be using attributes which are only found in another version of the HTML dtd. Check your preferences and see which dtd you are validating against. For example the ALIGN attribute is defined for headings and paragraphs only in HTML 3, not HTML 2.

(attribute value) = "(attribute value)" attribute value defaulted: invalid character
(attribute Name) = "(attribute value)" attribute value defaulted: token too long
(attribute Name) = "(attribute value)" attribute value defaulted: token not in group

The value of the given attribute is not correct. For example you may have spelled the attribute name incorrectly such as using ALIGN="Center" instead of ALIGN="Center". Also you may have missed the quotation marks that are required around attribute values.

Meta-Information

compatibility

<meta>

Usage:

The meta tag is a a generic tag that can be used to embed information about a document that does not fit within the other HTML tags. If there is a tag more appropriate for this information, such as a title, then it should be used. Sometimes however an author may want to include other information to be used for specific purposes. Meta information is information about the document as opposed to information contained in the document. Web search and indexing systems would be likely users of Meta information.

Netscape and Internet Explorer support client pull using the META tag. The META tag must be inside the HEAD tag of the HTML document. The URL attribute requires a fully qualified URL (for example, http://www.sample.com/reload.htm).

Attributes:

Required

content - the content to be associated with the http-equiv or name response header.

Optional

http-equiv - used for HTTP response headers when the semantics are known
name - meta-information name

N E **HTTP-EQUIV= "REFRESH"** - Causes a document to be automatically reloaded on a regular basis, specified in seconds.

```
<HTML><HEAD><TITLE>Reload Document</TITLE>
<META HTTP-EQUIV="REFRESH" CONTENT=2></HEAD><BODY>
<P>This document will be reloaded every two seconds.</P></BODY></HTML>
```

N E **CONTENT="n; URL=URL"** - Tells the browser to reload in n seconds. If a URL is specified, the browser will load the URL after the time specified has elapsed. If no URL is specified, it will reload the current document.

```
<HTML><HEAD><TITLE>Load Next Document</TITLE>
<META HTTP-EQUIV="REFRESH" CONTENT="5;
URL=http://www.sample.com/next.htm"></HEAD><BODY>
<P>After five seconds have elapsed, the document
"http://www.sample.com/next.htm" will be loaded.</P></BODY></HTML>
```

Example:

```
<HTML><HEAD>
<TITLE>Webber Home Page</TITLE>
<META NAME="Author" CONTENT="Cerebral Systems Development Corp.">
<META HTTP-EQUIV="Keywords" CONTENT="CSD Corp., Webber, HTML Editor
"></HEAD>
<BODY></BODY>
</HTML>
```

Typical Rendering (approximate):

Not applicable.

Multiple File Open

Multiple files can be opened at one time from a single directory. From the File|Open dialogue select multiple files by holding down the shift or ctrl keys while clicking on the names of the files you want opened. When you have selected the files, click the OK button.

NAME Token
A sequence of characters starting with an alpha character followed by alphanumeric characters of a fixed length. For HTML this is a maximum of 72 characters.

N When used in front of an attribute, this symbol means that the attribute is used for the current tag as a Netscape Extension. It is supported by Netscape 1.1 and later. If it appears in conjunction with the HTML 3.0 symbol it means that the attribute has been incorporated into the HTML 3.0 dtd.

Netscape extensions are not fully supported by all browsers. Check your browser documentation or test your document with your browser to see if the attribute you are using is supported. Also keep in mind that there are many browsers on the market and significant numbers of your audience may be using browsers which don't support this feature.

Netscape Ext.
Please note that while Netscape extensions are supported by Webber and our validation feature, they are not yet part of the HTML specification. While it is possible that many or all of these extensions may be adopted as part of the standard, there is no guarantee that this will happen. If you use these extensions you should be aware that they may undergo changes. Documents using these extensions are not yet officially compliant and may never be.

Netscape Extensions

Netscape Extensions are additions made to the HTML specification by Netscape Communications. For the complete list of tags supported by Netscape, see the Quick Tag Reference.

Tags:

Plus attributes on:

If this symbol appears alone it tells you that this tag is a ***Netscape Extension***. Not all attributes of this tag may apply - check the attribute list for this tag for more information.

If this symbol appears with an HTML 3 symbol (3), it means that Netscape has added support for this HTML 3 element to their browser.

Netscape Extensions are not fully supported by all browsers. Check your browser documentation or test your document with your browser to see if the tag you are using is supported. Keep in mind that if you use unsupported features there may be a significant number of people viewing your document who use other browsers which do not support these extensions.

Creating a New HTML Document

New HTML documents can be created in several ways.

1 Select File|New from the menu, or click the new button on the tool bar. Type or paste in your content and use the tag bar or assistants to add tags.

2 Select File|New HTML from the menu. A basic untitled HTML document will be created containing the HTML identifier, and HEAD, TITLE, and BODY tags. A stamp will also be inserted at the end of the document if the feature has been enabled in Preferences|User Info. Type or paste in your content and add more tags as appropriate.

3 Convert a text document to an HTML document by using the Auto Tag feature which inserts basic HTML elements into a copy of your document. Edit the tags or add new tags as appropriate.

!! NEW !! New in Version 1.4

Welcome to Version 1.4 of Webber™ - HTML Editor Extraordinaire! Version 1.4 has lots of new features that power-users of HTML will love! The following list gives a brief overview of the great new features of Webber™ and the bugs which have been fixed.

Custom Bits - can be defined and assigned to the tagbar. "Custom Bits" are user-definable 'bits' of text, tag combinations, entities etc. For example you can define a centered heading (i.e. <H1 ALIGN="center"></H1>) and place it on the tagbar, or assign a shortcut key combination to it.

Floating/Dockable tool bar and tagbar - the tool bar and and customizable tagbar can float or dock to the top or bottom, or sides of your window.

Custom keyboard shortcuts - keyboard shortcuts can be assigned to any tag or custom bit. The function keys, or the ctrl, shift and alt keys can be used in combination with the alphanumeric keys to create user-definable shortcut key combinations.

Frame Assistant - the frame assistant allows you to visually layout frames, and assign an HTML file to them. You can build both simple and complex, nested frames very easily.

Header (META) information - header (META) information can be generated automatically based on specified settings and/or several predefined 'macros', perfect for expiry dates, author name, version number, keywords etc. For example:

```
<META HTTP-EQUIV="Keywords" CONTENT="Webber V 1.4, CSD Corp., HTML
editor">
<META NAME="Generator" CONTENT="Webber V1.4">
<META NAME="Copyright" CONTENT="1996 Cerebral Systems Development
Corp.">
```

Structure Assistant - the structure assistant shows an outline structure view of your document. From it you can follow links to open local linked files, and create new links for anchors and images, or you can pass a tag to the tag assistant for editing.

Table Editor picks up tables - much like the tag assistant picks up tag and attribute information, the table editor can pick up existing tables, including their contents, from a document, making it easy to make changes to it.

Image Assistant and Tag Assistant - automatically pick up the height and width of images when you browse to them or select them from the drop-down lists.

New DTDs - Full support for Internet Explorer 3.0, and new Netscape extensions.

Fixed in V1.4
Save dialogs - now allow characters such as dashes to be used in file names.
table editor - no longer GPFs when opened maximized
alt+tab key - no longer inserts spaces when switching back to Webber
browsers - can be deleted from the preferences
spell checker - problems with the spell checker have been fixed

Next ID to Use

compatibility

<nextid>

Usage:

This element should not be inserted by hand. It is a parameter used by automated HTML editors to create unique identifiers for the documents.

Attributes:

Required

n - the number of the next numeric identifier available to be allocated in the current document. The syntax used is z###.

Optional

Example:

```
<html>
<head>
<title>Example</title>
<nextid n=z123>
</head>
<body><P>The document content.</P></body>
</html>
```

Typical Rendering (approximate):

Not applicable.

No Break

compatibility

<nobr></nobr>

Usage:

This tag stands for 'no break'. This means all the text between the start and end of the NOBR elements cannot have line breaks inserted between them.

In HTML 3.0, wrapping can be disabled by using the nowrap attribute on 'container tags' such as paragraphs and headings. It seems unlikely at this point that the <nobr> tag will be included in the HTML specification.

Attributes:

Required

Optional

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
</HEAD>
<BODY>
<H1><NOBR>Other information about HTML.</NOBR></H1>
<P>There is lots of information to be had about HTML on the
Internet.</P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

(sorry - can't represent no breaks here!)

Other Information about HTML.

There is lots of information to be had about **HTML** on the Internet.

Noframe

compatibility

<noframe></noframe>

Usage:

This tag is for content providers who want to create alternative content that is viewable by non-Frame-capable browsers. A Frames-capable browser ignores all tags and data between start and end NOFRAMES tags.

Attributes:

Required

Optional

Example:

```
<HTML><HEAD><TITLE>Test Frame Document</TITLE></HEAD>
<FRAMESET COLS="50%,50%">
  <FRAMESET ROWS="50%,50%">
    <FRAME SRC="cell.html">
    <FRAME SRC="cell.html">
  </FRAMESET>
  <FRAMESET ROWS="33%,33%,33%">
    <FRAME SRC="cell.html">
    <FRAME SRC="cell.html">
    <FRAME SRC="cell.html">
  </FRAMESET>
</FRAMESET>
<NOFRAME>
<BODY>
<H1>Webber Index</H1>
<P><A HREF="http://www.csdcorp.com/">CSD Home Page</A></P>
<P><A HREF="Welcome.html">Webber Home Page</A></P>
<P><A HREF="#Release">Release Notes</A></P>
<H3><A NAME="Release">Release Notes</A></H3>
<P>These are the release notes for Webber.</P>
</BODY>
</NOFRAME></HTML>
```

Normal Paragraph

compatibility

** **

Usage:

The paragraph tag is used to separate two blocks of text. It usually causes a vertical space between the two text blocks. Some browsers will indent the first line of a paragraph.

The paragraph tag is not an empty tag but it is not uncommon to use only the start tag and to omit the end tag. It is recommended that the end tag be used but it is not required. Many of Webber's features, such as Tag Select and Tags in Context, work best if end tags are used.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

3 align [LEFT | CENTER | RIGHT | JUSTIFY] - Used to specify the horizontal alignment of the paragraph. The default is left.

3 nowrap [NOWRAP] - Used to prevent the browser from automatically wrapping lines.

Example:

```
<HTML><HEAD>
<TITLE>Webber Home Page</TITLE>
</HEAD>
<BODY>
<H1>Welcome to the Webber Home Page</H1>
<P>This is the home page for the hot, easy to use, HTML editor -
Webber.</P>
<P>Webber was designed as a tool for HTML authors who want to see and
control what is happening in their document at the tag
level.</P></BODY>
</HTML>
```

Typical Rendering (approximate):

Welcome to the Webber Home Page

This is the home page for the hot, easy to use, HTML editor - Webber.

Webber was designed as a tool for HTML authors who want to see and control what is happening in their document at the tag level.

Note

compatibility

<note></note>

Usage:

This tag is for use as admonishments such as notes, cautions or warnings, as commonly used in technical documentation. The CLASS attribute specifies the type of the element and is typically associated with different graphics such as a road traffic warning sign. The graphic can be customized with the SRC attribute.

The class names: NOTE, CAUTION and WARNING are recommended for standard admonishments. (see example) In the absence of the CLASS attribute, this element is typically rendered indented, without an accompanying graphic.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

src - Used to specify an image to appear preceding the note.

md - Specifies a message digest used when you want to be sure that a linked graphic, specified by the SRC attribute, is the same one intended, and hasn't been changed in any way.

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
</HEAD>
<BODY>
<H1>Other information about HTML.</H1>
<NOTE CLASS=NOTE>The information that is contained on pages linked
from here is the property of the authors of those pages. CSD Corp.
claims no right to these pages.</NOTE>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

OF

compatibility

<of></of>

Usage:

Used in the <ROOT> tag, this tag allows you to specify arbitrary roots of an expression. The radix comes first, and is separated from the radicand by the <OF> tag. These tags are used only within the <MATH> element.

Attributes:

Required

Optional

Example:

```
<ROOT>3<OF>1 + x</ROOT>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Object

compatibility

<object>

Usage:

The OBJECT tag (formerly the INSERT tag) provides a means to insert objects such as ActiveX™ controls and OLE document objects into HTML files. At the time of writing this was supported by Microsoft Internet Explorer 3.0. Examples of Active controls can be found on the Microsoft Internet Explorer site.

The term object is used to describe the things that people want to place in HTML documents, but other terms for these things are: controls, components, applets, plug-ins, media handlers, etc. Related tags are PARAM, SCRIPT and ALIAS.

The OBJECT tag is allowed to contain headers, paragraphs, lists, form elements and even arbitrarily nested OBJECT.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

style - specifies rendering information about the element

dir [LTR | RTL] - Refers to the direction characters are written. (Left to right, right to left) When applied to TABLE, it indicates the geometric layout of rows (i.e. row 1 is on right if DIR=RTL, but on the left if DIR=LTR) and it indicates a default base direction for any text in the table's content if no other DIR attribute applies to that text.

classid - Used to specify a class identifier for an object.

codebase - (formerly 'code') Specifies a URL referencing where to find the code which implements the object's behaviour. If this URL is insufficient to locate the intended object, when for instance, a file contains the implementations for several classes, the CLASSID may be used to supply a disambiguating class identifier.

data - Specifies a URL referencing the object's properties when using an external file for storing them. Also used for encoded embedded properties. Both of these methods of specifying an object's properties are alternatives to using PARAM tags..

type - Specifies an Internet Media Type (see RFC 1590) for the object's data. The attribute can be used to allow user agents to quickly skip media they don't support, and instead to render the contents of the OBJECT element. It is also useful when loading objects off local drives as it allows the media type to be specified explicitly rather than being derived from the file extension.

align [TEXTTOP | MIDDLE | TEXTMIDDLE | BASELINE | TEXTBOTTOM | LEFT | CENTER | RIGHT] - Determines where to place the object. The ALIGN attribute allows objects to be placed as part of the current text line, or as a distinct unit, aligned to the left, center or right.

width - Gives the suggested width of a box enclosing the visible area of the object. The

width is specified in standard units. Browsers may use this value to scale an object to match the requested width if appropriate.
Smooth scaling a small image to a larger size provides an effective solution to reducing the time needed to download an image, offering better subjective results when compared to color reduction.

height - Gives the suggested height of a box enclosing the visible area of the object. The height is specified in standard units. Browsers may use this value to scale an object to match the requested height if appropriate.

border - Applies to the border shown when the object forms part of a hypertext link, as specified by an enclosing anchor element. The attribute specifies the suggested width of this border around the visible area of the object. The width is specified in standard units. For BORDER=0 no border should be shown. This is normally used when such a border would interfere with the visual affordances presented by the object itself. For instance, the object could render itself as a number of beveled buttons.

hspace - Suggested width of the space to the left and right of the box enclosing the visible area of the object. The width is specified in standard units. This attribute is used to alter the separation of preceding and following text from the object.

vspace - Suggested height of the space to the top and bottom of the box enclosing the visible area of the object. The height is specified in standard units.

usemap - Specifies a url for a client-side image map in the format proposed by Spyglass Inc. This is normally appropriate only for static images.

ismap - When the OBJECT element appears within a hypertext link, this attribute indicates that the server provides an image map, so that mouse clicks should be sent to the server in the same manner as for the IMG element. This is normally appropriate only for static images.

Example:

```
<HTML><HEAD>
<TITLE>Index</TITLE>
</HEAD>
<BODY>
<H1>Index</H1>
<OBJECT CLASSID="clsid:99B42120-6EC7-11CF-A6C7-00AA00A47DD2"
HEIGHT="200" WIDTH="95" ALIGN="left">
<PARAM Name="Caption" Value="CSD Corp.">
<PARAM Name="Angle" Value="90">
<PARAM Name="Alignment" Value="3">
<PARAM Name="Mode" Value="2">
<PARAM Name="FrColor" Value="8388608">
<PARAM Name="FontName" Value="Albertus Medium, Arial">
<PARAM Name="FontSize" Value="40">
<PARAM Name="FontBold" Value="1">
</OBJECT>
</BODY><HTML>
```

Option

compatibility

<option></option>

Usage:

The option tag can only occur within a Select element. Each option element represents one choice. The first option will be selected by default, unless a SELECTED attribute is present on any of the <OPTION> elements. The content of the option element is presented to the user to represent the option. It is used as a returned value if the VALUE attribute is not present.

(See Also: Form Assistant and Form Field Assistants)

Attributes:

Required

Optional

selected - [SELECTED] - Indicates that this option is initially selected.

value - indicates the value to be returned if this option is chosen. The field value defaults to the content of the <OPTION> element.

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 shape - used for graphical menus to specify the region of the background image to be associated with this option. It uses the same definition as for the anchor element.

3 **disabled** [DISABLED]- Used to create a "read-only" selection that can't be modified by the user. The option may 'look' disabled e.g. by graying out the text, etc.

3 **error** - This attribute specifies an error message explaining why the option is inappropriate. When this attribute is missing, the option can be assumed to be ok.

Example:

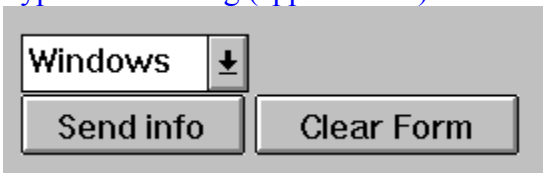
```
<HTML>
<HEAD>
<TITLE>Example Form</TITLE>
</HEAD>
<BODY><FORM ACTION="http://www.somewhere.com/cgi-bin/mailscript"
METHOD="POST">
<P><SELECT NAME="Operating System">
<OPTION>DOS</OPTION>
<OPTION>Mac</OPTION>
<OPTION SELECTED="SELECTED">Windows</OPTION>
<OPTION>Unix</OPTION>
</SELECT>
```

```
<BR><INPUT TYPE="SUBMIT" VALUE="Send info" NAME="Send info">
<INPUT TYPE="RESET" VALUE="Clear Form">
</P></FORM></BODY></HTML>
```

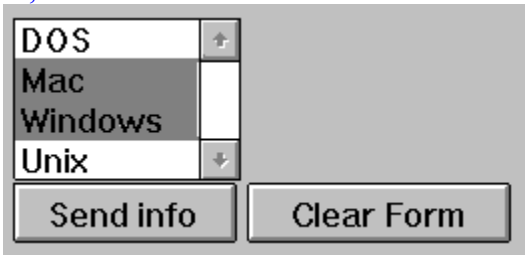
or

```
<HTML>
<HEAD>
<TITLE>Example Form</TITLE>
</HEAD>
<BODY><FORM ACTION="http://www.somewhere.com/cgi-bin/mailscript"
METHOD="POST">
<P><SELECT NAME="Operating System" MULTIPLE>
<OPTION VALUE="DOS">DOS</OPTION>
<OPTION VALUE="Mac">Mac</OPTION>
<OPTION VALUE="Windows">Windows</OPTION>
<OPTION VALUE="Unix">Unix</OPTION>
</SELECT>
<BR><INPUT TYPE="SUBMIT" VALUE="Send info" NAME="Send info">
<INPUT TYPE="RESET" VALUE="Clear Form">
</P></FORM></BODY></HTML>
```

Typical Rendering (approximate):



or, with the MULTIPLE attribute:



Options Menu

Options|Font...
Opens the font selection dialogue box, allowing you to choose a new font for the active window. All other new and previously opened windows will use the font selected in the Preferences Dialogue.
Keyboard Equivalent: Alt+O,F

Options|Preferences...
Opens the Preferences Dialogue where you can set your preferences for various items such as your browser, preferred dtd, name, e-mail address, etc.
Keyboard Equivalent: Alt+O,P

Options|Header Info...
Header information (META tags) can be generated automatically based on specified settings and/or several predefined 'macros', perfect for expiry dates, author name, version number, keywords, etc.
Keyboard Equivalent: Alt+O,H

Options|Show Toolbar
Keyboard Equivalent: Alt+O,R
Displays the Tool Bar. Selecting this menu item when the tool bar is showing will hide it again. The toolbar can float or be docked to the top, bottom, or side of the Webber window.

Options|Show Tagbar
Displays the customizable Tag Tool Bar. Selecting this menu item when the tagbar is showing will hide it again. Tags can be added to this tabbed, floating tool bar in Options|Edit Tagbar. The tagbar can also be docked to the top or bottom of your screen.
Keyboard Equivalent: Alt+O,S

Options|Show Structure
The document structure viewer shows an outline structure of your document. From it you can follow links to open local linked files, and create new links for anchors and images, or you can pass a tag to the tag assistant for editing.
Keyboard Equivalent: Alt+O,T

Options|Edit Tagbar
Opens the Edit Tagbar dialogue box where you can place any number of tags on any number of tabbed 'pages'.
Keyboard Equivalent: Alt+O,E

Options|Custom Bits
Custom Bits are user-definable 'bits' of text, tag combinations, entities etc. Shortcut keyboard combinations can be assigned to both custom bits and to tags.
Keyboard Equivalent: Alt+O,B

Options|Check Spelling
Opens the spell checker and checks the spelling of the content of your document. Dictionaries are available in several languages.
Keyboard Equivalent: Alt+O,C

Ordered List

compatibility

**
**

Usage:

HTML supports several types of lists which may be nested within themselves or each other. The ordered list is similar to the unordered list except that paragraphs will be numbered rather than bulleted. It contains list items which usually result in left aligned, numbered items.

In HTML 3.0, the addition of several attributes allows you to specify what number to start the number sequence.

Attributes:

Required

Optional

compact - [COMPACT] - suggests that the list be rendered with a minimum of white space, not all browsers make use of this attribute.

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

3 **continue** [CONTINUE] - Specifies that the the sequence number should not be restarted i.e. continue where previous list left off.

3 **seqnum -** Sets the starting sequence number for the first item, e.g. <OL SEQNUM=5>

N E **type** [A | a | i | I | 1] - Used to change the numbering type of a list item.

N E **start** - Does the same thing as seqnum (above - HTML 3.0) Converted based on TYPE before display. Thus START=5 would display either an 'E', 'e', 'V', 'v', or '5' based on the TYPE tag.

Example:

```
<HTML><HEAD>
<TITLE>Index</TITLE>
</HEAD>
<BODY>
<H1>Index</H1>
<OL>
```

```
<LI>Ch. 1</LI>
<LI>Ch. 2</LI>
<LI>Ch. 3</LI>
<LI>Ch. 4</LI>
</OL>
</BODY><HTML>
```

Typical Rendering (approximate):

Index

1 Ch. 1
2 Ch. 2
3 Ch. 3
4 Ch. 4

Over

compatibility

<over>

Usage:

Used when you want a line between numerator and denominator. This line is suppressed with the alternative <ATOP> tag. This tag is used only within the <BOX> element in a <MATH> element.

Attributes:

Required

Optional

Example:

```
{(<left>1+x<over> sin x <right>)}
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Overlay

compatibility

<overlay>

Usage:

Overlays provide for an effective use of caching - small changes to a figure in a subsequent document incur only the penalty of downloading the overlays and not the larger base figure.

Attributes:

Required

src - Used to specify the url of the image.

Optional

units [EN | PIXELS] - Specifies the choice of units for width and height. Pixels are the default.

imagemap - Used to specify a URI for processing image clicks and drags.

width - Used to specify the desired width in pixels or en units (according to the value of the UNITS attribute). User agents may scale the figure image to match this width.

height - Used to specify the desired height in pixels or en units (according to the value of the UNITS attribute). User agents may scale the figure image to match this height.

x - Used to specify the X offset from the top left corner of the base image. X increases to the right, and is given in the units specified.

y - Used to specify the Y offset from the top left corner of the base image. Y increases downwards, and is given in the units specified.

md - Specifies a message digest used when you want to be sure that a linked graphic, specified by the SRC attribute, is the same one intended, and hasn't been changed in any way.

Example:

```
<HTML><HEAD>
<TITLE>Webber (TM)</TITLE>
</HEAD>
<BODY>
<H1>Welcome to the Webber Home Page</H1>
<FIG SRC="spider.gif">
<OVERLAY SRC="map.gif">
<CAPTION><I>Webber</I> - HTML Editor Extraordinaire!</CAPTION>
<P>Our small arachnid friend tips his hat to you!
<CREDIT>Kathryn M. Dickson</CREDIT> </FIG>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Parameter

compatibility

<param>

Usage:

The PARAM element is used with the OBJECT tag and allows a list of named property values (used to initialize a OLE control, plug-in module etc.) to be represented as a sequence of PARAM elements. Note that PARAM is an empty element.

Attributes:

Required

name - Defines the property name. The case sensitivity of the name is dependent on the code implementing the object.

Optional

value - Used to specify the property value. It is an opaque character string whose meaning is determined by the object based on the property name. Note that CDATA attribute values need characters such as & to be escaped using the standard SGML character entities, e.g. & for "&". It is also essential to escape the > character to defend against incorrect handling by many existing browsers (use >).

valueref - Used when the property is itself an object. A distinct attribute is needed as in some cases the property type cannot be deduced from the property name. VALUEREF typically provides a URL based reference to an ALIAS element (see below) that defines the object itself. VALUEREF can also be used to specify an object directly. For example valueref=foo.gif. Another possibility is to use inline data with the "data:" URL scheme. Both of these options save having to include an associated ALIAS element.

type - The TYPE attribute is required when the "data:" URL scheme is used. It specifies the media type for the data stream and allows the user agent to decode the stream, e.g. to pick out an embedded class identifier.

Example:

```
<HTML><HEAD>
<TITLE>Index</TITLE>
</HEAD>
<BODY>
<H1>Index</H1>
<OBJECT CLASSID="clsid:99B42120-6EC7-11CF-A6C7-00AA00A47DD2"
HEIGHT="200" WIDTH="95" ALIGN="left">
<PARAM Name="Caption" Value="CSD Corp.">
<PARAM Name="Angle" Value="90">
<PARAM Name="Alignment" Value="3">
<PARAM Name="Mode" Value="2">
<PARAM Name="FrColor" Value="8388608">
<PARAM Name="FontName" Value="Albertus Medium, Arial">
<PARAM Name="FontSize" Value="40">
<PARAM Name="FontBold" Value="1">
</OBJECT>
</BODY><HTML>
```

Path Options

Use these settings to set the defaults for file name insertion when you use a browse button to select a file. The options for HREF and SRC are held separately.

Path (default - file name)
File Name - only the name of the file will be inserted. i.e. `new.html`
Relative - the relative path to the file and the file name will be inserted. i.e. `../webber/new.html`
Absolute - the drive name and the entire path to the file will be inserted along with the file name. i.e. `D:/webber/new.html`

Case Conversion (default - none)
None - will insert the file as is
Lowercase - will convert the path and file names to lowercase
Uppercase - will convert the path and file names to uppercase

Convert Separator to Unix (default - on)
When checked, back slashes used on Windows/DOS systems will be converted to forward slashes for use on Unix systems. Most browsers can understand forward slashes even on Windows systems so it will still be possible to test files locally even with this switch enabled.

Person

compatibility

<person></person>

Usage:

This tag is is used for names of people to allow them to be extracted automatically by indexing programs. It is used within a 'container' tag such as a paragraph.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Working Committee</TITLE>
</HEAD>
<BODY>
<H1>Committee Members</H1>
<P>The members of this committee are:
<BR><PERSON>Jane Smith</PERSON>
<BR><PERSON>John Doe</PERSON>
<BR><PERSON>Jan Green</PERSON>.</P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Plain Text

compatibility

<plaintext>

Usage:

The Plain Text element is used to terminate the HTML instance and to indicate that what follows is not SGML which does not require parsing. Instead, an old HTTP convention specified that what followed was an ASCII (MIME "text/plain") body. Its presence is an optimization. There is no closing tag.

This tag has been deprecated in HTML 2.0 and is already obsolete in HTML 2.0.

Attributes:

Required

Optional

Example:

```
<HTML><HEAD>
<TITLE>Webber Reference</TITLE>
</HEAD>
<BODY>
<H1>Webber Reference</H1>
<HR>
<P>To open the help file, select <KBD>Help|Contents</KBD> from the
menu.</P>
</BODY></HTML>
<PLAINTEXT>
0001 This is line one of a long listing
0002 This is line two of a long listing
```

Typical Rendering (approximate):

Webber Reference

To open the help file, select `Help|Contents` from the menu.

```
0001 This is line one of a long listing
0002 This is line two of a long listing
```

Preferences Dialogue Box

Preferences|Date

The time and date formats set here will effect the times and dates in the auto-stamp feature, as well as the time and date inserted by clicking on the status line.

Date Formats:

Windows Long (default) - uses the format specified in your Windows set-up for long dates - i.e. December 22, 1995.
Windows Short- uses the format specified in your Windows set-up for short dates - i.e. 22/12/95 or 12/22/95
Custom - allows you to specify a date format different than the windows standard settings. Choose the elements of the date from the drop-down lists and fill in the desired separators in the entry boxes. The example at the bottom of the screen will format itself to your settings so that you can ensure that you like them.

Time Formats:

Windows Standard - uses the time format from the Windows set-up.
Custom - allows you to specify a different format from the one you have set in your Windows set-up. Choose the time format you want from the drop-down list and fill in the desired separator and AM/PM flags in the space provided. The example at the bottom will show you how the time will look.

Preferences|Editor

Default Font:

Sets the default font for use when opening documents. The font for an individual document can be changed from the Options|Font menu. Font selections affect the entire document. There is no character level formatting at this time (although it is a feature planned for a future version of Webber™.)

Word Wrap:

When enabled, word wrap will cause the text in the windows to wrap automatically when you are typing and reach the end of a line, and when a window is resized.

Drag and Drop Editing:

When enabled, drag and drop editing allows you to move and copy a selection by dragging it to a new location with your mouse. Holding down the Ctrl key during the dragging action will copy the selection to the new location instead of moving it.

Syntax Highlighting:

When enabled, syntax highlighting changes the colour of tags and attributes to the colour you have defined here. This feature is extremely helpful for picking out your content in complex tag structures such as tables, and for immediately identifying missing angle brackets.

Tag Protect:

Tag protection helps stop accidental deletions and overwrites of tags. When enabled the angle brackets will be protected from the backspace key, the delete key from the outside of a tag. If you position your cursor inside a tag you will be able to remove the angle brackets.

Long File Names:

Long file name support allows users of Windows 95 and Windows NT to use long file and directory names.

Roll Tag Assistant:

The roll Tag Assistant setting allows you to specify whether or not you want the Tag Assistant to automatically roll up when you are not using it and roll down when you click on it or right-click on the tagbar.

Maximize Edit:

This option allows you to specify that you always want edit windows to open maximized.

Single Close:

This option allows you to specify that you want the 'close' button on the window to close only one file at a time, instead of all open files.

Auto Tag:

Paragraph Separator - This setting tells the auto tag feature whether to use hard returns or blank lines to determine the beginning of new paragraphs.

Max. Heading Characters - This setting tells the auto tag feature the maximum number of characters on a line to tag as a heading.

Preferences|Forms

Auto-Label Position:

The form assistant and form field assistants use these settings to determine the position of labels that are automatically generated for fields in a form.

If the setting is none, a label will not be generated but can be added manually in the form assistant, or directly into the document. If the setting is before or after, the label will be based on the name of the field but can be edited in the form field assistants before being placed in the document. 'Before' means that the label will be inserted before the field, 'after' means the label will be inserted after the field.

Preferences|HTML

Active DTD - allows you to select the dtd. The dtd that you choose determines the tags that are available on the tagbar and in tag assistant, as well as what tags are allowed when you validate. You can also select a dtd on-the-fly by right-clicking the dtd button on the status line.

Add Doctype Automatically - adds the doctype information required for HTML compliance if it is missing. This setting affects only documents without the doctype as they are opened. A doctype looks like this: `<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">`

Add Meta to New HTML - adds the header (docinfo) information to New HTML documents. The header info can be used to specify document expiry dates, keywords, etc.

Browser Definition:

Allows you to configure all of your browsers so that you can select between them when you preview your documents. Your default browser will be used when you click on the preview button. You can select another browser by right-clicking on the preview button and selecting the desired browser from the pop-up list. The pop-up list will display the names of the browsers as you have configured them below.

To Set Up Your Browsers:

1 Place your cursor in the 'Name' cell and type a unique name for your browser (i.e. Netscape 2.0).
2 Select the browser type from the pop-up list in the 'Type' column. Some browsers support inter-application communication through either OLE or DDE. If your browser is on this list Webber should be able to run the program, and load and reload documents. Some browsers also support being called to the front when a document is loaded or reloaded.
3 Use the browse (...) button in the path column to enter the full path and program name for your browser.
4 To set your default browser, click on the left-most cell of the browser row. The default browser will have an **'x'** beside it in this cell.
5 To add a new browser, click the 'add' button. To remove a browser, select it with your mouse and click the 'remove' button.

Preferences|Tags

The Tags page is included in the preferences so that you can:

1 Set the order that tags appear on the drop-down list on the Tag Assistant. The tags will appear there in the same order as they appear on this list.

- To change the order of a tag, click on the left hand column, hold the mouse button down and drag the tag to its new location. Release the mouse button to drop the tag.

2 Set certain tags to be inactive and not to appear on these lists at all. These are good techniques to use in order to keep the tag lists short and relevant to you.

- To set a tag to inactive, single-click on a cell in the left hand column. Inactive tags display a 'x' in this column. You can remove the x by single-clicking on the cell again.

3 Set the case of tags and attributes inserted by any of Webber's features to either upper case or lower case. Tags and attributes can be set separately.

- Check the appropriate boxes in the lower right corner of the dialog box.

4 Set-up keyboard shortcuts for tags.

- Select a tag. Click in the 'shortcut' edit box, then press the key combination you would like to use for a tag. The Ctrl, Alt and Shift check boxes are gray because you can not select them directly. Their status will be set automatically based on the keys you press.

Preferences|User Info.

Author Name, e-mail, Copyright:

If these items are enabled for the stamp below, the text entered into these boxes will be inserted with the stamp. You can create automatic hyperlinks by including HTML tags around the text. For example, you could place a hyperlink MAILTO around your e-mail address, or a link to your home page from your company name. (see the example)

Time Zone:

Select the appropriate time zone for your location from the drop-down list. This option is only important if you are using one of the date macros in the header information.

Auto Stamp:

New Docs. - new HTML documents created by selecting File|New HTML from the menu will have the stamp added automatically if this check box is checked.

Converted Docs. - new HTML documents created by converting an existing text document will have the stamp added automatically if this check box is checked.

Info. in Stamp.

Any items checked off here will be inserted automatically at the cursor position when you select HTML|Stamp from the menu. If you choose to include the date and/or time updated, these items will be refreshed whenever a document containing them is saved as long as the comments inserted with the stamp are not removed or modified.

Use Stamp File

Specify a custom file to be used as the auto-stamp and inserted stamp instead of the user info specified above. You can also create a custom stamp file that will be automatically updated by Webber.

Preformatted Text

compatibility

<pre></pre>

Usage:

The preformat tag allows an author to format text on screen in the way that they wish it to be displayed. Unlike other paragraph tags, extra spaces, carriage returns, line breaks etc. will be displayed as they were typed. While text mark-up such as emphasis can be used within the preformat tag, paragraph tags such as <P>, <ADDRESS>, and headings should not be used.

The font and indentation used is up to the browser but usually a monospaced font will be used.

Attributes:

Required

Optional

width - specifies the maximum number of characters for a line

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

Example:

```
<HTML><HEAD>
<TITLE>Webber Home Page</TITLE>
</HEAD>
<BODY>
<H1>Welcome to the Webber Home Page</H1>
<P>This is the home page for the hot, easy to use, HTML editor -
Webber.</P>
<P>Webber was designed as a tool for HTML authors who want to see and
control what is happening in their document at the tag level.</P>
<PRE>
The preformat tag allows the entry of extra spaces     like
this      so that one can line up columns.

It also pays attention to carriage returns.
<STRONG>
For example:</STRONG><EM>
       Column 1  Column 2   Column 3</EM>
Row A     A1        A2         A3
Row B     B1        B2         B3
```

```
Row C     C1        C2         C3 </PRE></BODY>
</HTML>
```

Typical Rendering (approximate):

Welcome to the Webber Home Page

This is the home page for the hot, easy to use, HTML editor - Webber.

Webber was designed as a tool for HTML authors who want to see and control what is happening in their document at the tag level.

```
The preformat tag allows the entry of extra spaces like     this
so that one can line up columns. It uses a mono-space font to do this.

It also pays attention to carriage returns.
```

For example:

```
       Column 1  Column 2   Column 3
Row A     A1        A2         A3
Row B     B1        B2         B3
Row C     C1        C2         C3
```

Quick Tag Reference

Index to Tag Reference Topics

Note: The following chart shows which tag set each HTML tag belongs to. Explorer tags are supported by Internet Explorer 3.0+ (some may also be supported by IE2.0). Netscape tags are supported by Netscape 2.0+.

Tag Name	HTML 2	HTML 3	HTML 3.2	Explorer	Netscape
a	Y	Y	Y	Y	Y
abbrev		Y			
above		Y			
acronym		Y			
address	Y	Y	Y	Y	Y
alias		Y			
app					Y
applet			Y	Y	Y
area			Y	Y	Y
array		Y			
atop		Y			
au		Y			
b	Y	Y	Y	Y	Y
banner		Y			
bar		Y			
base	Y	Y	Y	Y	Y
basefont					Y
below		Y			
bgsound				Y	
big		Y	Y	Y	Y
blink					Y
blockquote	Y	Y	Y	Y	Y
body	Y	Y	Y	Y	Y
box		Y			
bq		Y			
br	Y	Y	Y	Y	Y
bt		Y			
caption		Y	Y	Y	Y
center			Y	Y	Y
choose		Y			
cite	Y	Y	Y	Y	Y
code	Y	Y	Y	Y	Y
col		Y			
colgroup		Y			
credit		Y			
dd	Y	Y	Y	Y	Y
ddot		Y			
del		Y			
dfn		Y	Y	Y	
dir	Y	Y	Y	Y	Y

div		Y	Y	Y	Y
dl	Y	Y	Y	Y	Y
dot		Y			
dt	Y	Y	Y	Y	Y
em	Y	Y	Y	Y	Y
embed					Y
fig		Y			
fn		Y			
font			Y	Y	Y
form	Y	Y	Y	Y	Y
frame				Y	Y
frameset				Y	Y
h1	Y	Y	Y	Y	Y
h2	Y	Y	Y	Y	Y
h3	Y	Y	Y	Y	Y
h4	Y	Y	Y	Y	Y
h5	Y	Y	Y	Y	Y
h6	Y	Y	Y	Y	Y
hat		Y			
head	Y	Y	Y	Y	Y
hr	Y	Y	Y	Y	Y
html	Y	Y	Y	Y	Y
i	Y	Y	Y	Y	Y
img	Y	Y	Y	Y	Y
input	Y	Y	Y	Y	Y
ins		Y			
isindex	Y	Y	Y	Y	Y
item		Y			
kbd	Y	Y	Y	Y	Y
lang		Y			
left		Y			
lh		Y			
li	Y	Y	Y	Y	Y
link	Y	Y	Y	Y	Y
math		Y			
listing	Y		Y	Y	Y
map			Y	Y	Y
marquee				Y	
menu	Y	Y	Y	Y	Y
meta	Y	Y	Y	Y	Y
nextid	Y	Y			
nobr				Y	Y
noframe				Y	Y
note		Y			
object				Y	
of		Y			
ol	Y	Y	Y	Y	Y
option	Y	Y	Y	Y	Y
over		Y			

overlay		Y			
p	Y	Y	Y	Y	Y
param			Y	Y	Y
person		Y			
plaintext	Y		Y	Y	Y
pre	Y	Y	Y	Y	Y
q		Y			
range		Y			
right		Y			
root		Y			
row		Y			
s		Y		Y	
samp	Y	Y	Y	Y	Y
script				Y	Y
select	Y	Y	Y	Y	Y
small		Y	Y	Y	Y
spot		Y			
sqrt		Y			
strike			Y	Y	Y
strong	Y	Y	Y	Y	Y
style		Y	Y	Y	
sub		Y	Y	Y	Y
sup		Y	Y	Y	Y
t		Y			
tab		Y			
table		Y	Y	Y	Y
tbody		Y			
td		Y	Y	Y	Y
textarea	Y	Y	Y	Y	Y
textflow			Y		
tfoot		Y			
th		Y	Y	Y	Y
thead		Y			
tilde		Y			
title	Y	Y	Y	Y	Y
tr		Y	Y	Y	Y
tt		Y	Y		
u		Y		Y	
ul	Y	Y	Y	Y	Y
var	Y	Y	Y	Y	Y
vec		Y			
wbr				Y	Y
xmp	Y		Y	Y	Y

Quick Tips

Q *Why are some of the tag buttons on my tagbar grayed out?*
A Which tags that are available on the tagbar and on the tag assistant are determined by the currently selected dtd. For example if you have assigned FONT to the tagbar but have selected HTML 2 as your dtd, the FONT tag will be grayed out. To enable the FONT tag in this case, switch to the Netscape or Explorer dtd.

Q *How do I set a background color for my page?*
A Open the tag assistant. Place your cursor inside the BODY start tag in your document. Make sure you have Netscape or Explorer selected as your dtd. Click on the tag assistant. The background color is set by using the BGCOLOR attribute. Click beside BGCOLOR in the list of attributes. A properties button [...] will appear on the right. Click this button and select a color from the color picker. Return to the tag assistant and click the + button to insert the changed body tag. (*Hint: you can set text colors, font colors, and background colors for tables in the same way.*)

Q *How do I select a dtd? (And what is a dtd anyway?)*
A DTDs are files that describe what tags and attributes are allowed in a document, and where they can be legally used. Webber uses dtds for validation, and for determining what tags and attributes are available based on which dtd you have selected. If you want to be sure that all browsers can view your document (even text-based ones), you should use the HTML 2 dtd. If you want to use some of the new features of HTML, such as background colors, tables, etc. you should use the HTML 3.2, Explorer or Netscape dtd. To select a dtd, right-click the dtd button at the right of the status line and choose a new one from the pop-up menu.

Q *What help can Webber give me with complex tags like tables and frames?*
A Webber has a host of 'Assistants' to help you insert complicated tag structures. Check out the table assistant, the frames assistant, the list assistant and the form assistant.

Q *Why doesn't the trademark (etc.) entity or the TAB (etc.) tag work in my browser?*
A Although Webber tries to help you create documents that are valid for the selected tag set, some entities and tags are not supported by all browsers. For example, although there is an entity for the trademark symbol defined for HTML 3, none of the major browsers support it yet.

Quick-Tag Buttons

Some common tags are available as buttons on the Webber tool bar. They are:

 Anchor - Opens the Anchor Assistant dialogue box which allows you to quickly add an anchor to your document and set the common attributes.
Keyboard Equivalent: Alt + T, A
Right Click?: No

 Image - Opens the Image Assistant dialogue box which allows you to quickly add a picture or graphic to your document and to set its common attributes.
Keyboard Equivalent: Alt + T, I
Right Click?: No

 List - Opens the List Assistant dialogue box which allows you to quickly add a lists and list items to your document.
Keyboard Equivalent: Alt + T, L
Right Click?: No

 Table - Opens the Table Assistant dialogue box which allows you to set the size (number of rows and columns), attributes and caption for a table. From the Table Assistant you can open the Table Editor where you can set row and column spanning and enter your data into the cells of your table.
Keyboard Equivalent: Alt + T, I
Right Click?: No

 Entity - Opens the Entity Assistant dialogue box which allows you to select character entities from a list, or pick one from a graphical representation.
Keyboard Equivalent: Alt + T, E
Right Click?: No

 Tag Assistant - Opens the Tag Assistant dialogue box which allows you to select tags from a list, and shows you what attributes can be set for the tag.
Keyboard Equivalent: Alt + T, T
Right Click?: No

 Frame Assistant - Opens the Frame Assistant dialogue box which allows you to design frame layouts.
Keyboard Equivalent: Alt + T, F
Right Click?: No

Quote

compatibility

<q></q>

Usage:

This tag is used for a short quotation. It is would be shown enclosed in quotation marks as appropriate to the language context. For English these would be matching double or single quotation marks, alternating for nested quotes. The language context is set by the LANG attribute. This tag is used inside a 'container' tag such as a paragraph.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Webber Home Page</TITLE>
</HEAD>
<BODY>
<H1>Welcome to the Webber Home Page</H1>
<P>Webber was designed as a tool for HTML authors who want to see and
control what is happening in their document at the tag level.</P>
<P>Everyone who tries Webber says, <Q>I love it!</Q>. Give it a try
today!</P>
</BODY></HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Radio Buttons Assistant

Radio Buttons are input fields that allow the user to 'check' a field by clicking on it with their mouse in a form. Radio buttons are usually used in groups. One set of radio buttons is actually several input fields sharing the same name and type=radio attribute. The Radio Button Assistant is meant to help you place a set of radio buttons in a form with the needed attributes set. The Assistant is available from the HTML|Control|Radio Button... menu and from the Form Assistant.

The attributes usually used with radio buttons are **name**, **value** and **checked**. Several radio buttons sharing the same name attribute will work as a unit - turning one button on will turn another off. The name is returned when the form is submitted as the first in the name/value pair. The value is returned as the second in the name/value pair. The return is made for only the button that is checked or 'on'. i.e. A three button set of radio buttons, all with a name of "Operating System" would have three buttons with values of "Windows", "Mac", and "Unix" and would return : "Operating System=Windows" if the radio button with the value of "Windows" is checked. The checked attribute is used to set the default state of the button when the form is loaded.

To Use:

1 Open the Radio Button Assistant from the Form Assistant Dialogue by choosing Text Area from the drop-down list or from the menu by selecting HTML|Controls|Radio Button... If you are using the latter method be sure to first position your cursor where you want the radio buttons inserted.

2 Enter the name of the radio buttons. This name will be used for all buttons in this set. If you have auto-label enabled in Preferences|Forms, the label will fill in with the name value. This label will be used for the set of buttons. You can edit this if you want.

3 Double click on a cell in the type column to add a radio button, line break or label.

If you select a line break, no further action is required since it does not have any attributes.

If you select a label, a dialogue asking you for the text of the label will appear. A label is not an HTML tag - it is optional content that appears in your form to identify the radio button to the user. You can have the Form Assistant generate the labels for you by enabling the Auto-Label feature in the Preferences|Forms. The position of the labels can also be set there.

If you select a radio button, a radio button dialogue will appear.

4 Enter the value you want returned when the radio button is checked.

5 Edit the label text if you want. If you do not have the auto-label feature enabled, you will not be able to enter into this field.

6 When you are finished, click the Ok button to insert the radio button into the Radio Assistant. Click the Cancel button to leave without inserting the button.

7 Click in the left hand column to turn the default state of the button to checked. A 'x' will appear in this column when this attribute is set.

8 Radio buttons, line breaks and labels can be reordered by clicking on a row of the left column, holding down the mouse button, dragging the row to its new location and releasing the mouse button.

9 Items can be deleted by clicking in the first cell of the row and clicking the delete button. Radio buttons and labels can be edited by clicking on the first cell of the row to be edited and clicking the edit button.

10 When you have added all of the radio buttons, labels and line breaks for the radio button set, click the Ok button to insert the radio buttons into the form (if you are using the Form Assistant) or your document. Click the Cancel button to leave the Radio Button Assistant without inserting the set of radio buttons.

Range

compatibility

<range>

Usage:

This tag is used to mark a range of the document, for example for highlighting regions of the document matching some search criteria, or which are the subject of an annotation etc. i.e. `<RANGE CLASS=Search FROM=spot01 UNTIL=spot02>` It is used within the <HEAD> tag.

The FROM and UNTIL attributes specify positions in the document using SGML identifiers. Most tags in the document body can use the ID attribute for identification. The <SPOT> tag is also useful, as it allows search software etc. to insert IDs at random places.

Attributes:

Required

from - IDRef - References an SGML identifier - NAME token - for an element in the document body. It identifies the start of the marked range.

until - IDRef - References an SGML identifier - NAME token - for an element in the document body. It identifies the end of the marked range.

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
<RANGE CLASS=Search FROM="spot01" UNTIL="spot02">
</HEAD>
<BODY>
<H1>Other information about HTML.</H1>
<SPOT ID="spot01"><P>There is lots of information to be had about
<ACRONYM>HTML</ACRONYM> on the Internet.</P><SPOT ID="spot02">
</BODY>
</HTML>
```

Typical Rendering (approximate):

Not applicable.

Recommended
Uses only the strict, recommended aspects of the HTML dtd when validating a document. If you select this option, your document will only validate if it meets the strictest requirements of the selected dtd. Using this setting when creating new documents will ensure that your documents remain compliant even as the specifications change.

Registration Information

Purchasing a license for Webber will give you with the right to use the software. You must purchase a license if you choose to use Webber after a trial period of 30 days. There are other great reasons to register too! If you do not purchase a license, you are not allowed to keep or use Webber (see License Agreement).

Webber is great value for your money, only US$30.00 / CAN$40.00! To place an order... (remember to provide your e-mail address so that you get notified of updates)...

- **ORDERING BY CHEQUE**

Contact Cerebral Systems Development Corp. by mailing the Cheque Order Form with a US or Canadian Cheque or $US money order. You can find us on the World Wide Web at : http://www.csdcorp.com/. We will have current information on how to register at that site.

- **CREDIT CARD ORDERS ONLY**

1 You can pay by credit by ordering through Compuserve (GO SWREG, Product Id#7026).

2 You can register Webber with your credit card via secure server on the Internet. This is a service that will be offered by PsL (below), the link to their order page is available on our web site at http://www.csdcorp.com/order.htm.

3 You can order with MC, Visa, Amex, or Discover from Public (software) Library by calling: (Reference Product #14129)
1.800.2424.PsL [1.800.242.4775] or +1 713.524.6394
or by FAX to +1 713.524.6398
or by Compuserve E-mail to 71355,470.
You can also mail credit card orders to PsL at P.O. Box 35705, Houston, TX 77235-5705. Use the Credit Order form for faxed or mailed credit card orders. For e-mail orders send an e-mail message containing all of the information found on the credit order form.

THE ABOVE NUMBERS ARE FOR CREDIT CARD ORDERS ONLY.
THE AUTHOR OF THIS PROGRAM CANNOT BE REACHED AT THESE NUMBERS.

Any questions about the status of the shipment of the order, registration options, product details, technical support, volume discounts, dealer pricing, site licenses, non-credit card orders, etc., must be directed to webber@csdcorp.com or FAX +1 416.651.7685.

To insure that you get the latest version, PsL will notify us the day of your order and we will ship the product directly to you.

- **SITE LICENSES**

For information on site licenses, please contact CSD Corp. at admin@csdcorp.com or fax to +1 416.651.7685.

- **REFUNDS**

Since you have ample opportunity to try Webber before you register, we do not provide refunds. So please take advantage of the thirty day evaluation period to be certain that Webber meets

your needs.

When ordering, please follow these guidelines:

- Print or type clearly. Be sure to give us all the details such as your city, state/province, and country that you live in. This is especially important for foreign orders, we may not be familiar with the addressing conventions of your country/state. For fax numbers, please include your country code, and/or make sure that the country is spelled out completely. Please do not use abbreviations such as DN, or FR - we may not be familiar with the short form.
- We can only accept US or Canadian checks, credit card cheques and international bank drafts or money orders in US funds. All checks must be drawn on a US or Canadian bank. Use the Cheque Order Form and mail with your cheque to CSD Corp. For Credit Card orders you must order from PsL by phone or use the Credit Order Form and mail or fax to PsL. Check our Web site for up-to-date information on ordering. You can also pay by credit if you register through Compuserve (GO SWREG, Product Id#7026)

Requirements, Setup & Installation

System Requirements:

- Windows 3.1 or higher
- 4 MB RAM minimum
- Approximately 2 MB disk space
- World Wide Web browser *(optional - works best with the newest versions of Netscape or Mosaic - requires OLE2 - available on the trouble-shooting page of our web site.)*

Setup & Installation:

- Webber comes with a complete installation program that copies and decompresses all files to the appropriate places.
- From Program Manager choose File|Run. Type `A:\webinst` (or B:\).
- Specify a directory for the program files and the Webber install does the rest. Be sure to read the readme.txt file for any changes or additions to these instructions.
- The Webber install program will create a program group and/or program item in your Program Manager.

To reduce the disk space required:

- Webber will install the i4isas.dll in the Webber program directory. You may move it to your Windows system directory If you already use this files with another program, Webber will happily load it from there.

To Remove Webber™ from your system:

- Run the uninstall program that was installed with Webber.

Reset Button Assistant

Reset Buttons are input fields that appear on a form as a button. Their purpose is to clear the contents of the form and reset all of the fields to their initial (default) state. The Reset Button Assistant is meant to help you place a reset button in a form with the needed attributes set. The Assistant is available from the HTML|Control|Text Field... menu and from the Form Assistant.

There is usually only one attribute used with the reset button - **value**. The value is used as the label on the button.

To Use:

1 Open the Reset Button Assistant from the Form Assistant Dialogue by choosing Reset Button from the drop-down list or from the menu by selecting HTML|Controls|Reset Button. If you are using the latter method, be sure to first position your cursor where you want the button inserted.

2 Enter the value of the button. This is used as the button label for display on the form.

3 When you are finished, click the Ok button to insert the button into the form (if you are using the Form Assistant) or your document. Click the Cancel button to leave the Reset Button Assistant without inserting the button.

Right

compatibility

<right></right>

Usage:

Used to place a delimiter, which grows to match the height of the expression, after an expression. This tag is used only within the <BOX> element in a <MATH> element.

Attributes:

Required

Optional

Example:

```
{(<left>1+x<over> sin x <right>)}
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Right Click Behaviour

Clicking the right mouse button in a document will bring up a pop-up menu which allows you to select from several actions.

Undo
Multiple level undo - reverses changes made to tags and text in reverse order. Large operations such as pasting over a selection of text are undone in one action.
Keyboard Equivalent: Alt + E, U
Shortcut: Ctrl Z
Button?: Yes

Cut
Cuts the selected text and tags to the clipboard.
Keyboard Equivalent: Alt + E, T
Shortcut: Ctrl X
Button?: Yes

Copy
Copies the selected text and tags to the clipboard.
Keyboard Equivalent: Alt + E, C
Shortcut: Ctrl C
Button?: Yes

Paste
Pastes the pastes the contents of the clipboard at the current caret position.
Keyboard Equivalent: Alt + E, P
Shortcut: Ctrl V
Button?: Yes

Tag Select
Selects the text and tag from the beginning of a start tag to the end of its end tag. **Note: this feature does not work if a tag does not have an end tag. The selection will be made to the end of the document instead.**
Keyboard Equivalent: Alt + E, G
Shortcut: None
Button?: No

Tag Help
Opens this help file at the help topic for a selected tag. Tags may be selected by double-clicking on them.
Keyboard Equivalent: Alt + H, T
Shortcut: None
Button?: No

Validate
Starts the process of testing that your document is compliant with the rules laid out in the HTML specification. Which version of the specification is checked depends on which dtd you have selected.
Keyboard Equivalent: Alt + H, V

Shortcut: None
Button?: No

Root

compatibility

<root></root>

Usage:

This allows you to specify arbitrary roots of an expression. The radix comes first, and is separated from the radicand by the <OF> tag. This tag is used only within a <MATH> element.

Attributes:

Required

Optional

Example:

```
<ROOT>3<OF>1 + x</ROOT>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Row

compatibility

<row></row>

Usage:

This tag is used for rows of <ITEM>s within the <ARRAY> element which is used only in the <MATH> tag.

Attributes:

Required

Optional

Example:

```
<array>
<row><item>a_11_<item>a_12_<item>&cdots;<item>a_1n_
<row><item>&vdots;<item>&vdots;<item>&ddots;<item>&vdots;
<row><item>a_n1_<item>a_n2_<item>&cdots;<item>a_nn_
</array>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

SGML

HTML is defined as an application of ISO Standard 8879:1986 Information Processing Text and Office Systems; Standard Generalized Markup Language (SGML).

SQRT

compatibility

<sqrt></sqrt>

Usage:

Used to draw a square root sign around the contents. This tag is used only within a <MATH> element.

Attributes:

Required

Optional

Example:

```
<SQRT>1 + x</SQRT>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Sample

compatibility

<samp></samp>

Usage:

The sample tag is used to indicate a sequence of literal characters. It is usually rendered in a monospaced font.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Reference</TITLE>
</HEAD>
<BODY>
<H1>Reference</H1>
<HR>
<P>It can be spelled <SAMP>Inquiries</SAMP> or <SAMP>Enquiries</SAMP>,
either is acceptable.</P>
</BODY></HTML>
```

Typical Rendering (approximate):

Reference

It can be spelled `Inquiries` or `Enquiries`, either is acceptable.

Script

compatibility

<script>...</script>

Usage:

The SCRIPT tag is used to embed Visual Basic Script (for Internet Explorer 3.0) and/or Javascript (Netscape & IE3.0). These scripts can be used to control ActiveX controls (in the case of IE3.0), or add java programs to HTML documents.

Javascript event handlers are associated with some tags and are supported by Webber.

Example:

```
<HTML><HEAD>
<SCRIPT LANGUAGE="JavaScript">
<!--  to hide script contents from old browsers
  function square(i) {
    document.write("The call passed ", i ," to the function.","<BR>")
    return i * i
  }
  document.write("The function returned ",square(5),".")
// end hiding contents from old browsers  -->
</SCRIPT>
</HEAD>
<BODY>
<BR>
All done.
</BODY>
</HTML>
```

Typical Rendering (approximate):

The call passed 5 to the function.
The function returned 25.
All done.

Search Menu

Search|Find...
Searches the document for a specified word, phrase or character(s). Type the word(s) you want to search for into the dialog or highlight a word(s), then select Search|Find... and your selection is automatically picked up by the search dialog.
Keyboard Equivalent: Alt + S, F
Shortcut: Ctrl F
Button?: Yes

Search|Find Next
Repeats the most recent search.
Keyboard Equivalent: Alt + S, N
Shortcut: F3
Button?: Yes

Search|Replace...
Searches the document for a specified word, phrase or character(s) and replaces them with another specified word, phrase or character(s). Type the word(s) you want to search for into the dialog or highlight a word(s), then select Search|Replace... and your selection is automatically picked up by the search and replace dialog.
Keyboard Equivalent: Alt + S, R
Shortcut: Ctrl R
Button?: No

Selecting a dtd

Select the dtd that you would like to use by either opening the Preferences and choosing one from the drop-down list, or by right-clicking on the dtd button at the far right of the status line. The current dtd will determine what tags are available on the tagbar and in the tag assistant, as well as what tags and attributes are allowed when you validate.

dtd Support:

HTML 2.0 (R)- provides support for tags specified in the recommended HTML 2.0 specification. This includes forms.

HTML 2.0 - provides support for tags specified in the deprecated HTML 2.0 specification. This includes forms.

HTML 3.0 (R)- provides support for tags specified in the draft HTML 3.0 specification with the recommended setting. This includes tables.

HTML 3.0 - provides support for tags specified in the draft HTML 3.0 specification with the deprecated setting. This includes tables.

Netscape Ext. (R) - provides recommended support for extensions to tags and attributes in the HTML 2.0 and 3.0 specifications made by Netscape Communications. This includes font sizes, frames, additional alignment options for text and images, tables, and background images.

Netscape Ext. - provides deprecated support for extensions to tags and attributes in the HTML 2.0 and 3.0 specifications made by Netscape Communications. This includes font sizes, frames, additional alignment options for text and images, tables, and background images.

Explorer (R) - provides recommended support for extensions to tags and attributes in the HTML 2.0 and 3.0 specifications made by Microsoft. This includes background sounds, marquees, frames, and additional attributes on some tags.

Explorer Ext. - provides deprecated support for extensions to tags and attributes in the HTML 2.0 and 3.0 specifications made by Microsoft. This includes background sounds, marquees, frames, and additional attributes on some tags.

Selection List Assistant

Selection Lists are fields that allow the user to select a value(s) for entry in a form. A selection list is actually a group of options within the select tag. The Selection List Assistant is meant to help you place a list in a form with the needed attributes set. The Assistant is available from the HTML|Control|Text Field... menu and from the Form Assistant.

The attributes usually used with selection list are **name**, **size**, and **multiple**. The name is returned when the form is submitted as the first in the name/value pair. The size is the number of rows displayed on the form. A size of 1 will cause a drop-down list to be displayed. Multiple indicates that more than one value can be selected by the user. The value returned can be either the selection text or the value. If you enter a return value, it will be returned by the form. If no value is specified, the text of the selected option will be returned instead. i.e. A selection field with a name of "operating system" and three options on the list of "Windows", "Mac" and "Unix" would return: "operating system=Windows" if Windows was selected by the user.

To Use:

1 Open the Selection List Assistant from the Form Assistant Dialogue by choosing Select from the drop-down list or from the menu by selecting HTML|Controls|Selection List. If you are using the latter method, be sure to first position your cursor where you want the selection list inserted.
2 Enter the name of the selection list. If you have auto-label enabled in Preferences|Forms, the label will fill in with the name value. You can edit this if you want.
3 Enter the size (height) of the list.
4 Edit the label text if you want. If you do not have the auto-label feature enabled, you will not be able to enter into this field.
5 Enter the Selection Text that you want displayed on the form.
6 Enter a return value if you want a value returned that is different from the selection text.
7 Click in the Selected column to indicate that you want the option selected by default when the form is loaded. A 'x' will appear when the option is to be selected.
8 Selection options can be reordered by clicking on a row of the left column, holding down the mouse button, dragging the row to its new location and releasing the mouse button.
9 Selection options can be deleted by clicking on a Selection Text cell and clicking the delete button. You can change the Text or Value by clicking in a cell and changing the content.
10 When you are finished, click the Ok button to insert the selection list into the form (if you are using the Form Assistant) or your document. Click the Cancel button to leave the Selection List Assistant without inserting the list.

Selection of Option(s)

compatibility

<select></select>

Usage:

The select tag provides a list of values for the user to select from. The values are given in <OPTION> elements.

HTML 3.0 extends the SELECT element to support graphical menus. This allows you to specify an image for the SELECT element, and hotzones for each of the OPTION elements. In this way the same menu can be rendered as a conventional text-based menu for text based browsers and a graphical menu for graphical browsers.

J This tag has javascript events associated with it. For details see the description of the Javascript event handlers.

(See Also: Form Assistant and Form Field Assistants)

Attributes:

Required

name - specifies the name of the form field.

Optional

size - specifies the number of visible items. Select fields of size one are typically drop-down lists, whereas select fields with size greater than one are typically lists.

multiple - [MULTIPLE] - indicates that more than one option may be selected by the user and included in the value.

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 src - Used to specify an image to appear preceding the header.

3 md - Specifies a message digest used when you want to be sure that a linked graphic, specified by the SRC attribute, is the same one intended, and hasn't been changed in any way.

3 **disabled** [DISABLED]- Used to create a "read-only" menu that can't be modified by the user. The menu may 'look' disabled e.g. by graying out the text, etc.

3 **error** - This attribute specifies an error message explaining why the menus current selections are incorrect. When this attribute is missing, the menu can be assumed to be ok.

3 **align** - [TOP | MIDDLE | BOTTOM | LEFT | RIGHT] - vertical alignment of the image. For use only with 'TYPE=IMAGE'. With ALIGN=LEFT, the graphic will float down and over to the

current left margin, and subsequent text will wrap around the right hand side of the graphic. Likewise for ALIGN=RIGHT, the graphic aligns with the current right margin and, and text wraps around the left.

- **width** - Used to specify a suggested width for the image. By default, this is given in pixels.

- **height** - Used to specify a suggested height for the image. By default, this is given in pixels.

- units - Used to specify the units for the width and height attributes.

Example:

```
<HTML>
<HEAD>
<TITLE>Example Form</TITLE>
</HEAD>
<BODY><FORM ACTION="http://www.somewhere.com/cgi-bin/mailscript"
METHOD="POST">
<P>Operating System <SELECT NAME="Operating
System"><OPTION>Windows95</OPTION>
</SELECT>
</P></FORM></BODY></HTML>
```

Typical Rendering (approximate):

Not applicable on its own. See Option.

Small

compatibility

<small></small>

Usage:

This tag specifies that the enclosed text should be displayed, if practical, using a smaller font than the current font.

Netscape has started supporting this tag with V2.0 of their browser.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
</HEAD>
<BODY>
<H1>Other information about HTML.</H1>
<P>There is lots of information to be had about <SMALL>HTML</SMALL> on
the Internet.</P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Spell Checking

(registered version only)

The spell checking feature allows you to check the spelling of the content of your document. The spell checker knows not to check tags, attributes and entities - that is the validators job! You can choose to use any combination of language dictionaries along with a custom dictionary.

To check the spelling of your document:

1 Select Options|Check Spelling from the menu.

2 The spell checker will begin checking your document. If it does not find any errors it will close again after telling you that it is finished.

3 When it finds a mistake it will highlight and display the word it thinks is incorrect and offer suggestions if it can.

4 To replace the word with another from the list, select the desired word and click the replace button. To replace the word with another not on the list, type the word into the 'Replace with' box and click the replace button. All subsequent words misspelled in the same way will be replaced with the same correction automatically.

5 To ignore a word once, click the 'Ignore' button. To ignore a word always, click the 'Ignore all' button.

6 To add the word to your custom dictionary, click the 'Add' button.

7 To set options for the spell checker such as ignoring words with numbers, click the 'Options' button.

8 The american english dictionary is used by default. Dictionaries are available in several other languages as well. To specify which dictionary to use click the 'Dictionaries' button.

Spell Checking Options

(registered version only)

To set spell checking options:

1 Click the 'Options' button on the spell checker.

2 Check 'Ignore words containing numbers' to tell the spell checker to ignore words that contain a mix of characters and numbers such as Canadian and British postal codes, phone numbers etc. Any words made up of numbers alone will always be ignored.

3 Check 'Ignore full capitalization' to ignore any words that are completely in upper case.

4 Check 'Hyphen is word separator' to tell the spell checker to check any words separated by a hyphen as whole words on their own. For example: world-wide would be checked as two separate words, world and wide. If the parts of the hyphenated word are spelled correctly than the word is considered to be correct. If this option is not enabled then the word world-wide would have to be in the dictionary as a whole to be considered correct.

5 Click the 'Ok' button to accept the changes and return to the spell checker.

Spot

compatibility

<spot>

Usage:

This tag is used to insert IDs at arbitrary places. i.e. for end points of a marked <RANGE>. i.e.

```
<SPOT ID=spot01> ... <SPOT ID=spot02>
```

Attributes:

Required

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

Optional

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
<RANGE CLASS=Search FROM="spot01" UNTIL="spot02">
</HEAD>
<BODY>
<H1>Other information about HTML.</H1>
<SPOT ID="spot01"><P>There is lots of information to be had about
<ACRONYM>HTML</ACRONYM> on the Internet.</P><SPOT ID="spot02">
</BODY>
</HTML>
```

Typical Rendering (approximate):

Not applicable.

Status Line

The status line is the area at the bottom of the application window. This area is used to display messages, hints about program elements, the state of Tag Protect, insert mode and the column (character) and row (line) number.

- Tag Protect and Insert Mode can be toggled on and off by clicking on their buttons on the status line.
- The current date and time can be inserted at the cursor position by clicking on their buttons. Date and time formats can be set in the Preferences.
- You can also see what dtd you are working with and change it by right-clicking on the button and selecting the desired specification from the pop-up list.
- New tabbed edit windows can be opened by right-clicking on other areas of the status line.

Strategies for Fixing Invalid Documents

Here are some basic strategies for fixing problems reported by Webber™ validation:

- Fix the first problem first. It may seem obvious but sometimes problems are related, if you fix the first problem and don't understand the others try doing a 'Restart'. The other problems may also be fixed or you may get a different message.

- Fix the easy ones first. No, it's not cheating, go after any problems you understand and leave the hard ones till last, if you're lucky they'll go away.

- Check your help file. The Webber™ help file has a wealth of information on correct HTML usage and all of the Webber validation problem description messages.

- Check the Tag Assistant with Tags in Context enabled. It is possible that you have a tag or content where it does not belong.

- Change things. If you don't know what's wrong try changing something and see what affect it has on the messages. Add a tag, remove a tag, anything, just play around with it. After a while you'll get a sense of what's causing each message.

- Try commenting out an offending section. A great feature of SGML, and by extension HTML, is the comment. The validator, (and browsers), will ignore something that has been surrounded with a comment. Comments start with a '<!--' with a '-->' to end. This is a good way to isolate the problem.

Strike

compatibility

<strike></strike>

Usage:

This tag specifies that the enclosed text should be displayed with a horizontal line striking through the text.

This tag is supported by Netscape and Internet Explorer, the HTML 3 equivalent is <S>, which is supported by Mosaic and Internet Explorer.

Attributes:

Required

Optional

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
</HEAD>
<BODY>
<H1>Other information about HTML.</H1>
<P>There is lots of information to be had about HTML on the
<STRIKE>Internet</STRIKE> World Wide Web.</P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Other Information about HTML

There is lots of information to be had about HTML on the ~~Internet~~ World Wide Web.

Strike Through

compatibility

<s></s>

Usage:

This tag specifies that the enclosed text should be displayed with a horizontal line striking through the text. If this is not practical, an alternative mapping is allowed.

This tag is currently supported by Mosaic and Internet Explorer but Netscape has its own version - <STRIKE>

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
</HEAD>
<BODY>
<H1>Other information about HTML.</H1>
<P>There is lots of information to be had about HTML on the
<S>Internet</S> World Wide Web.</P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Other Information about HTML

There is lots of information to be had about HTML on the ~~Internet~~ World Wide Web.

Stripping the HTML Tags

The HTML tags can be easily removed from a document using the Strip Tags feature. Strip Tags is available from the menu by selecting HTML|Document|Strip Tags.

Strip Tags creates and removes the HTML mark-up from a new untitled copy of your document - your original is not touched. Hard returns are added after headings, paragraphs, etc. in an attempt maintain the layout of the document.

Strong

compatibility

****

Usage:

This tag is used for placing emphasis on a word or phrase. Browsers often render it in bold.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Glossary</TITLE>
</HEAD>
<BODY>
<H1>Glossary</H1>
<P>This is the <STRONG>glossary</STRONG> where you can find
definitions for common phrases.</P>
</BODY><HTML>
```

Typical Rendering (approximate):

Glossary

This is the **glossary** where you can find definitions for common phrases.

Style

compatibility

<style></style>

Usage:

This tag provides a means for including rendering information using a specified style notation. Information in the STYLE element overrides client defaults and that of linked style sheets. It allows authors to specify overrides, while for the most part using a generic style sheet, and as such improves the effectiveness of caching schemes for linked style sheets. There is one attribute - NOTATION - which specifies an entity identifying an SGML notation in the HTML 3.0 DTD.

Attributes:

Required

notation - Specifies an entity identifying an SGML notation in the HTML 3.0 DTD.

Optional

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
<STYLE NOTATION="dsssl-lite">
--specifics for overrides and rendering...--
</STYLE></HEAD>
<BODY>
<H1>Other information about HTML.</H1>
<P>There is lots of information to be had about HTML on the
Internet.</P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Submit Button Assistant

Submit Buttons are input fields that appear on a form as a button. Their purpose is to submit the contents of the form. The Submit Button Assistant is meant to help you place a submit button in a form with the needed attributes set. The Assistant is available from the HTML|Control|Text Field... menu and from the Form Assistant.

The attributes usually used with submit are **name** and **value**. The name is returned when the form is submitted as the first in the name/value pair. This attribute is optional. The button will still submit the form and does not need to be retuned itself. It is useful to get a return from a submit button when there is more than one button on a form and you (or a script) need to know which one was pushed by the user. The value is used as the label on the button and should therefore be filled regardless of whether you want it returned or not.

To Use:

1 Open the Submit Button Assistant from the Form Assistant Dialogue by choosing Submit Button from the drop-down list or from the menu by selecting HTML|Controls|Submit Button. If you are using the latter method, be sure to first position your cursor where you want the button inserted.

2 Enter the name of the button if you want it returned.

3 Enter the value of the button. This is used as the button label for display on the form.

4 When you are finished, click the Ok button to insert the button into the form (if you are using the Form Assistant) or your document. Click the Cancel button to leave the Submit Button Assistant without inserting the button.

Subscript

compatibility

****

Usage:

This tag specifies that the enclosed text should be displayed as a subscript, and if practical, using a smaller font compared with the current text. The ALIGN attribute for SUB is only meaningful within the <MATH> element.

Netscape has started supporting this tag with V2.0 of their browser.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 align [LEFT | CENTER | RIGHT] -

Example:

```
<HTML><HEAD>
<TITLE>Our Planet</TITLE>
</HEAD>
<BODY>
<H1>Information about Our Planet</H1>
<P>Our planet is covered mostly by water. The chemical symbol for
water is H<SUB>2</SUB>O</P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Superscript

compatibility

****

Usage:

This tag specifies that the enclosed text should be displayed as a superscript, and if practical, using a smaller font compared with the current text. The ALIGN attribute for SUP is only meaningful within the <MATH> element.

Netscape has started supporting this tag with V2.0 of their browser.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 align [LEFT | CENTER | RIGHT] -

Example:

```
<HTML><HEAD>
<TITLE>Webber (TM)</TITLE>
</HEAD>
<BODY>
<H1>Information about Webber<SUP>TM</SUP></H1>
<P>Webber<SUP>TM</SUP> is a great HTML Editor!</P>
</BODY></HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Syntax Highlighting

This feature is extremely helpful for picking out your content in complex tag structures such as tables, and for immediately identifying missing angle brackets. Any text inside the left and right angle brackets will be highlighted in a user-definable colour. Another helpful tool is the Bracket Matcher.

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 3.0//EN">
<HTML><HEAD><TITLE>September Calendar</TITLE>
</HEAD><BODY><TABLE BORDER>
<TR><TD></TD><TH COLSPAN="5" VALIGN="Top" ALIGN="Center"> September
1995</TH><TD></TD></TR>
<TR><TH VALIGN="Top" ALIGN="Center">Sunday</TH><TH VALIGN="Top"
ALIGN="Center">Monday</TH><TH VALIGN="Top"
ALIGN="Center">Tuesday</TH><TH VALIGN="Top"
ALIGN="Center">Wednesday</TH><TH VALIGN="Top"
ALIGN="Center">Thursday</TH><TH VALIGN="Top"
ALIGN="Center">Friday</TH><TH VALIGN="Top"
ALIGN="Center">Saturday</TH></TR>
</TABLE>
</BODY></HTML>
```

This feature is enabled by default. To disable the feature or to change the colour, open Options| Preferences from the menu and click on the 'Editor' tab. Change the colour selection by clicking on the 'syntax colour' button and choosing a new colour from the resulting selection box. To disable the feature, uncheck the 'Syntax Highlighting' check box.

T

compatibility

<t></t>

Usage:

This tag is used, within a <MATH> tag, to change the default font used to render variables and constants. It suggests that its contents should be rendered in an upright font rather than an italic font. The <BT> tag combines this and bold rendering into one tag.

Attributes:

Required

Optional

class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<T>Variable 1</T>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Tab

compatibility

<tab>

Usage:

This tag is used when you want fine control over the horizontal positioning. The TAB tag is used with the ID attribute to define named tab stops. Subsequently, you can use the TAB tag with the TO attribute to move to the previously defined tab stop. This approach avoids the need to know the font metrics in advance.

The TAB element, together with style sheets, allows conversion software to preserve layout information when importing documents created with conventional word processing software.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document. Used to name a new tab stop at the current position. The scope of the tab stop is the rest of the document.

to - IDRef - References an SGML identifier - NAME token - for an element in the document body. Specifies a previously defined tab stop (see ID attribute).

indent - Specifies the number of en units before the tab stop. It allows authors to control the leading indent before text . The INDENT attribute is not meaningful when combined with the TO attribute.

align [LEFT | CENTER | RIGHT | DECIMAL] - Lines are usually rendered according to the alignment option for the enclosing paragraph element. The ALIGN attribute can be used to explicitly specify the horizontal alignment.

dp - Specifies the character to be used for the decimal point with the ALIGN attribute, e.g. dp="." (the default) or dp=",". The default may be altered by the language context, as set by the LANG attribute on enclosing elements.

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
</HEAD>
<BODY>
<H1>Other information about HTML.</H1>
<P><B>Hyper<TAB ID="t1">text Mark-up</B><BR>
<TAB TO="t1">To mark-up a document with a language designed for
creating hypertext links.</P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Tabbed Window

Each document you open in Webber will be displayed in a tabbed window, allowing you to easily manage large groups of open documents.. You can switch between open documents by clicking on their tab at the bottom of the window. To open a new tabbed window, select Window|New Edit Window from the menu. You can open a blank document into a new tabbed window, an HTML outline, or an existing file. You can also right-click on the status line for a pop-up window menu.

To close a tabbed window use the close button (or control menu) on the title bar - by default this will close all files open in the window. To close one file in the window select File|Close from the menu or right-click on the document and select Close from the pop-up list. You can also set the 'single close' preference to specify that you want the close button to close only one file at a time.

To close all open edit windows, select Window|Close All from the menu. To tile two documents, open each file in its own edit window, then select 'tile' from the window menu.

Table

compatibility

<table></table>

Usage:

Tables allow the presentation of tabular data in HTML. Previously data had to be formatted manually with the preformat tag using a fixed width font. Tables can contain a wide range of content, such as headers, lists, paragraphs, forms, figures, preformatted text and nested tables, and thus can also be used for layout of text and graphics. Under the recommended dtd's, table cells can not contain text, images etc. - they must be within a container tag like a paragraph.

Tables start with an optional caption, followed by an optional COL and/or COLGROUP. The COL and COLGROUP can be repeated. They are followed by an optional THEAD, then an optional TFOOT, followed by a required TBODY. There can be more than one table body. Table headers, footers and bodies can contain one or more rows. Each row is formed by one or more header cells and data cells.

(See Also: Table Assistant and Table Editor)

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

style - specifies rendering information about the element

dir [LTR | RTL] - Refers to the direction characters are written. (Left to right, right to left) When applied to TABLE, it indicates the geometric layout of rows (i.e. row 1 is on right if DIR=RTL, but on the left if DIR=LTR) and it indicates a default base direction for any text in the table's content if no other DIR attribute applies to that text.

cellspacing - Used to specify the amount of space inserted between individual cells in a table.

cellpadding - Used to specify the amount of space between the border of the cell and the contents of the cell.

width - Specifies the width of the table in standard units.

border - Specifies that the width of the border around the table.

cols - Specifies the number of columns for the table. If present the browser may render the table as data is received from the network without waiting for the complete table. If the COLS attribute is absent, a prepass through the table's contents is needed to determine the number of columns together with suitable values for the widths of each column.

align [LEFT | CENTER | RIGHT] - Specifies horizontal alignment of the table (*not its contents*).

frame [VOID | ABOVE | BELOW | HSIDES | LHS | RHS | BOX | VSIDES | BORDER] - Specifies which sides of the frame to render.

rules [NONE | BASIC | ROWS | COLS | ALL] - Specifies where to draw rules within the table interior.

There are also some attributes that are Internet Explorer Extensions.

Example:

```
<HTML>
<HEAD><TITLE>Example Table</TITLE>
</HEAD><BODY>
<TABLE BORDER="1" ALIGN="Center" COLS="3">
<CAPTION ALIGN="TOP">A test table showing merged cells</CAPTION>
<TR><TH COLSPAN="3" VALIGN="Top" ALIGN="Center"> Average Temperature
in Celsius</TH></TR>
<TR><TD></TD>
<TH VALIGN="Top" ALIGN="Center">Toronto</TH><TH VALIGN="Top"
ALIGN="Center">Vancouver</TH></TR>
<TR><TH VALIGN="Top" ALIGN="Right">January</TH>
<TD VALIGN="Middle" ALIGN="Center">-5</TD>
<TD VALIGN="Middle" ALIGN="Center">2</TD></TR>
<TR><TH VALIGN="Top" ALIGN="Right">June</TH>
<TD VALIGN="Middle" ALIGN="Center">32</TD>
<TD VALIGN="Middle" ALIGN="Center">25</TD></TR>
</TABLE>
</BODY></HTML>
```

Typical Rendering (approximate):

A test table showing merged cells

	Average Temperature (in Celcius)	
	Toronto	Vancouver
January	-5	+2
June	32	25

Table Assistant

The Table Assistant allows you to quickly set the number of columns and rows that you want in your table. In addition you can set optional attributes and enter a caption.

Size Tab
Table (Optional) Tab
Caption (Optional) Tab

When you are finished modifying the table settings you can click the 'Edit' button to open the Table Editor. The Table Editor takes the information from the assistant and formats a grid which you can use to set cell alignment, column and row spanning, and enter your data.

3 If you do not want to use the Table Editor you can click the Copy button. This will convert the contents of this dialogue to HTML and store it on the clipboard. You can then paste the resulting table into any document.

Table Attributes Tab

This page of the Table Assistant allows you to set the optional attributes applicable to a table. These attributes may be not all be supported by any one browser. It is best to check your browser documentation or test the features yourself.

Fill out any attributes you wish to use. For a description of any of these attributes see the Table topic.

When finished with the attributes, you can also choose to adjust the size or add a caption. Then you can either edit the table using the Table Editor, or copy it to the clipboard so that you can paste it into your document.

Table Body

compatibility

<tbody></tbody>

Usage:

Tables may be divided up into head and body sections. The THEAD and TFOOT elements are optional, but one or more TBODY elements are always required. If the table only consists of a TBODY section, the TBODY start and end tags may be omitted, as the parser can infer them.

The THEAD, TFOOT and TBODY elements provide a convenient means for controlling rendering. If the table has a large number of rows in the body, browsers may choose to use a scrolling region for the table body sections. When rendering to a paged device, tables will often have to be broken across page boundaries. The THEAD, TFOOT and TBODY elements allow the user agent to repeat the table foot at the bottom of the current page, and then the table head at the top of the new page before continuing on with the table body.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

style - specifies rendering information about the element

dir [LTR | RTL] - Refers to the direction characters are written. (Left to right, right to left) When applied to TABLE, it indicates the geometric layout of rows (i.e. row 1 is on right if DIR=RTL, but on the left if DIR=LTR) and it indicates a default base direction for any text in the table's content if no other DIR attribute applies to that text.

valign [TOP | MIDDLE | BOTTOM | BASELINE] - Used to specify the vertical alignment of material within a table cell.

align [LEFT | CENTER | RIGHT | JUSTIFY | CHAR] - Used to explicitly specify the horizontal alignment.

char - Used to specify an alignment character for use with align=char, e.g. char=".". The default character is the decimal point for the current language, as set by the LANG attribute. The CHAR attribute value is case sensitive.

charoff - Specifies the offset to the first occurrence of the alignment character on each line. In addition to standard units, the "%" sign may be used to indicate that the value specifies the alignment position as a percentage offset of the current cell, e.g. CHAROFF="30%" indicates the alignment character should be positioned 30% through the cell.

Example:

```
<HTML>
<HEAD><TITLE>Example Table</TITLE>
</HEAD><BODY>
<TABLE BORDER="1" ALIGN="Center" FRAME="HSides" RULES="Rows"
CELLSPACING="10" CELLPADDING="10" COLS="3">
<THEAD><TR><TD>January</TD><TD>February</TD><TD>March</TD></TR>
```

```
</THEAD>
<TFOOT><TR><TD COLSPAN="3">Average Temperatures</TD></TR>
</TFOOT>
<TBODY>
<TR><TD>-20</TD><TD>-10</TD><TD>5</TD></TR>
<TR><TD>-21</TD><TD>-15</TD><TD>8</TD></TR>
<TR><TD>-15</TD><TD>-21</TD><TD>10</TD></TR>
</TBODY></TABLE>
</BODY></HTML>
```

Table Caption

This page of the Table Assistant allows you to include an optional caption for your table. A caption will appear above a table by default. You can change this alignment by using the radio buttons on this form.

For a description of a caption and its attributes, see the Caption topic.

When finished with the caption, you can choose to adjust the size or add or edit the table attributes. Then you can either edit the table using the Table Editor, or copy it to the clipboard so that you can paste it into your document.

Table Data Cells

compatibility

<td></td>

Usage:

The TD and <TH> tags are used for table cells. TH is used for table header cells while TD is used for table data cells. This distinction gives browsers a means to render each type of cell distinctly, for instance by using a larger or heavier font for header cells. It is also needed when rendering to speech.

The CLASS attribute can be used to further differentiate cells, for instance into heads and subheads. This can be used together with style sheets to control the cell border style, and fill color etc.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

style - specifies rendering information about the element

dir [LTR | RTL] - Refers to the direction characters are written. (Left to right, right to left) When applied to TABLE, it indicates the geometric layout of rows (i.e. row 1 is on right if DIR=RTL, but on the left if DIR=LTR) and it indicates a default base direction for any text in the table's content if no other DIR attribute applies to that text.

valign [TOP | MIDDLE | BOTTOM | BASELINE] - Used to specify the vertical alignment of material within a table cell.

align [LEFT | CENTER | RIGHT | JUSTIFY | CHAR] - Used to explicitly specify the horizontal alignment.

char - Used to specify an alignment character for use with align=char, e.g. char=".". The default character is the decimal point for the current language, as set by the LANG attribute. The CHAR attribute value is case sensitive.

charoff - Specifies the offset to the first occurrence of the alignment character on each line. In addition to standard units, the "%" sign may be used to indicate that the value specifies the alignment position as a percentage offset of the current cell, e.g. CHAROFF="30%" indicates the alignment character should be positioned 30% through the cell.

colspan - The number of columns spanned by this cell. This allows you to merge cells across columns. It defaults to 1 (one). COLSPAN=0 has a special significance and implies that the cell spans all columns from the current column up to the last column of the table.

rowspan - The number of rows spanned by this cell. This allows you to merge cells across rows. It defaults to 1 (one). ROWSPAN=0 has a special significance and implies that the cell spans all rows from the current row up to the last row of the table. nowrap [NOWRAP] - Used to prevent the browser from automatically wrapping lines.

axis - Defines an abbreviated name for a header cell, which can be used when rendering to speech. It defaults to the cell's content.

axes - A comma separated list of axis names which together identify the row and column headers that pertain to this cell. It is used when rendering to speech to identify the cell's position

in the table. If missing the user agent can try to follow up columns and left along rows (right for some languages) to find the corresponding header cells.

N **width** - Used to describe the desired width of the cell, either as an absolute width in pixels, or a percentage of table width. (Since this attribute is not part of the HTML 3 spec it is recommended that you use the width attribute for the COL element instead.)

There are also some attributes that are Internet Explorer Extensions.

Example:

```
<HTML>
<HEAD><TITLE>Example Table</TITLE>
</HEAD><BODY>
<TABLE BORDER="1" ALIGN="Center" COLS="3">
<CAPTION ALIGN="TOP">A test table showing merged cells</CAPTION>
<TR><TH COLSPAN="3" VALIGN="Top" ALIGN="Center"> Average Temperature
in Celsius</TH></TR>
<TR><TD></TD>
<TH VALIGN="Top" ALIGN="Center">Toronto</TH><TH VALIGN="Top"
ALIGN="Center">Vancouver</TH></TR>
<TR><TH VALIGN="Top" ALIGN="Right">January</TH>
<TD VALIGN="Middle" ALIGN="Center">-5</TD>
<TD VALIGN="Middle" ALIGN="Center">2</TD></TR>
<TR><TH VALIGN="Top" ALIGN="Right">June</TH>
<TD VALIGN="Middle" ALIGN="Center">32</TD>
<TD VALIGN="Middle" ALIGN="Center">25</TD></TR>
</TABLE>
</BODY></HTML>
```

Typical Rendering (approximate):

A test table showing merged cells

	Average Temperature (in Celcius)	
	Toronto	Vancouver
January	-5	+2
June	32	25

Table Editor

The Table Editor is a powerful addition to the Table Assistant. Using the Table Editor you can set **column and row spanning**, **horizontal and vertical cell alignment**, and **enter your content and HTML tags**. Other Tag Assistants such as the Image Assistant and the Anchor Assistant work with the Table Editor so you can use them to add anchors, forms and images to your tables.

The Table Editor can also pick up existing tables from your document, and load them into itself. To do this, place your cursor in the table start tag and click the table editor button. (*please note that if you have modified the table to use attributes not supported by the table editor, i.e. background colors, you will have to add them back manually since the table editor will not understand them.*)

The **copy button** converts the contents of the editor into HTML and stores it on the clipboard so that you can paste it into your document. When you are finished setting up your table, click the copy button, then place your cursor in your document on the position you want the table to be inserted. Click the paste button on the tool bar or select Edit|Paste from the menu.


The Table Assistant can be re-opened at any time to modify the size or attributes of the table. (Be careful when resizing - content contained in rows and columns that have been removed will be lost.)

Right Click Menu Behaviour:

Column and Row Spanning

To **join** cells of a column or a row, select the cells you want to join, right click and select 'Join Cells' from the pop-up menu. Any content already contained in the cells to be joined will be separated from other content with a space.

To **split** previously joined cells, select the cell you want to split, right click and select 'Split Cell' from the pop-up menu. Any content from the cell will be transferred in to the top-most or the left-most cell.

Inserting Rows and Columns

New rows and columns will be inserted before the currently selected row or column. To insert a new row (or column), click on any cell in a row (or column), right click and select 'Row|Insert' (or Column|Insert) from the pop-up menu.

To add a new row or column at the end of your table, right click and select Row|Append (or Column|Append) from the pop-up menu. Alternatively, you can use the Table Assistant|Size dialogue to resize your table.

Deleting Rows and Columns

To delete row or column, select any cell in the row (or column), right click and select Row|Delete (or Column|Delete) from the pop-up menu. Alternatively, you can use the Table Assistant|Size dialogue to resize your table. A message will ask you to confirm that you want to delete your selection.

Horizontal Cell Alignment

To set the **horizontal alignment** for a cell or group of cells, select the cells, right click and select 'H Align' and your choice from the pop-up menu. Alternatively you can use the alignment buttons on the tool bar of the table editor.

Vertical Cell Alignment

To set the **vertical alignment** for a cell or group of cells, select the cells, right click and select 'V Align' and your choice from the pop-up menu. Alternatively you can use the alignment buttons on the tool bar of the table editor. Due to constraints of our table editor, the content of cells will not *look* vertically aligned within the editor, but the attributes will be set in the HTML.

Setting Table Headers

To set a cell or group of cells as table header cells, select the cells, right click and select 'Header' from the pop-up menu. The default attributes of header cells are different than those of table data cells.

Alignment Buttons:

To set the vertical or horizontal alignment of a cell or groups of cells, select the cells and click one of the alignment buttons on the tool bar. You can also use the right-click menu for these behaviours. It is not possible to display the contents of the cells in their vertically aligned state but the buttons on the tool bar will show you what they are set to.

Internet Explorer Extensions (applicable to the table tag, data and header cells)

ALIGN=LEFT, or RIGHT - Specifies that the table or the text can be left- or right-aligned. The default is left-aligned for TABLE, TR, and TD. The default is center-aligned for TH.

BGCOLOR= #rrggbb or colorname - Sets background color. rrggbb is a hexadecimal number denoting a red-green-blue color value (the pound sign is optional). BGCOLOR and each of the other color attributes here can also be set to a colorname. See Internet Explorer web site for info on colornames.

BORDERCOLOR= #rrggbb or colorname - Sets border color and must be used with the BORDER attribute. BORDERCOLOR can also be set to a colorname.

BORDERCOLORLIGHT= #rrggbb or colorname - Sets independent border color control over one of the two colors used to draw a 3-D border, opposite of BORDERCOLORDARK, and must be used with the BORDER attribute.

BORDERCOLORDARK= #rrggbb or colorname - Sets independent border color control over one of the two colors used to draw a 3-D border, opposite of BORDERCOLORLIGHT, and must be used with the BORDER attribute. Can also be set to a colorname.

VALIGN=TOP, or BOTTOM - Specifies that the text can be top- or bottom-aligned. The default is center-aligned.

BGCOLOR is also supported by Netscape starting with Atlas.

Table Foot

compatibility

<tfoot></tfoot>

Usage:

Tables may be divided up into head and body sections. The THEAD and TFOOT elements are optional, but one or more TBODY elements are always required. If the table only consists of a TBODY section, the TBODY start and end tags may be omitted, as the parser can infer them.

The THEAD, TFOOT and TBODY elements provide a convenient means for controlling rendering. If the table has a large number of rows in the body, browsers may choose to use a scrolling region for the table body sections. When rendering to a paged device, tables will often have to be broken across page boundaries. The THEAD, TFOOT and TBODY elements allow the user agent to repeat the table foot at the bottom of the current page, and then the table head at the top of the new page before continuing on with the table body.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

style - specifies rendering information about the element

dir [LTR | RTL] - Refers to the direction characters are written. (Left to right, right to left) When applied to TABLE, it indicates the geometric layout of rows (i.e. row 1 is on right if DIR=RTL, but on the left if DIR=LTR) and it indicates a default base direction for any text in the table's content if no other DIR attribute applies to that text.

valign [TOP | MIDDLE | BOTTOM | BASELINE] - Used to specify the vertical alignment of material within a table cell.

align [LEFT | CENTER | RIGHT | JUSTIFY | CHAR] - Used to explicitly specify the horizontal alignment.

char - Used to specify an alignment character for use with align=char, e.g. char=".". The default character is the decimal point for the current language, as set by the LANG attribute. The CHAR attribute value is case sensitive.

charoff - Specifies the offset to the first occurrence of the alignment character on each line. In addition to standard units, the "%" sign may be used to indicate that the value specifies the alignment position as a percentage offset of the current cell, e.g. CHAROFF="30%" indicates the alignment character should be positioned 30% through the cell.

Example:

```
<HTML>
<HEAD><TITLE>Example Table</TITLE>
</HEAD><BODY>
<TABLE BORDER="1" ALIGN="Center" FRAME="HSides" RULES="Rows"
CELLSPACING="10" CELLPADDING="10" COLS="3">
<THEAD><TR><TD>January</TD><TD>February</TD><TD>March</TD></TR>
```

```
</THEAD>
<TFOOT><TR><TD COLSPAN="3">Average Temperatures</TD></TR>
</TFOOT>
<TBODY>
<TR><TD>-20</TD><TD>-10</TD><TD>5</TD></TR>
<TR><TD>-21</TD><TD>-15</TD><TD>8</TD></TR>
<TR><TD>-15</TD><TD>-21</TD><TD>10</TD></TR>
</TBODY></TABLE>
</BODY></HTML>
```

Table Head

compatibility

<thead></thead>

Usage:

Tables may be divided up into head and body sections. The THEAD and TFOOT elements are optional, but one or more TBODY elements are always required. If the table only consists of a TBODY section, the TBODY start and end tags may be omitted, as the parser can infer them.

The THEAD, TFOOT and TBODY elements provide a convenient means for controlling rendering. If the table has a large number of rows in the body, browsers may choose to use a scrolling region for the table body sections. When rendering to a paged device, tables will often have to be broken across page boundaries. The THEAD, TFOOT and TBODY elements allow the user agent to repeat the table foot at the bottom of the current page, and then the table head at the top of the new page before continuing on with the table body.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

style - specifies rendering information about the element

dir [LTR | RTL] - Refers to the direction characters are written. (Left to right, right to left) When applied to TABLE, it indicates the geometric layout of rows (i.e. row 1 is on right if DIR=RTL, but on the left if DIR=LTR) and it indicates a default base direction for any text in the table's content if no other DIR attribute applies to that text.

valign [TOP | MIDDLE | BOTTOM | BASELINE] - Used to specify the vertical alignment of material within a table cell.

align [LEFT | CENTER | RIGHT | JUSTIFY | CHAR] - Used to explicitly specify the horizontal alignment.

char - Used to specify an alignment character for use with align=char, e.g. char=".". The default character is the decimal point for the current language, as set by the LANG attribute. The CHAR attribute value is case sensitive.

charoff - Specifies the offset to the first occurrence of the alignment character on each line. In addition to standard units, the "%" sign may be used to indicate that the value specifies the alignment position as a percentage offset of the current cell, e.g. CHAROFF="30%" indicates the alignment character should be positioned 30% through the cell.

Example:

```
<HTML>
<HEAD><TITLE>Example Table</TITLE>
</HEAD><BODY>
<TABLE BORDER="1" ALIGN="Center" FRAME="HSides" RULES="Rows"
CELLSPACING="10" CELLPADDING="10" COLS="3">
<THEAD><TR><TD>January</TD><TD>February</TD><TD>March</TD></TR>
```

```
</THEAD>
<TBODY>
<TR><TD>-20</TD><TD>-10</TD><TD>5</TD></TR>
<TR><TD>-21</TD><TD>-15</TD><TD>8</TD></TR>
<TR><TD>-15</TD><TD>-21</TD><TD>10</TD></TR>
</TBODY></TABLE>
</BODY></HTML>
```

Table Header Cells

compatibility

<th></th>

Usage:

The TH and <TD> tags are used for table cells. TH is used for table header cells while TD is used for table data cells. This distinction gives browsers a means to render each type of cell distinctly, for instance by using a larger or heavier font for header cells. It is also needed when rendering to speech.

The CLASS attribute can be used to further differentiate cells, for instance into heads and subheads. This can be used together with style sheets to control the cell border style, and fill color etc.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

style - specifies rendering information about the element

dir [LTR | RTL] - Refers to the direction characters are written. (Left to right, right to left) When applied to TABLE, it indicates the geometric layout of rows (i.e. row 1 is on right if DIR=RTL, but on the left if DIR=LTR) and it indicates a default base direction for any text in the table's content if no other DIR attribute applies to that text.

valign [TOP | MIDDLE | BOTTOM | BASELINE] - Used to specify the vertical alignment of material within a table cell.

align [LEFT | CENTER | RIGHT | JUSTIFY | CHAR] - Used to explicitly specify the horizontal alignment.

char - Used to specify an alignment character for use with align=char, e.g. char=".". The default character is the decimal point for the current language, as set by the LANG attribute. The CHAR attribute value is case sensitive.

charoff - Specifies the offset to the first occurrence of the alignment character on each line. In addition to standard units, the "%" sign may be used to indicate that the value specifies the alignment position as a percentage offset of the current cell, e.g. CHAROFF="30%" indicates the alignment character should be positioned 30% through the cell.

colspan - The number of columns spanned by this cell. This allows you to merge cells across columns. It defaults to 1 (one). COLSPAN=0 has a special significance and implies that the cell spans all columns from the current column up to the last column of the table.

rowspan - The number of rows spanned by this cell. This allows you to merge cells across rows. It defaults to 1 (one). ROWSPAN=0 has a special significance and implies that the cell spans all rows from the current row up to the last row of the table. nowrap [NOWRAP] - Used to prevent the browser from automatically wrapping lines.

axis - Defines an abbreviated name for a header cell, which can be used when rendering to speech. It defaults to the cell's content.

axes - A comma separated list of axis names which together identify the row and column headers that pertain to this cell. It is used when rendering to speech to identify the cell's position

in the table. If missing the user agent can try to follow up columns and left along rows (right for some languages) to find the corresponding header cells.

N **width** - Used to describe the desired width of the cell, either as an absolute width in pixels, or a percentage of table width. (Since this attribute is not part of the HTML 3 spec it is recommended that you use the width attribute for the COL element instead.)

There are also some attributes that are Internet Explorer Extensions.

Example:

```
<HTML>
<HEAD><TITLE>Example Table</TITLE>
</HEAD><BODY>
<TABLE BORDER="1" ALIGN="Center" COLS="3">
<CAPTION ALIGN="TOP">A test table showing merged cells</CAPTION>
<TR><TH COLSPAN="3" VALIGN="Top" ALIGN="Center"> Average Temperature
in Celsius</TH></TR>
<TR><TD></TD>
<TH VALIGN="Top" ALIGN="Center">Toronto</TH><TH VALIGN="Top"
ALIGN="Center">Vancouver</TH></TR>
<TR><TH VALIGN="Top" ALIGN="Right">January</TH>
<TD VALIGN="Middle" ALIGN="Center">-5</TD>
<TD VALIGN="Middle" ALIGN="Center">2</TD></TR>
<TR><TH VALIGN="Top" ALIGN="Right">June</TH>
<TD VALIGN="Middle" ALIGN="Center">32</TD>
<TD VALIGN="Middle" ALIGN="Center">25</TD></TR>
</TABLE>
</BODY></HTML>
```

Typical Rendering (approximate):

3

Table Row

compatibility

<tr></tr>

Usage:

This tag acts as a container for a row of table cells defined with the <TH> or <TD> elements. You can set default horizontal and vertical alignment of cell contents for the row. You also have the ability to disable word wrap for the row, and thereafter use the <BR> tag to determine line breaks and hence cell widths.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

style - specifies rendering information about the element

dir [LTR | RTL] - Refers to the direction characters are written. (Left to right, right to left) When applied to TABLE, it indicates the geometric layout of rows (i.e. row 1 is on right if DIR=RTL, but on the left if DIR=LTR) and it indicates a default base direction for any text in the table's content if no other DIR attribute applies to that text.

valign [TOP | MIDDLE | BOTTOM | BASELINE] - Used to specify the vertical alignment of material within a table cell.

align [LEFT | CENTER | RIGHT | JUSTIFY | CHAR] - Used to explicitly specify the horizontal alignment.

char - Used to specify an alignment character for use with align=char, e.g. char=".". The default character is the decimal point for the current language, as set by the LANG attribute. The CHAR attribute value is case sensitive.

charoff - Specifies the offset to the first occurrence of the alignment character on each line. In addition to standard units, the "%" sign may be used to indicate that the value specifies the alignment position as a percentage offset of the current cell, e.g. CHAROFF="30%" indicates the alignment character should be positioned 30% through the cell.

Example:

```
<HTML><HEAD><TITLE>Table Test</TITLE></HEAD>
<BODY><TABLE BORDER>
<CAPTION>A test table showing merged cells</CAPTION>
<TR><TH ROWSPAN=2><TH COLSPAN=2>Average Temperature (in Celsius)
<TR><TH>Toronto<TH>Vancouver
<TR><TH ALIGN=LEFT>January<TD ALIGN=CENTER>-5<TD ALIGN=CENTER>+2
<TR><TH ALIGN=LEFT>June<TD ALIGN=CENTER>32<TD ALIGN=CENTER>25
</TABLE></BODY></HTML>
```

Typical Rendering (approximate):

Table Size Tab

This page of the Table Assistant allows you to select the number of columns and rows that you want to include in your table. This is the only requirement for setting up a table. When you have completed this step you can copy the table to the clipboard and paste it into your document.

You can set the size parameters in two ways:
1 Use the spin controls to adjust the number of columns and rows.
2 Use your mouse by holding down your mouse button while you drag the table to the desired size.

When finished with the size, you can also choose to set other attributes or add a caption. Then you can either edit the table using the Table Editor, or copy it to the clipboard so that you can paste it into your document.

Tag Assistant

The Tag Assistant is able to stay open all the time on your desktop. You can switch between it and your document freely. If you want to open the tag assistant quickly to a specific tag, you can either right-click on the tag name on the tagbar, or place your cursor on the tag name in your document and then click on the tag assistant. When you click in a start tag in your document, the Tag Assistant will 'pick-up' the attributes set for the tag, allowing you to modify, remove or add to them. You can also change a tag to another by placing your cursor in the start tag name and then clicking on the Tag Assistant. In the Tag Assistant, select a new tag from the list and click the insert button. The new tag will replace the original.

The Tag Assistant provides a list of tags allowed within the current dtd. As a tag is selected from the drop-down list, the attributes applicable to the tag are shown in a list box below. Attributes filled in here will be inserted with the tag when the Insert button is clicked. Context sensitive help for the current tag will appear when the Help button is clicked.

The Tags in Context button will restrict the tags on the list to those that are allowed by the HTML dtd within the context of the cursor position. You may use the insert button to add a tag without leaving the dialogue box. The cursor will be located after the start tag of the newly inserted tag, and Tags in Context will now show the tags allowed at this new position.

A superscript 3 following a tag name means that it is an HTML tag. Two stars (**) following a tag name means that it is an extension to HTML.

The tag assistant allows you to browse to images for the src attribute and will try to automatically get the height and width for you. This only works if the image is local and the directory path that you have entered is correct for your local system.

To Use:

1 Place your cursor where you want the new tag to appear. If you want to wrap the tag around existing content, select the desired text (including any tags you want inside the new tag) first.

2 Select HTML|Tag... from the menu, use the button on the toolbar, or right-click on a tag on the tagbar.

3 Choose the tag from the drop-down list if it is not already there.

4 Fill in required attributes and any optional attributes that you wish.

5 Some attributes offer some assistance in filling out their values. Attributes such as ALIGN for the image tag provides a scrolling list of the correct choices. The SRC attribute of the image tag will allow you to browse for a file, the as will the HREF attribute of the anchor tag and the BACKGROUND attribute of the body tag. The BGCOLOR and text colour attributes of the body tag will provide a colour selection dialog. To get this help click on the attribute name. A properties button [...] will appear at the right of the entry field. You can either type in your value or click the button for assistance in filling out the attribute. Also some attributes have a limited number of acceptable values. Such attributes, such as ALIGN, will pop-up a list of valid choices when you click in their entry boxes. You can select from the list by double-clicking your choice.

6 Click the Insert button to insert the new tag.

7 After you insert a tag, the Tag Assistant will roll-up automatically so that it does not take your valuable screen real estate. When you click back on the Tag Assistant it will unroll itself again. This feature can be turned on and off in the Preferences.

The Tag Assistant and Tagbar
- The Dynamic Duo!

The Tag Assistant and Tagbar have become a powerful team. Here is what they allow you to do:

- add your choice of tags to the tabbed tagbar - you can add new tabs and put any number of tags on a single tab
- left-click on the tagbar to insert a tag
- right-click on the tagbar to open the tag assistant to that tag
- fill in the attributes you want to use and click the insert button on the tag assistant. Use the property buttons to select sources for image tags and anchors, as well as background colours and text colours. Attributes with a limited number of options (as in align attributes) will show a pop-up list of the valid choices for selection with a double-click.
- the tag bar rolls up and down automatically to display and hide the attributes for a tag. (You can turn off the auto-roll in the preferences.)
- position your cursor inside the angle brackets of a start tag, then click on the tag assistant (or open it if its not open). The tag assistant will pick up the tag name and any attributes that are set for the tag. In this way you can easily modify existing attributes or add new ones. (Invalid attributes will be ignored and overwritten when you insert the changes.)
- position the cursor inside a start tag and click on the tag assistant. Select a new tag from the tag list and click the insert button - the tag assistant will change the original tag to the newly selected one - including the end tag. (This will not work if the end tag is not in the right place - i.e. <P>Paragraph content <H1>more content </P> - here the heading would end the paragraph so the end tag would not be found.)
- click the 'Tags in Context' button on the tag assistant to find out what tags are allowed at the cursor position
- although you can still 'hide' tags in the preferences, the tag assistant will work best if you make all tags visible

Alphabetical Tag List (HTML 2.0)

HTML Tags used in Version 2.0 of the HTML specification are listed below. Clicking on a tag name will take you to the topic for the tag. Each tag topic shows the common usage of the tag, the required and optional attributes for the tag, an example of a simple document using the tag, and the typical rendering of the document by graphical browsers.

For the complete list of tags in HTML 2, see the Quick Tag Reference.

You can also find out what tags are allowed within the context of your document and your location within the document by using the Tags in Context button in the Tag Assistant. *(See Also: Tags by Usage)* You can check this topic on Allowed Tags as well.

Tag List (HTML 3.0)

HTML Tags used in Version 3.0 of the HTML specification are listed below. Clicking on a tag name will take you to the topic for the tag. Each tag topic shows the common usage of the tag, the required and optional attributes for the tag, an example the tag use. Most of these tags are not yet well supported by browsers, we have not displayed a typical rendering of the tags.

For the complete list of tags HTML 3, see the Quick Tag Reference.

You can also find out what tags are allowed within the context of your document and your location within the document by using the Tags in Context button in the Tag Assistant.

Top level document Tag:

HTML (HTML Identifier)

Tags used to describe a document:

HEAD (Document Head)
- TITLE (Title)
- META (Meta-Information)
- BASE (Base)
- ISINDEX (Document is a Searchable Index)
- NEXTID (Next ID to Use)
- LINK (Link from Document)
- RANGE (Range)
- STYLE (Style)

Tags used for the content of a document:

BODY (Body)

- ALIAS (Alias) - also used in the head
- BANNER (Banner)
- INSERT (Insert Object)
- PARAM (Parameter)

Headings:

- H1 (Heading - Level 1)
- H2 (Heading - Level 2)
- H3 (Heading - Level 3)
- H4 (Heading - Level 4)
- H5 (Heading - Level 5)
- H6 (Heading - Level 6)

Text Block Elements:

- ADDRESS (Address)
- BLOCKQUOTE (Blockquote)
- BQ (Big quote)
- P (Paragraph)
- PRE (Preformatted Text)
- FN (Footnote)
- NOTE (Note)

Hypertext & Image Linking:

A (Anchor)
IMG (Image)
FIG (Figure)
    OVERLAY (Overlay)
SPOT (Spot)

List Elements:

OL (Ordered List)
UL (Unordered List)
    LH (List Header)
    LI (List Item)
DL (Definition List)
    DD (Definition Description)
    DT (Definition Term)

Font Style Elements:

B (Bold)
BIG (Big print)
I (Italics)
S (Strike through)
SMALL (Small print)
SUB (Subscript)
SUP (Superscript)
TT (Teletype)
U (Underline)

Other:

DIV (Division)
BR (Line Break)
HR (Horizontal Rule)
LANG (Language)
TAB (Tab)

Information Type Elements:

ABBREV (Abbreviation)
ACRONYM (Acronym)
AU (Author)
CITE (Citation)
CODE (Code)
DEL (Deleted)
DFN (Definition)
EM (Emphasis)
INS (Inserted)
KBD (Keyboard)
Q (Short Quote)
PERSON (Person)
SAMP (Sample)
STRONG (Strong)
VAR (Variable)

Tables:

TABLE (Table)
    CAPTION (Caption)

Forms:

Math Mark-up:

Obsolete Tags:

Tag Protection

Tag protection helps avoid accidental overwrites or deletions of HTML tags, while maintaining the flexibility of Webber as a tool that allows direct manipulation of tags. Tag Protect is enabled and disabled in Preferences|Editor, or by clicking on the button on the status line where the state of Tag Protect is displayed.

Tag Protect prevents:

- the deletion of an angle bracket by using the backspace key or the delete key from *outside* the tag.
- the overwriting of an angle bracket when insert mode is turned off.

Tag Protect allows:

- tags to be cut or deleted after being selected with the mouse.
- changes to be made to the tag inside the angle brackets. For example <H1> could be changed to <H3>.
- angle brackets to be deleted from *inside* the two angle brackets.

Allowed Tags & Content

There are times when you may be unsure about where content (text) is allowed or what tags are allowed in a certain context. This topic gives a brief summary for the most common tags in the HTML 2.0 dtd. It is not a comprehensive list and is meant only as a guide. You can find out what tags are allowed within the context of your document by using the Tags in Context feature. The validator will also help you find out of context tags and content.

- The HTML identifier is required and does not contain content. It can contain only the HEAD tag, the BODY tag, and the PLAINTEXT tag which has been deprecated.
- The HEAD tag does not contain content. It contains the TITLE of the document and may also contain META, BASE, ISINDEX, NEXTID, and LINK tags.
- The TITLE tag contains only content, and no other tags.
- The BODY tag does not contain content or text formatting tags. It contains 'container tags' such as headings, paragraphs, lists blockquote, preformat, address, and form, as well as horizontal rules.
- Text formatting tags, and anchors contain content but must themselves be contained within a container tag. They do not surround container tags.
- The H1 through H6 tags, paragraphs and lists contain content, IMG and A tags, as well as text formatting tags.
- The image tag is an empty tag and therefore cannot contain other tags or content. Under the recommended version of the HTML 3 dtd, the image tag must be contained within a block tag such as a paragraph or heading.
- The anchor tag contains content and must be placed within a container.
- The HR tag is an empty tag and therefore can't contain other tags or content. It is used to separate sections of a document and is not contained within other tags except the BODY tag.
- Headings and paragraphs do not contain each other, horizontal rules, or other container tags.
- The list tags - UL, OL, MENU, and DIR do not contain content. They can contain only LI tags and no other.
- The DL also does not contain content or many tags. It contains only the DT and DD tags.
- The definition term can contain content and text formatting tags but does not contain container tags.
- The list item and the definition description tags can contain content and text formatting tags, as well as most container tags except headings. They can contain other lists as well. In HTML 3 recommended you must use a paragraph, heading etc. to contain the content inside a list item. The same goes for table cells.
- The FORM tag cannot contain content but most other tags are legal.
- The form field tags INPUT, SELECT and TEXTAREA are only legal within a form tag.
- The input tag is an empty tag.
- The select tag can only contain the OPTION tag and not content.
- The text area can contain text and text formatting tags. All content of this tag is displayed as the value of the field when the form is loaded.
- The option tag contains only content.

Tag Surround

Webber will wrap a tag around selected text. Select the text you wish to place inside a tag and select the tag using the tagbar or a tag assistant. Tags can be wrapped around any content, including other tags.

Floating Tagbar

The tag bar is a customizable tool bar on which you can place any number of tags on any number of tabbed 'pages'. The tagbar can float freely on your screen, or be docked to the top or bottom of the Webber window. To change the position of the tagbar, click on an open area of the bar, hold down the mouse button and drag it to the desired position.

You can assign a tag to the tagbar and to a particular tab on the tagbar in Options|Edit Tagbar. Pausing your mouse pointer over the tag buttons will display the colloquial name for the tag. Right-clicking on a tag on the tag bar will open the Tag Assistant to that tag, showing you the attributes that are applicable to it. You can also assign custom bits to the tagbar.

To Use:

1 Customize the tag bar as desired or use the pre-defined default settings.

2 Display the tag bar by selecting Options|Show Tagbar from the menu. The floating tag bar will always stay on top. If it is currently displayed select the same menu option to hide it again.

3 Select the tab location for the desired tag by clicking on the tab names on the tag bar. If the tab you are looking for does not appear, use the scroll bar on the tag bar to scroll to the tab of your choice. When assigning tab names, you may want to use short names so that more tabs will appear on the tag bar at one time.

4 Highlight the text you want to tag, or place your cursor in the position you want the new tag to be inserted, then click the desired tag button. The tag will be inserted into your document with the cursor in position between the start and end tags.

5 Move the tag bar by clicking on its title bar, holding down the mouse button and dragging it to a new location.

Note: Remember that although you may have assigned a tag to the tool bar, it will appear grayed out if it is not part of the currently selected dtd.

Tagging a Document

Webber offers the flexibility of putting tags into an HTML document in many ways. Tags can be typed in by hand, selected from the drop-down list on the tag assistant, selected from the tagbar, or by using one of the Assistants. The complexity of the tag and your own comfort level with HTML will determine the best method. Simple tags such as line breaks and horizontal rules are easy to type in or add from the tagbar. For complex tags such as forms, we provide Assistants. Assistants provide more detailed information on a tag such as the required and allowed attributes and guide you through using them correctly.

For more information see the following:

- Tag Assistant and Tagbar - The Dynamic Duo
- Tagbar
- Tag Assistant
- Anchor Assistant
- Image Assistant
- List Assistant
- Table Assistant
- Form Assistant
- Form Control Assistants

Tag List (Grouped by Usage)

HTML Tags used in Version 2.0 of the HTML specification are listed below. Clicking on a tag name will take you to the topic for the tag. Each tag topic shows the common usage of the tag, the required and optional attributes for the tag, an example of a simple document using the tag, and the typical rendering of the document by browsers.

You can also find out what tags are allowed within the context of your document and your location within the document by using the Tags in Context button in the Tag Assistant. *(See Also: Tags Alphabetically)* You can check this topic on Allowed Tags as well.

Top level document Tag:

HTML (HTML Identifier)

Tags used to describe a document:

HEAD (Document Head)
- TITLE (Title)
- META (Meta-Information)
- BASE (Base)
- ISINDEX (Document is a Searchable Index)
- NEXTID (Next ID to Use)
- LINK (Link from Document)

Tags used for the content of a document:

BODY (Body)

Headings:
- H1 (Heading - Level 1)
- H2 (Heading - Level 2)
- H3 (Heading - Level 3)
- H4 (Heading - Level 4)
- H5 (Heading - Level 5)
- H6 (Heading - Level 6)

Paragraphs:
- ADDRESS (Address)
- BLOCKQUOTE (Blockquote)
- P (Paragraph)
- PRE (Preformatted Text)

Hypertext & Image Linking:
- A (Anchor)
- IMG (Image)

Regular Lists:
- DIR (Directory Listing)
- MENU (Menu List)

OL (Ordered List)
UL (Unordered List)

Regular List Items:
LI (List Item)

Other Lists:
DL (Definition List)

Other List Items:
DD (Definition Description)
DT (Definition Term)

Text Formatting:
B (Bold)
EM (Emphasis)
I (Italics)
STRONG (Strong)
TT (Teletype)

Other:
BR (Line Break)
CITE (Citation)
CODE (Code)
HR (Horizontal Rule)
KBD (Keyboard)
SAMP (Sample)
VAR (Variable)

Forms:
FORM (Form)

Form Entry:
INPUT (Form Input)
TEXTAREA (Text input area)
SELECT (Selection of option)
OPTION (Option)

Obsolete Tags:
LISTING (Listing)
XMP (Example)

Optional Tags used within <HTML>:

PLAINTEXT (Plain Text)

Target

Targeting is accomplished by means of the TARGET attribute. This attribute can be added to several of the HTML tags to target the links referred to by that tag. The syntax of the attribute is:
TARGET="window_name"

A

The anchor tag normally specifies a link to be loaded when the active item is clicked on, adding the TARGET attribute to the anchor tag forces the load of that link into the targeted window. Example:
<A HREF="url" TARGET="window_name">Targeted Anchor</A>

BASE

This is used when you want all (or most) of the links in a document to be targeted to the same window. In this case the TARGET attribute establishes a default window_name that all links in this document will be targeted to. This default is of course overridden by specific instances of the TARGET attribute in individual anchor tags. Example:
<BASE TARGET="window_name">

AREA

The IETF Internet-Draft of Client-Side Image Maps defines an area tag. This tag describes a shaped area in a client-side image map, and provides the link that should be followed when the user clicks there. Adding the TARGET attribute to the area tag forces the load of that link into the targeted window. Example:
<AREA SHAPE="shape" COORDS="x,y,..." HREF="url" TARGET="window_name">

FORM

The form tag normally displays the results of a form submission in the same window the form was submitted from. By adding the TARGET attribute to the form tag, the result of the form submission is instead loaded into the targeted window. Example:
<FORM ACTION="url" TARGET="window_name">

Common Tasks

Tag Help

Editor Window Behaviour

Status Line

Teletype

compatibility

<tt></tt>

Usage:

The teletype or typewriter tag indicates that the text should be rendered in typewriter font. This will usually be a monospaced font.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Webber Reference</TITLE>
</HEAD>
<BODY>
<H1>Webber Reference</H1>
<HR>
<P>To open the help file, select <TT>Help|Contents</TT> from the
menu.</P>
</BODY></HTML>
```

Typical Rendering (approximate):

Webber Reference

To open the help file, select `Help|Contents` from the menu.

Templates

The templates feature allows you to set up commonly used skeleton files. Create new templates by saving any HTML file to the /tmplts subdirectory in your Webber program directory. Any files in this directory will be listed in the Template dialog box when you select 'Open Template' from the file menu. The file will be opened as an untitled document to prevent it from being accidentally overwritten when you save your document.

Term in Definition List

compatibility

<dt></dt>

Usage:

The definition term is used only in definition lists. It follows the <DL> tag and contains the term to be defined. It is usually only a single line and is followed by the definition which can be multiple lines.

In HTML 3.0 an optional list header has been added. If used the list header must precede the first term in the list. Term names are restricted to character level markup only, including emphasis, in-line images and footnotes. Paragraph tags and other block-like tags such as headers are not allowed.

Attributes:

Required

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

Example:

```
<HTML><HEAD>
<TITLE>Index</TITLE>
</HEAD>
<BODY>
<H1>Glossary</H1>
<DL>
<DT>Definition List</DT>
<DD>Used to present a list of terms (a single line) and their
definitions (multiple lines).</DD>
<DT>Unordered List</DT>
<DD>Contains multiple lined list items which usually result in left
aligned, bulleted paragraphs. </DD>
</DL>
</BODY><HTML>
```

Typical Rendering (approximate):

Glossary

Definition List

Used to present a list of terms (a single line) and their definitions (multiple lines).

Unordered List

Contains multiple lined list items which usually result in left aligned, bulleted paragraphs.

Previewing Your Document

Shortcut - ctrl+b

If your browser supports it, you can preview your document by clicking the preview button on the tool bar, or by selecting HTML|Document|Preview. This will run your default browser (if it is not already running), and load your document. Your document must have a name before you can preview it. If it has a name but changes have been made to it and not saved, you will be prompted to do so - you can preview changes before saving them by choosing not to save. If your browser is already running, your document will be loaded (or reloaded) when you click the preview button.

When you want to preview your document with a different browser you can do so by right-clicking on the preview button, and selecting the desired browser from the '**use**' list on the pop-up menu. You can configure your browsers in the Preferences. Netscape, NCSA Mosaic, Internet Explorer and Spyglass Mosaic can all be used in this way.

If your browser does not support inter-application communication, Webber can start it and load a document but cannot change the document once the browser is running.

The preview feature also works from the Form Assistant but it is **not** necessary to save the form first - Webber will create and display a temporary file so that you can edit the form before inserting it into your document.

Text Area Assistant

Text areas are multiple line input fields that allow the user to type messages into a form. The Text Area Assistant is meant to help you place a text area in a form with the needed attributes set. The Assistant is available from the HTML|Control|Text Area... menu and from the Form Assistant.

The attributes usually used with text area are **name**, **rows** and **cols** (columns). The name is returned when the form is submitted as the first in the name/value pair, while the value is the text that the user types. i.e. A field with a name of message might return: "message=I think your page is great..." The rows attribute sets the displayed height of the field, while the cols attribute sets the width.

To Use:

1 Open the Text Area Assistant from the Form Assistant Dialogue by choosing Text Area from the drop-down list or from the menu by selecting HTML|Controls|Text Area... If you are using the latter method be sure to first position your cursor where you want the text area field inserted.
2 Enter the name of the text area field. If you have auto-label enabled in Preferences|Forms, the label will fill in with the name value. You can edit this if you want.
3 If you want the field to have a default value when the form is loaded, enter it in the value area.
4 Enter the rows (height) and cols (width) for the field.
5 Edit the label text if you want. If you do not have the auto-label feature enabled, you will not be able to enter into this field.
6 When you are finished, click the Ok button to insert the text field into the form (if you are using the Form Assistant) or your document. Click the Cancel button to leave the Text Area Assistant without inserting the field.

Text Field Assistant

Text fields are single line input fields that allow the user to type information such as their name, e-mail address etc. into a form. The Text Field Assistant is meant to help you place a text field in a form with the needed attributes set. The Assistant is available from the HTML|Control|Text Field... menu and from the Form Assistant.

The attributes usually used with text field are **name** and **size**. The name is returned when the form is submitted as the first in the name/value pair, while the value is the text that the user types. i.e. A field with a name of e-mail address might return: "e-mail address=webber@csdcorp.com". The size attribute sets the displayed size of the field, which should be large enough to accommodate the majority of expected values.

Other attributes sometimes used are: **value**, the initial value of the field when the form is loaded; **password,** sets the field to password mode which disguises the characters typed in as the user types; and **maxlength** which sets the maximum number of characters that can be entered into a text input field. If the value of MAXLENGTH is greater the the value of the SIZE attribute, the field should scroll appropriately. There is no default maximum.

To Use:

1 Open the Text Field Assistant from the Form Assistant Dialogue by choosing Text Field from the drop-down list or from the menu by selecting HTML|Controls|Text Field. If you are using the latter method, be sure to first position your cursor where you want the text field inserted.

2 Enter the name of the text field. If you have auto-label enabled in Preferences|Forms, the label will fill in with the name value. You can edit this if you want.

3 If you want the field to have a default value when the form is loaded, enter it in the value area.

4 Enter the size and maximum length for the field.

5 Edit the label text if you want. If you do not have the auto-label feature enabled, you will not be able to enter into this field.

6 Check the Password checkbox to set the field to password mode.

7 When you are finished, click the Ok button to insert the text field into the form (if you are using the Form Assistant) or your document. Click the Cancel button to leave the text Field Assistant without inserting the field.

Textarea

compatibility

<textarea></textarea>

Usage:

The text area is used to allow a user to type in multiple lines of text. It is usually used as a free-form message area. The content of the field is used as the default value. The rows and columns determine the size of the text area.

J This tag has javascript events associated with it. For details see the description of the Javascript event handlers.

(See Also: Form Assistant and Form Field Assistants)

Attributes:

Required

name - name for the form field corresponding to this element. Returned when the form is submitted as the first in the name/value pair.

rows - the height of the text area

cols - the width of the text area.

Optional

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 align [LEFT | CENTER | RIGHT] - Used to explicitly specify the horizontal alignment.

3 **disabled** [DISABLED] - Will result in the field being rendered as usual except that it can't be modified by users. It may 'look' disabled i.e. be grayed out. This is for read-only fields.

3 **error** - This attribute specifies an error message explaining why the field's current value is incorrect.

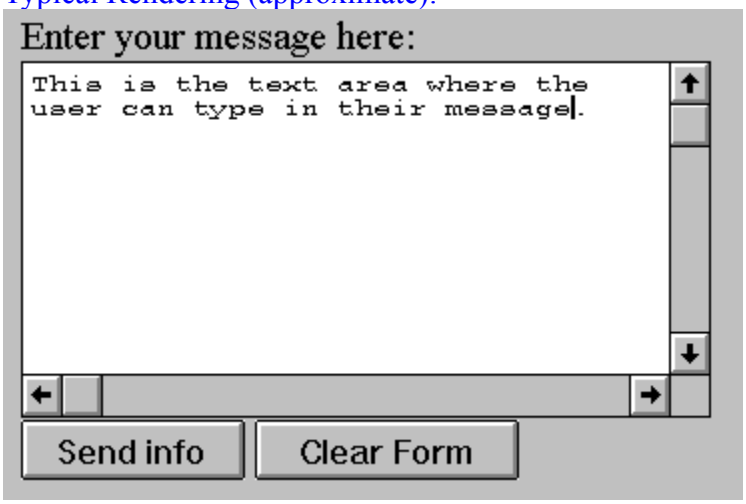
N **wrap** [OFF | VIRTUAL | PHYSICAL] - specify how to handle word-wrapping in text input areas in forms. The default is OFF, VIRTUAL means that the display word-wraps, but long lines are sent as one line without new-lines, PHYSICAL means that the display word-wraps, and the text is transmitted at all wrap points. This is new with Netscape 2.0.

Example:

```
<HTML>
<HEAD>
<TITLE>Example Form</TITLE>
</HEAD>
<BODY><FORM ACTION="http://www.somewhere.com/cgi-bin/mailscript"
METHOD="POST">
<P>Enter your message here:<BR>
```

```
<TEXTAREA NAME="message" ROWS=12 COLS=35>
</TEXTAREA>
<BR><INPUT TYPE="SUBMIT" VALUE="Send info" NAME="Send info">
<INPUT TYPE="RESET" VALUE="Clear Form">
</P></FORM></BODY></HTML>
```

Typical Rendering (approximate):



Tilde

compatibility

<tilde></tilde>

Usage:

This tag places a tilde (~) above the term enclosed. It is used in the <MATH> tag as an alternative to <ABOVE>.

Attributes:

Required

Optional

Example:

```
<TILDE>X</TILDE>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Title

compatibility

<title></title>

Usage:

The title is an important element of any HTML document and is often used by browsers for display in the title bar of the browser window. Web search engines often use the title along with the url to determine what a document is about, so be sure to choose a brief but descriptive title that will identify a document even out of context. For example this file would be "Webber Help File" rather than simply "Help File". There is no limit to the size of a title but using less than 64 characters will reduce the chances of truncation in the title bar.

Attributes:

Required

Optional

Example:

```
<html>
<head>
<title>The title of a very simple document.</title>
</head>
<body><P>This is the text of the very simple document.</P></body>
</html>
```

Typical Rendering (approximate):

Not applicable to document although many browsers will display the document title in the title bar of the window.

To Hide or Not To Hide

Whether or not to hide the HTML from the user (you) is a question of some debate. Generally, editors that hide the HTML from you are also WYSIWYG. We have chosen not to hide the HTML tags, and to give as much assistance and explanation of HTML and tagging documents as we can. We have also chosen not to use WYSIWYG editing. We made these decisions for several reasons.

1 In the end, no matter what tool you use to create your HTML document, you end up with a plain text file that contains your content and the HTML mark-up. It is completely human-readable. So why not be able to read it, learn it, and understand it as you create it? This way it will all make sense to you, and you will become more skilled at debugging problems as they occur.

2 The industry changes so quickly that it is difficult for any tool vendor to keep up - the HTML specification continues to evolve, and the vendors of the browsers continue to add their own touches to the tags and attributes. If you are using a tool which hides the HTML from you, you probably can't edit the attributes directly, or easily add tags not yet supported by that tool. Webber gives you the flexibility of being able to type in anything you want - you are only constrained by what the browsers support. While we will always make efforts to bring out new versions that support new features as quickly as possible, you will be able to use your current version of Webber until then.

3 The goal of Webber has always been to provide a powerful yet fast and flexible tool for creating documents for the World Wide Web. It is our belief that hiding the HTML from you would have compromised that goal.

4 WYSIWYG editors are actually deceiving. There is no standard way to display an HTML file. Browsers can handle the tags differently, and many now let the user decide what fonts and sizes to use for the elements of displayed documents. While a WYSIWYG editor is pretty to work in, it doesn't realistically show you what your document will look like to others. That being said, it is our intention to add character level formatting to a future version of Webber but only as an adjunct to the tag display which will remain intact .

Toolbar

The toolbar is detachable so that it floats freely on the screen, or dockable so that it attaches to the top, bottom, or side of the Webber window. To change the position of the tool bar, click on an open area of the bar, hold down the mouse button and drag it to the desired position.

You can also close the toolbar by making it into a floating window and clicking the close button, or by unchecking 'Show toolbar' from the options menu.

 New - Opens a new untitled document. Right click this button to choose between a new blank document and a new HTML document.
Keyboard Equivalent: Alt + F, N
Shortcut: Ctrl N
Right Click?: No

 Open - Opens a dialogue box allowing you to select a file (or multiple files) to open. Multiple files can be selected with standard Windows behaviour - i.e. holding down the ctrl key while clicking on multiple file names in sequence.
Keyboard Equivalent: Alt + F, O
Shortcut: Ctrl O
Right Click?: No

 Save - Saves the active file (i.e. the file whose window has the focus) to the name and directory indicated in the title bar of its window. If the file does not have a name yet, you will be prompted to enter one.
Keyboard Equivalent: Alt + F, S
Shortcut: Ctrl S
Right Click?: No

 Undo - Multiple level undo - reverses changes made to tags and text in reverse order. Large operations such as pasting over a selection of text are undone in one action.
Keyboard Equivalent: Alt + E, U
Shortcut: Ctrl Z
Right Click?: Yes

 Cut - Cuts the selected text and tags to the clipboard.
Keyboard Equivalent: Alt + E, T
Shortcut: Ctrl X
Right Click?: Yes

 Copy - Copies the selected text and tags to the clipboard.
Keyboard Equivalent: Alt + E, C
Shortcut: Ctrl C
Right Click?: Yes

Paste - Pastes the selected text and tags from the clipboard.
Keyboard Equivalent: Alt + E, P
Shortcut: Ctrl V
Right Click?: Yes

Find - Searches the document for a specified word, phrase or character(s). Type the word(s) you want to search for into the dialog or highlight a word(s), then click the search button and your selection is automatically picked up by the search dialog. Right click on this button to choose between find or find and replace.
Keyboard Equivalent: Alt + S, F
Shortcut: Ctrl F
Right Click?: No

Find Next - Repeats the most recent search.
Keyboard Equivalent: Alt + S, N
Shortcut: F3
Right Click?: No

Anchor - Opens the Anchor Assistant dialogue box which allows you to quickly add an anchor to your document and set the common attributes.
Keyboard Equivalent: Alt + T, A
Right Click?: No

Image - Opens the Image Assistant dialogue box which allows you to quickly add a picture or graphic to your document and to set its common attributes.
Keyboard Equivalent: Alt + T, I
Right Click?: No

List - Opens the List Assistant dialogue box which allows you to quickly add a lists and list items to your document.
Keyboard Equivalent: Alt + T, L
Right Click?: No

Table - Opens the Table Assistant dialogue box which allows you to set the size (number of rows and columns), attributes and caption for a table. From the Table Assistant you can open the Table Editor where you can set row and column spanning and enter your data into the cells of your table.
Keyboard Equivalent: Alt + T, I
Right Click?: No

Entity - Opens the Entity Assistant dialogue box which allows you to select character entities from a list, or pick one from a graphical representation.
Keyboard Equivalent: Alt + T, E

Right Click?: No

Tag Assistant - Opens the Tag Assistant dialogue box which allows you to select tags from a list, and shows you what attributes can be set for the tag.
Keyboard Equivalent: Alt + T, T
Right Click?: No

Preview - Launches the browser identified as the default in Preferences|HTML (if it is not already open) and automatically loads your document for previewing if this is supported by the browser. If you right-click on this button you can select from a list of all configured browsers.
Short Cut: Ctrl+B
Keyboard Equivalent: Alt + T, D, B
Right Click?: No

Validate - Starts the validation feature for checking that your HTML is valid.
Short Cut: No
Keyboard Equivalent: Alt + T, D, V
Right Click?: Yes

Help - Opens this help file at its contents.
Keyboard Equivalent: Alt + H, C
Shortcut: F1
Right Click?: No

Form Tutorial

Getting Prepared

The creation of a form is probably the most complicated set of tags you will deal with. In addition to the complexity of the tags, a form also needs to be able to call a server script. The script responds to the submit button and send the data from the submitted form to its destination. This tutorial will focus only on the creation of the form using HTML tags. You will need to discuss the script with the administrator of your Web server.

Webber helps reduce the complexity of creating a form by offering the Form Assistant. Start a new HTML document by selecting New HTML from the File Menu. As we learned in the Home Page tutorial this command will create a new untitled document with the basic structural elements of the document in place and the cursor, (in purple below), in position for you to type your document title.

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<HTML>
<HEAD>
<TITLE>|</TITLE>
</HEAD>
<BODY><HR><ADDRESS><!--Webber_Auto_StampDNR-->Page maintained by Name,
<A HREF="mailto:e-mail@company.com"</A>. Copyright(c) 1995
Companyname. Created: 01/06/95 Updated: <!--DS-->01/06/95<!---->
</ADDRESS></BODY>
</HTML>
```

Two elements of the example above may or may not be present depending on how your preferences are set. The Doctype (in red) is added if it is enabled in Preferences|HTML. The Stamp (in blue) is added if it is enabled in Preferences|User Info.

Type your document title, add a heading and a brief descriptive paragraph about the form. For example :

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<HTML>
<HEAD>
<TITLE>My First Form Example</TITLE>
</HEAD>
<BODY>
<H1>This is the heading for my first form</H1>
<HR>
<P>This is the descriptive paragraph for my first form.</P>
<HR><ADDRESS><!--Webber_Auto_StampDNR-->Page maintained by Name, <A
HREF="mailto:e-mail@company.com">e-mail@company.com</A>. Copyright(c)
1995 Companyname. Created: 01/06/95 Updated: <!--DS-->01/06/95<!---->
</ADDRESS></BODY>
</HTML>
```

Place your cursor between the end tag for the paragraph and the horizontal rule. Open the Form Assistant by selecting HTML|Form... from the menu. The form assistant helps you set up and

preview a form before adding it to your document. Form tags have many attributes that determine how the form is handled by the server and the types of input fields displayed on the form.
At the top of the form assistant you can fill in the ACTION, METHOD and ENCTYPE attributes. You will have to find out what values to give these attributes by discussing the purpose of the form with your server administrator. The action attribute usually specifies the script that will handle the form when it is submitted.

When a form is submitted by the user (usually by clicking a 'submit' button), the names (the NAME attribute) of the fields are passed to the script along with the values of the fields. For example a text input field named 'e-mail address' might return '`e-mail address=webber@csdcorp.com`', where *webber@csdcorp.com* was typed in by the user. These return values are referred to as a name/value pair. You must give each field on your form a unique name so that you know which field the returned value refers to. If you want the field to have a default value (or initial value) when the form is displayed to the user, you use the VALUE attribute to specify what the default value should be. Other attributes will be discussed with each field type.

The grid in the middle of the form assistant allows you to set the field types you want in your form, and allows you to add labels and line breaks. Double-clicking on the Type column will bring up a list of objects that can be inserted. Labels inserted here are separate from those which can be automatically generated as described in the next lesson.

Next - Text Fields, Line Breaks and Lists

Form Tutorial

Text Fields, Line Breaks and Lists

Each field type has an assistant to help you set the attributes and to label the field. These assistants are used in conjunction with the form assistant and can also be used on their own to add fields to an existing form. Field labels can be generated automatically by the assistants if you have your preferences configured to do so. You can set whether you want the label to appear before or after the field, or not at all. If you select None you will have to enter labels in the form assistant.

3

Double-click in the first cell of the Type column. Select Text Field from the list. The Text Field Assistant will appear.
Enter the name of the field. The name will automatically fill in to the label box if auto-label is enabled. The name of an input field is returned by the form when it is submitted to identify the value, it is not displayed to the user. For our example we will use 'company' as the NAME.
The value field should be filled in only if you want the field to have an initial value when the form is loaded. Otherwise the field will be blank when the form is loaded. We will leave this blank for the example.
The size refers to the number of characters that will fit into the input area without scrolling. Select a size big enough to accommodate the majority of entries. This is an optional attribute, if you do not set one the field will default to a reasonable size for you. We will use 30 for our example.
The maximum length attribute is also optional. You can use it to limit the number of characters returned by a field. This can be useful if you intend to import the returned data into database where you have fixed length fields. Enter a number into this box if you want to have a maximum limit. We will use 50 for our example.
The password attribute disguises the input that the user types as they type it into the field. Check this box if you wish to use a password. We will not use the password attribute for our example.
Edit the label if you are using auto-label. If not labels will have to be added from the form dialogue or directly into your document. We will change the label to read 'Company Name : ' for our example.

Click the Ok button to add this field to the form. In the Form Assistant you will see the field displayed as a line item in the grid. The type, name, length and value for the field will be shown so that you can identify it in order to edit it if necessary. To edit a field, select the field and click the Edit button.

In the cell below the text field, double-click to get the list of fields again. Select Line Break by double-clicking it from the list. This will insert a <BR> into the form.

Below the line break choose Select from the list. The Select Assistant will appear. The select tag allows you to add a selection list to your form. We will add a list of Countries so that the user can select his/her place of connection.

For the name of the field we will use 'country'. We will set the size to 1. This means that the list will show only one value at a time. The Multi-select Allowed check box adds the MULTIPLE

attribute which means that the user can select more than one value on the list. We will leave this check box unchecked for this example.
In the first cell of the selection text we will type Canada. Enter U.K., U.S.A., Japan, and Europe below. You can change the order of the selection options by clicking in the left-most cell of a row, holding down the mouse button and dragging it to a new location. Place the options on your list in alphabetical order. The selection text is what will appear to the user when the form is loaded. The selection made by the user will be returned by default when the form is submitted. If you want a value other than the selection option to be returned you can use the 'return value' column which sets the VALUE attribute for the option. For our example we will leave the return value blank.
You can also set which option (if any) you want to be selected as a default when the form is loaded. To do so click once in the 'selected' column beside desired option. We will set U.S.A. to be the default for our example form. A 'x' will appear in the cell of the default choice.

Click the Ok button when you are finished.

We will be adding more fields in the following lessons but if we were to generate the form at this time the resulting HTML would look something like this:

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<HTML>
<HEAD>
<TITLE>My First Form Example</TITLE>
</HEAD>
<BODY>
<H1>This is the heading for my first form</H1>
<HR>
<P>This is the descriptive paragraph for my first form.</P>
<FORM ACTION="http://www.company.com/cgi-bin/script?script"
METHOD="POST"><P>
Company : <INPUT TYPE="TEXT" NAME="company" SIZE="30" MAXLENGTH="50">
<BR>Country : <SELECT NAME="country" SIZE="1"><OPTION>Canada</OPTION>
<OPTION>Europe</OPTION>
<OPTION>Japan</OPTION>
<OPTION>U.K.</OPTION>
<OPTION SELECTED>U.S.A.</OPTION>
</SELECT>
</P></FORM><HR><ADDRESS><!--Webber_Auto_StampDNR-->Page maintained by
Name, <A HREF="mailto:e-mail@company.com">e-mail@company.com</A>.
Copyright(c) 1995 Companyname. Created: 01/06/95 Updated: <!--DS--
>01/06/95<!----> </ADDRESS></BODY>
</HTML>
```

So that you don't have to generate the form until we are happy with it, you can preview it from the Form Assistant.
3 Click the test button at the bottom of the dialogue. A temporary file will be created and displayed to you by your browser. This allows you to preview your form in progress, before you add it to your document. If there are problems with the layout, labeling or display of the form you can use the edit button to modify the fields, or add line breaks and labels where appropriate.

Add another line break below the selection list now.

Next - Radio Buttons & Check Boxes

Form Tutorial

Radio Buttons and Check Boxes

In the cell below the line break select Radio Button from the list. The Radio Buttons Assistant will pop up. When several input fields with a type of radio share the same NAME, they will form a set of radio buttons. When one radio button of a set is turned on the others will be turned off. The Radio Buttons Assistant allows you to specify a set of buttons or a single button.

For this example we will create a set of radio buttons that a user could use to tell you how they would like to receive a response from you. In the name field we will enter 'response'. In the label field we will enter 'Please respond by : '. This label will label be used for the set of buttons - we will specify labels for the individual buttons separately.

Double-click in the first cell and select Radio Button from the list. Enter the value of the button and a label for it. Click Ok to add the button to the list. Create several more buttons in the same way. For this example we will use the following for our buttons (value/label): email/E-mail, snail/Snail Mail, fax/Fax, phone/Phone. We should now have a list of four buttons. We will add line breaks between each so that they appear on separate lines. In the fist empty cell select Line Break from the list. Click on the row marker beside the line break, hold down the mouse button and drag it to a position before the first button. Add line breaks after each radio button in the same way.

One radio button in a set can be turned on by default when the form is loaded. To set the default button, click once beside one button in the row header column. A 'x' will appear beside the button. Only one button in a set can be the default. For this example we will check the email button as the default.

Click the Ok button to add this set of radio buttons to the Form Assistant. Once again we won't generate the form yet but if we did the resulting HTML for the radio buttons would look like this:

```
Please Respond By : <BR>
<INPUT TYPE="RADIO" NAME="response" VALUE="e-mail"> E-mail<BR>
<INPUT TYPE="RADIO" NAME="response" VALUE="fax"> Fax<BR>
<INPUT TYPE="RADIO" NAME="response" VALUE="snail"> Snail Mail<BR>
<INPUT TYPE="RADIO" NAME="response" VALUE="phone"> Phone<BR>
```

Use the test button again to preview your form.

We will now add a check box to the form. Select Check Box from the list in the Form Assistant. The Check Box Assistant will pop up. We will provide a check box for the user to indicate whether or not they want to be placed on your mailing list. As the name of the check box, type 'maillist'. For the value type 'yes'. This value will be returned when the check box is checked and the form is submitted (i.e. 'maillist=yes'). For the label, type 'Add me to your mailing list'. When you are finished, click the Ok button to return to the Form assistant. Add another line break after the check box.

The html generated by the check box assistant will look like this:

```
<INPUT TYPE="CHECKBOX" NAME="maillist" VALUE="yes" CHECKED="CHECKED">
Add me to your Mailing List<BR>
```

Preview your form again. You can add additional space between types of fields by adding extra line breaks.

Next - Text Areas, Buttons and Previewing

Form Tutorial

Text Areas, Submit and Reset Buttons, Previewing

Large message areas that a user can type a message into are called text areas. From the list select Text Area. The Text Area Assistant will appear.
As the name of the text area we will use 'message'. The value need only be filled in if you want the text area to have a default value when the form is loaded. For this example we will leave it blank.
The rows and columns attributes determine the display size of the field. Rows refers to the number of lines of characters that will fit vertically within the text area. Type 8 into the Rows box. Columns refers to the number of characters that will fit side by side within the text area. Enter 50 into the Columns box.
As the label we will use "Type your message for us here : ".

When you are finished click the Ok button to return to the form assistant. Add a line break below the text area.

The HTML generated by the text area assistant in this example would look like this:

```
Type your message for us here : <BR>
<TEXTAREA NAME="message" ROWS="8" COLS="50"></TEXTAREA><BR>
```

Preview your form once more.

Now that we have a basic form put together we need a way for the user to submit it. Select Submit Button from the list. The Submit Button Assistant will pop up. The name of the button is optional since it is not usually important to get a return value from a button. You might want to use a name if there is more than one submit button on a form and you need to know which was clicked by the user. If you leave the name attribute off and just fill in the value you will not receive a return from the button but it will submit the values from the other fields. The value attribute is also used as the label for the button. For this example we will not use a name and will enter 'Send Message' as the value.

When you are finished click the Ok button and return to the form assistant. Preview your form again to view your button. The HTML added for the Submit Button will look like this :

```
<INPUT TYPE="SUBMIT" VALUE="Send Message">
```

Another button commonly placed on forms is the reset button. It does not return a value and is used only to reset a form to its initial (default) values. Select Reset Button from the list. the Reset Button Assistant will appear. Because this button does not return a value, it does not have a name. We will use 'Clear Form' as the value.

When you are finished click the Ok button to return to the form assistant. The HTML added for the reset button will look like this:

```
<INPUT TYPE="RESET" VALUE="Clear Form">
```

We are finally ready to generate our form! Preview it one more time to make sure that you have line breaks and labels where you want them and that the fields appear in the order that you

want. We will add an additional line break before the Submit Button by selecting Line Break in the first empty cell and dragging it into position.

When you are satisfied click the Ok button on the Form Assistant to insert the tags for the form into your document. Our example form will look like this:

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<HTML>
<HEAD>
<TITLE></TITLE>
</HEAD>
<BODY>
<FORM ACTION="http://www.company.com/cgi-bin/script?script"
METHOD="POST"><P>
Company : <INPUT TYPE="TEXT" NAME="company" SIZE="30" MAXLENGTH="50">
<BR>Country : <SELECT NAME="country" SIZE="1"><OPTION>Canada</OPTION>
<OPTION>Europe</OPTION>
<OPTION>Japan</OPTION>
<OPTION>U.K.</OPTION>
<OPTION SELECTED>U.S.A.</OPTION>
</SELECT><BR>
Please Respond By : <BR>
<INPUT TYPE="RADIO" NAME="response" VALUE="e-mail" CHECKED="CHECKED">
E-mail<BR>
<INPUT TYPE="RADIO" NAME="response" VALUE="fax"> Fax<BR>
<INPUT TYPE="RADIO" NAME="response" VALUE="snail"> Snail Mail<BR>
<INPUT TYPE="RADIO" NAME="response" VALUE="phone"> Phone<BR>
<INPUT TYPE="CHECKBOX" NAME="maillist" VALUE="yes" CHECKED="CHECKED">
Add me to your Mailing List<BR><BR>
Type your message for us here : <BR>
<TEXTAREA NAME="message" ROWS="8" COLS="50"></TEXTAREA><BR><BR>
<INPUT TYPE="SUBMIT" VALUE="Send Message"><INPUT TYPE="RESET"
VALUE="Clear Form">
</P></FORM>
<HR><ADDRESS><!--Webber_Auto_StampDNR-->Page maintained by Name, <A
HREF="mailto:e-mail@company.com">e-mail@company.com</A>. Copyright(c)
1995 Companyname. Created: 01/06/95 Updated: <!--DS-->28/06/95<!---->
</ADDRESS></BODY>
</HTML>
```

This form would be rendered something like this:

Company :

Country : U.S.A.

Please Respond By :

- ◉ E-mail
- ○ Fax
- ○ Snail Mail
- ○ Phone
- ☒ Add me to your Mailing List

Type your message for us here :

Send Message | Clear Form

Page maintained by Name, e-mail@company.com. Copyright(c) 1995 Companyname. Created: 01/06/95 Updated: 28/06/95

Congratulations! You have reached the end! You can now use the form control assistants to add fields directly to the document within the form. Try adding fields for the name and e-mail address of the user. Remember that any tag you insert will be placed in the document at the current cursor position. Good luck!!

Home Page Tutorial

Document Essentials

To use Webber to create a home page, you will need to understand the basics of HTML. Some HTML editors hide the HTML tags from you, Webber does not. For a discussion of our philosophy, click here.

HTML documents are simple text documents with HTML tags marking up the content. HTML stands for Hypertext Mark-up Language. The tags are the mark-up which defines how a document should be displayed by a browser. There are several types of tags - document level tags, which describe the document and it's relationship to others, and content level tags which describe how the content should be presented.

The document level tags include the HTML identifier, the HEAD, and the BODY. Every HTML document must have these components. Webber makes it easy for you to add the required elements of an HTML document. From the menu, select File|New HTML. A basic HTML document outline will be created in a new untitled document.

```
<HTML>
<HEAD><TITLE>|
</TITLE></HEAD>
<BODY>
</BODY></HTML>
```

You will notice that the cursor (in purple above) is inside the TITLE tag, ready for your input. Every HTML document must have a title. The title is typed between the start tag for the title, <TITLE> and the end tag </TITLE>. Type the title of your home page. Your title should uniquely identify your document since it is often used by Web search engines for indexing purposes.

```
<HTML>
<HEAD><TITLE> The title of your document goes here.
</TITLE></HEAD>
<BODY>
</BODY></HTML>
```

Hard returns, tabs and extra spaces that you put into an HTML document are ignored by browsers. This means that the example above will be rendered in exactly the same way as this:

```
<HTML><HEAD><TITLE> The title of your document goes
here.</TITLE></HEAD><BODY></BODY></HTML>
```

You may use either method but separating the different elements of your document with hard returns can make it easier to see the components and to edit them at a later date.

Next - The Page Heading & Logo

Home Page Tutorial

Document Heading & Logo

Now that you have created the basic outline for your home page, it's time to put in some content. The first element we will use is a heading. Place your cursor after the start tag for the body. From the drop-down list on the tool bar, select `H1, Heading 1.` Don't forget that you can rearrange the tags on this list to suit your preferences.

```
<HTML>
<HEAD><TITLE> The title of your document goes here.
</TITLE></HEAD>
<BODY><H1>|</H1>
</BODY></HTML>
```

The tag for a first level heading will be inserted into your document with the cursor positioned for input. The first heading should tell the reader what the page is about. Type your heading between the <H1> and the </H1> (the start and end tag for the heading level 1).

```
<HTML>
<HEAD><TITLE> The title of your document goes here.
</TITLE></HEAD>
<BODY><H1>My First Home Page</H1>
</BODY></HTML>
```

Beside the heading you may want to put in your company logo or a picture of yourself. Place your cursor after the start tag for the heading.

Click the image Quick-Tag button on the tool bar. This will open the Image Assistant. The Image Assistant is one of many tag assistants available in Webber. Enter the name of the image or it's URL. If the graphic will be in the same directory as the document all you need is the name with no path.

```
<HTML>
<HEAD><TITLE> The title of your document goes here.
</TITLE></HEAD>
<BODY><H1><IMG SRC="image.gif"> My First Home Page</H1>
</BODY></HTML>
```

Image tags are empty tags - meaning that they do not contain content and therefore do not have end tags.

Next - Content & Previewing

Home Page Tutorial

A Start on Content

Now we are ready for the most important thing about any Web document - the content! We will start with a paragraph. Place your cursor after the end tag for the heading. From the drop-down tag list select `P, Paragraph`.

```
<HTML>
<HEAD><TITLE> The title of your document goes here.
</TITLE></HEAD>
<BODY><H1><IMG SRC="image.gif"> My First Home Page</H1>
<P>|</P>
</BODY></HTML>
```

The paragraph tag is inserted with the cursor inside ready for your input. Type a brief description about the purpose for the page between the start and end tags for the paragraph. Remember that tabs and extra spaces are ignored in HTML. All formatting of a document is handled by the tags - not the keyboard!

```
<HTML>
<HEAD><TITLE> The title of your document goes here.
</TITLE></HEAD>
<BODY><H1><IMG SRC="image.gif"> My First Home Page</H1>
<P>This is a brief description of my home page.</P>
</BODY></HTML>
```

Save your document by selecting File|Save from the menu. Let's have a look at the page by clicking on the document preview button on the tool bar. If you have not set up the path to your browser you will have to so in order for the preview to work. So far the page should look something like this :

My First Home Page

This is a brief description of my home page.

If your page does not display correctly (or at all), don't despair! Go back through the previous examples and make sure that all the tags (including the end tags) are present and in the right order. You should also look for missing angle brackets around the tag names.

If your page looks great, congratulations! You can move on to the next step.

Next - Horizontal Rules & Lists

Home Page Tutorial

Horizontal Rules & Lists

After previewing your page you might decide that you would like a line between the heading and the paragraph. In HTML these are called horizontal rules. Place your cursor between the end tag for the heading and the start tag for the paragraph. You can put in a hard return to make it easier to read.

```
<HTML>
<HEAD><TITLE> The title of your document goes here.
</TITLE></HEAD>
<BODY><H1><IMG SRC="image.gif"> My First Home Page</H1>
|
<P>This is a brief description of my home page.</P>
</BODY></HTML>
```

From the drop-down list select `HR, Horizontal Rule`. The tag will be inserted into your document with the cursor after it. Note that the <HR> tag is an empty tag with no end tag.

```
<HTML>
<HEAD><TITLE> The title of your document goes here.
</TITLE></HEAD>
<BODY><H1><IMG SRC="image.gif"> My First Home Page</H1>
<HR>|
<P>This is a brief description of my home page.</P>
</BODY></HTML>
```

Now let's add a list below the description paragraph. This list could link to other pages on the Web, it could list the products and services of your company, or the features of your product. Place your cursor between the end tag for the paragraph and the end tag for the body.

```
<HTML>
<HEAD><TITLE> The title of your document goes here.
</TITLE></HEAD>
<BODY><H1><IMG SRC="image.gif"> My First Home Page</H1>
<HR>
<P>This is a brief description of my home page.</P>
|
</BODY></HTML>
```

Open the Tag Assistant by selecting HTML|Tags... from the menu. Click the Tags in Context checkbox at the bottom left of the dialogue. From the drop-down list select `UL, Unordered List`. Click the Insert button. If you position the dialogue so that you can see your document you will notice that the <UL> tag has been inserted with the cursor positioned between the start and end tags. Click on the drop-down list again. You will notice that only the List Item is available for selection. This is because of the Tags in Context setting. It is telling you that only the <LI> tag is allowed inside the unordered list. Select the `LI, List Item` and click the Insert button. The list item will be inserted inside the unordered list. Click the Close button to close the Tag Assistant.

```
<HTML>
<HEAD><TITLE> The title of your document goes here.
</TITLE></HEAD>
<BODY><H1><IMG SRC="image.gif"> My First Home Page</H1>
<HR>
<P>This is a brief description of my home page.</P>
<UL><LI>|</LI>
</UL>
</BODY></HTML>
```

Unordered lists are usually rendered as bulleted lists. You can change your list to be a numbered list by changing the UL to an OL which is the tag for an ordered list. Add two more list items by copying and pasting the <LI></LI>.

```
<HTML>
<HEAD><TITLE> The title of your document goes here.
</TITLE></HEAD>
<BODY><H1><IMG SRC="image.gif"> My First Home Page</H1>
<HR>
<P>This is a brief description of my home page.</P>
<UL><LI></LI>
<LI></LI>
<LI></LI>
</UL>
</BODY></HTML>
```

Add content to the lists by typing it between the start and end tag for each list item.

```
<HTML>
<HEAD><TITLE> The title of your document goes here.
</TITLE></HEAD>
<BODY><H1><IMG SRC="image.gif"> My First Home Page</H1>
<HR>
<P>This is a brief description of my home page.</P>
<UL><LI>Item one in the list</LI>
<LI>Item two in the list</LI>
<LI>Item three in the list</LI>
</UL>
</BODY></HTML>
```

We now have document with a heading, a paragraph, and a bulleted list containing three items. Save the document by clicking the save button on the tool bar. Click the preview button to reload the document in your browser. It should now look something like this:

3 My First Home Page

This is a brief description of my home page.

- This is item one in the list
- This is item two in the list

- This is item three in the list

Lists can be nested inside one another to make multi-level outline type lists. There are three other types of lists. The directory list and the menu list are very similar to the unordered list and use the <LI> tag for their list items. The definition list contains a one line definition term and an indented multi-line definition description.

Besides nested list tags, list items can also contain paragraph tags. Putting a paragraph tag inside the list item and around your content will increase the space between items when the document is displayed by some browsers. Place paragraphs tags around the content of your list items by selecting the content of one item and selecting `P, Paragraph` from the drop-down list on the tool bar. This method of 'surrounding' text and then selecting a tag, places the start tag at the beginning of the selected text and the end tag at the end of the selected text.

```
<HTML>
<HEAD><TITLE> The title of your document goes here.
</TITLE></HEAD>
<BODY><H1><IMG SRC="image.gif"> My First Home Page</H1>
<HR>
<P>This is a brief description of my home page.</P>
<UL><LI><P>Item one in the list</P></LI>
<LI><P>Item two in the list</P></LI>
<LI><P>Item three in the list</P></LI>
</UL>
</BODY></HTML>
```

You can save and preview your document again to see the difference in rendering that this change has made. Some browsers may have already added space between each list item. If this was the case with your browser and you do not want space between list items, you can use the COMPACT attribute. This attribute is used with some types of lists to tell the browser to render the list in as compact manner as possible. The attribute looks like this : `<UL compact="compact"></UL>`. You can use the Tag Assistant to insert tags with attributes set.

Next - Anchors, Signatures & Validation

Home Page Tutorial

Anchors, Signatures & Validation

Arguably the most important aspect of HTML is the ability to hyperlink documents to other documents. We will be creating another sample document in the tutorial for building a form so let's create an anchor to it now. Select the text that is in your first list item. Click the anchor button on the tool bar. The Anchor Assistant will pop up. The anchor assistant allows you to set either of the two most important attributes for anchors - HREF and NAME. We will be using the HREF attribute.

If we were linking to a document on another Web server we would need to identify the protocol that the server should use to find the object by selecting one from the drop-down list. The most common protocols are HTTP, MAILTO, and #. The # is used when linking to a specific place in the current document. If the object being linked to is to be found on the same server as your document you do not need the protocol. If it is to be in the same directory, you need only the file name. These are called 'relative' links because the object is found in a location relative to the current location.

We will use a relative link here and identify only the file name. This will enable you to test the success of your anchor. In the HREF text box to the right of the drop-down list, type the name of an HTML file to be found in the same directory as this tutorial file. If you do not have any HTML file, type `FORMTUT.HTM`. We will use this name for our example in the form tutorial. Click the Ok button to insert the anchor tag.

```
<HTML>
<HEAD><TITLE> The title of your document goes here.
</TITLE></HEAD>
<BODY><H1><IMG SRC="image.gif"> My First Home Page</H1>
<HR>
<P>This is a brief description of my home page.</P>
<UL><LI><P><A HREF="formtut.htm">Item one in the list</A></P></LI>
<LI><P>Item two in the list</P></LI>
<LI><P>Item three in the list</P></LI>
</UL>
</BODY></HTML>
```

You can add anchors to other documents from the other list items in the same way. Save and preview your document. Depending on how you have your browser configured, the text located between the anchor start tag and the anchor end tag should be in a different colour and possibly underlined.

We are almost finished our home page now. A valuable aspect of any page is the 'signature'. This is an optional element but it is important to let people know who created the page and how they can get in touch with you. It is also useful for declaring a copyright for your material and letting people know when your document was created and/or last updated.

Webber helps you to add a signature in a couple of ways. If you have enabled it in your preferences, a stamp will be added automatically to new html and converted documents. You can also add one to any document from the menu. First make sure that you have filled in the

information for your stamp in the preferences. You can use your name, your e-mail address, a copyright notice, and creation and update dates and times in your stamp. These elements can be used singly or in any combination. HTML can be included in the text fields to automatically create a hyperlink. For example, you can place an anchor tag around your e-mail address like this: `<A HREF="mailto:email@address.com">email@address.com</A>`. Your e-mail address will be automatically hyperlinked to the MAILTO protocol. When a user clicks on your e-mail address, the mail dialogue of their browser will pop-up ready to send a message to you. After you have checked off the elements that you want inserted with your stamp, you can save them by clicking Ok to leave the preferences dialogue.

Place your cursor between the end tag for the list </UL>, and the end tag for the body </BODY>. From the menu select HTML|Stamp.

```
<HTML>
<HEAD><TITLE> The title of your document goes here.
</TITLE></HEAD>
<BODY><H1><IMG SRC="image.gif"> My First Home Page</H1>
<HR>
<P>This is a brief description of my home page.</P>
<UL><LI><P><A HREF="formtut.htm">Item one in the list</A></P></LI>
<LI><P>Item two in the list</P></LI>
<LI><P>Item three in the list</P></LI>
</UL>
<HR><ADDRESS><!--Webber_Auto_StampDNR-->Page maintained by Name, <A
HREF="mailto:email@address.com"> email@address.com</A>. Copyright(c)
1995 Companyname. Created: 01/06/95 Updated: <!--DS-->01/06/95<!---->
</ADDRESS>
</BODY></HTML>
```

If you have chosen to add the date or time updated, Webber will automatically refresh these with the current date and time when the document is saved. If you want this to happen you must not remove any of the comments placed in the stamp. You can modify the text of the stamp and the tag used for the stamp if you wish.

Save your document and preview the result. If there are any problems, go back through the examples in the previous lessons.

Webber also has a validation feature that allows you to check your document to make sure that it is compliant. The validator will also check for errors such as missing angle brackets and start or end tags. To use the validator, right-click and select Validate Document from the pop-up menu. This option is also available from the HTML|Document menu.

The validator will go through your document searching for errors. If it does not find any, you will see a message in the status line that says "This document is valid".

If errors are found, the validator will display a dialogue box with the text of the error message. The place where the error occurs in your document will be highlighted. You can correct the error and click the Next button to go on to the previous error or the Restart button to check the whole document again.

Congratulations! You have created a Home Page using Webber!. Your final document will look

something like this:

```
<HTML>
<HEAD><TITLE> The title of your document goes here.
</TITLE></HEAD>
<BODY><H1><IMG SRC="image.gif"> My First Home Page</H1>
<HR>
<P>This is a brief description of my home page.</P>
<UL><LI><P><A HREF="formtut.htm">Item one in the list</A></P></LI>
<LI><P>Item two in the list</P></LI>
<LI><P>Item three in the list</P></LI>
</UL>
<HR><ADDRESS><!--Webber_Auto_StampDNR-->Page maintained by Name, <A
HREF="mailto:email@address.com"> email@address.com</A>. Copyright(c)
1995 Companyname. Created: 01/06/95 Updated: <!--DS-->01/06/95<!---->
</ADDRESS>
</BODY></HTML>
```

Save and preview your page once more. It will be rendered something like this:

My First Home Page

This is a brief description of my home page.

- This is item one in the list
- This is item two in the list
- This is item three in the list

Page maintained by Name, email@address.com. Copyright(c) 1995 Companyname. Created: 01/06/95 Updated: 01/06/95

To go on to the Form Tutorial click here. To go back the beginning of the tutorial click here.

uri
Universal Resource Identifier - the address or location of a document.

Underline

compatibility

<u></u>

Usage:

This tag specifies that the enclosed text should be displayed as underlined. It is used within a 'container' tag such as a paragraph.

Attributes:

Required

Optional

id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

class - Used to create more specific types (or classes) of an existing generic tag.

clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
</HEAD>
<BODY>
<H1>Other information about HTML.</H1>
<P>There is lots of information to be had about <U>HTML</U> on the
Internet.</P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Unordered List

compatibility

**
**

Usage:

HTML supports several types of lists which may be nested within themselves or each other. The unordered list is similar (with some browsers identical) to the directory list and menu list. It contains multiple lined list items which usually result in left aligned, bulleted paragraphs.

HTML 3.0 gives you the ability to customize the bullets, to do without bullets and to wrap list items horizontally or vertically for multi-column lists.

Attributes:

Required

Optional

compact - [COMPACT] - suggests that the list be rendered with a minimum of white space, not all browsers make use of this attribute.

3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

3 clear - This attribute allows you to move down so that the text contained in this tag will start below a table or figure rather than beside it.

3 **wrap** [VERT | HORIZ | NONE] - Used for multicolumn lists. Use wrap=vert if you want to arrange the list items down the page before wrapping to the next column. Use wrap=horiz if you want to arrange the items across the page (less useful).

3 **plain** [PLAIN] - this attribute suppresses the display of bullets.

3 dingbat - Used to specify an iconic image to appear as a bullet.

3 src - Used to specify an image to appear as a bullet.

3 md - Specifies a message digest used when you want to be sure that a linked graphic, specified by the SRC attribute, is the same one intended, and hasn't been changed in any way.

N **type** [CIRCLE | DISC | SQUARE] - Used to change the bullet type for a list items. This is handled by the dingbat and src attributes in HTML 3.0.

Example:

```
<HTML><HEAD>
<TITLE>Index</TITLE>
```

```
</HEAD>
<BODY>
<H1>Index</H1>
<UL>
<LI>Ch. 1</LI>
<LI>Ch. 2</LI>
<LI>Ch. 3</LI>
<LI>Ch. 4</LI>
</UL>
</BODY><HTML>
```

Typical Rendering (approximate):

Index

- Ch. 1
- Ch. 2
- Ch. 3
- Ch. 4

Validation

Shortcut - right click for pop-up menu

What is Validation?

Validation is the process of Webber checking your document against the rules laid out in the HTML specification. Which version of the specification is checked depends on which dtd you have selected. You can check the status line to see which dtd you are using. (HTML 3.0 support is based on a draft specification, not all extension elements have been adopted yet.) You may wonder why you should validate - in short, because you will be ensuring that the documents you are creating will always be supported.

If the document is valid, Webber will tell you that by posting a message in the status line. A valid document just means that it correctly follows the rules laid out in the HTML specification. Validating also allows you to check for missing angle brackets, end tags and other mark-up elements that could mean your document won't display correctly. Sometimes missing angle brackets are valid according to the dtd but still cause problems for browsers. You can use the Bracket Matcher feature to help find them.

If there are problems Webber will tell you where they are and give a message to help point you at the problem.

How do I Validate?

1 Select Validate from the HTML|Document menu or from the popup menu when you right click in a document.
2 This will either bring up the 'Validation Results' dialog box (booh!) or a message on the status line that the document is valid (yeah!).
3 If you got the message on the status line you're done (now wasn't that easy?). The rest of you have to stay after class and study a little harder...
4 The 'Validation Results' dialog has a problem description area with a brief message describing the first problem encountered.
5 Webber will also position your document to the offending line.
6 The Prior and Next buttons on the dialog will take you to the prior or next problem description. You can continue to use all the features of Webber to fix problems encountered. *(see Strategies for fixing invalid documents)*
7 At any time you can use the 'Restart' button on the 'Validation Results' dialog to validate the document again and see if your repairs have been successful.
8 You are finished when you get a message saying that your document is valid.

You can also try the Validation Tutorial.

Validation Messages

The most common messages generated when validating a document are shown below. A brief description of the messages are included in a pop-up window when you click on each message. While there may be messages which are not included on this list, these are the most common and should help you understand the general nature of errors that the validator finds. Some errors are 'fatal' and will end the validation (parse) of the document. Others are just warnings and the problem is ignored while the validation continues.

Try using the 'Tags in Context' feature of the tag assistant, angle bracket matching. You can also check this topic on Allowed Tags and the dtd Quick Reference.

1 Out-of-context data ended **tagname** document element (and parse)
2 Out-of-context **tagname** start-tag ended **tagname** document element (and parse)
3 Non-SGML character ignored
4 Non-SGML character found; should have been character reference
5 Incorrect character in markup; markup terminated
6 **tagname** end-tag ignored: doesn't end any open element (current is **tagname)**
7 Start-tag omitted from **tagname** with empty content
8 No element declaration for **tagname** end-tag GI; end-tag ignored
9 **attribute name** = "**attribute value**" attribute ignored: not defined for this element
10 **attribute value** = "**attribute value**" attribute value defaulted: invalid character
11 **attribute name** = "**attribute value**" attribute value defaulted: token too long
12 **attribute name** = "**attribute value**" attribute value defaulted: token not in group
13 Required **attribute value** attribute was not specified; may affect processing
14 **tagname** end-tag implied by **tagname** end-tag; not minimizable
15 **tagname** end-tag implied by **tagname** start-tag; not minimizable
16 **tagname** end-tag implied by data; not minimizable
17 **tagname** end-tag implied by short start-tag (no GI); not minimizable
18 **tagname** start-tag implied by tagname start-tag; not minimizable
19 **tagname** start-tag implied by data; not minimizable
20 **tagname** start-tag implied by short start-tag (no GI); not minimizable
21 Possible attributes treated as data because none were defined
22 **tagname** element ended prematurely; required **tagname** omitted
23 **tagname** element ended prematurely; required subelement omitted
24 No declaration for entity "**entity name**"; reference ignored
25 Short end-tag (no GI) ignored: no open elements
26 Short start-tag (no GI) ended **tagname** document element (and parse)
27 Start tag of document element omitted; not minimizable
28 Length of name, number, or token exceeded NAMELEN or LITLEN limit
29 **tagname** start-tag implied by short start-tag (no GI); not minimizable
30 **tagname** start-tag exceeds open element limit; possible lies from **tagname** on
31 **tagname** element not allowed at this point in **tagname** element
32 Data not allowed at this point in **tagname** element

Variable

compatibility

<var></var>

Usage:
The variable tag indicates that the text is a placeholder. It is usually rendered in italics.

Attributes:
Required

Optional
3 id - ID - An SGML identifier - NAME token - which must be unique within the scope of the current document.

3 lang - Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc.

3 class - Used to create more specific types (or classes) of an existing generic tag.

Example:

```
<HTML><HEAD>
<TITLE>Home Page Template</TITLE>
</HEAD>
<BODY>
<H1>Home Page Template</H1>
<HR>
<H2><VAR>Place your Company Name here.</VAR></H2>
<P><VAR>Place your company blurb here.</VAR></P>
<HR>
<ADDRESS><VAR>Place your e-mail address here.</VAR></ADDRESS>
</BODY></HTML>
```

Typical Rendering (approximate):

Home Page Template

Place your Company Name here.

Place your company blurb here.

Place your e-mail address here.

Vec

compatibility

<vec></vec>

Usage:

This tag places a right arrow (-->) above the term enclosed. It is used in the <MATH> tag as an alternative to <ABOVE>.

Attributes:

Required

Optional

Example:

```
<VEC>X</VEC>
```

Typical Rendering (approximate):

Since this tag is not yet commonly supported by browsers, we can not display a typical rendering at this time.

Webber as a DDE Server

Webber can act as a DDE server which allows it to be called from other applications including File Manager in Windows NT, and Windows Explorer in Windows 95. Setting Webber up as a DDE server for .htm/.html documents will allow you to launch Webber and open HTML files by double-clicking on them in File Manager/Explorer. It will also allow you to use the Edit|Current Page item in Microsoft Internet Explorer to pass files from Explorer to Webber for editing. Webber's DDE interface also allows developers to call Webber from their applications.

Windows 95 allows you to associate multiple applications that can handle files of a particular type. For example, you could have Netscape, Internet Explorer and Webber all set up as servers for .htm/.html files. You can also have Webber set up to both open and edit files.

Setting up Webber as a DDE server in Windows 95 and NT 4.0:

1 Open Windows Explorer.
2 Select View|Options from the menu.
3 Click on the 'File Types' tab.
4 If you have a browser such as Netscape or Internet Explorer installed on your system, there will probably be a definition for .htm/.html files. Scroll down the list of file types to find the association for HTML files. It might be called 'Hypertext Document', or 'Netscape Hypertext Document', or 'Internet Explorer Document'.
5 Select the definition and click 'Edit'. If you can't find a definition for HTML files, click the 'New Type...' button to create one.
6 Give the type a description if it does not have one, or edit the existing description if desired. The description is what will appear beside the document name in Windows Explorer under 'type'. (If you are creating a new definition you will also need to enter `.htm, .html` for the associated extension, and select a Content Type of `text/html`.)
7 Under the Actions list, click the 'New' button.
8 Type 'Open with Webber' as the action name. (If you want Internet Explorer to be able to open files for editing in Webber, you should also set one up with 'edit' as the action - the rest of the steps would be the same.)
9 Use the 'Browse' button to locate webber.exe. to enter the complete path. (i.e. `C:\WINAPPS\WEBBER\Webber.exe`)
10 Append a switch "-startdde" to the path so that it looks like this: `C:\WINAPPS\WEBBER\Webber.exe -startdde`
11 Click the 'Use DDE' button.
12 Type `open %1` as the DDE Message.
13 Type `Webber` as the Application.
14 Leave 'DDE Application Not Running' blank.
15 Type `WebberCommand` as the Topic.
16 Click the 'Ok' button to save the action.
17 If you would like Webber to be the default application associated with HTML files, click on the 'Open with Webber' action and click the 'Set Default' button.
18 Change the icon to the Webber icon if desired.
19 Click 'Close' twice to return to Explorer.
20 If you have set Webber as the default action, you will be able to open HTML files in Webber by double-clicking on them. You can also select a file (or multiple HTML files), right-click and select 'Open with Webber' from the pop-up menu.

Windows NT 3.51 does not provide the option of associating file types with more then one application. Setting up Webber as the DDE server for HTML files in Windows NT 3.51:

1 Open File Manager.

2 Select File|Associate from the menu.

3 Select HTM from the drop-down list of extensions (or add it if it is not there).

4 If you have a browser such as Netscape or Internet Explorer installed on your system, there will probably be a definition for .htm/.html files. Scroll down the list of file types to find the association for HTML files. It might be called 'Hypertext Document', or 'Netscape Hypertext Document'.

5 Select the definition and click 'Change Type'. If you can't find a definition for HTML files, click the 'New Type...' button to create one.

6 Give the type a description if it does not have one, or edit the existing description if desired.

7 Use the 'Browse' button to locate webber.exe. to enter the complete path. (i.e. `C:\WINAPPS\WEBBER\Webber.exe`)

8 Append a switch "-startdde" to the path so that it looks like this: `C:\WINAPPS\WEBBER\Webber.exe -startdde`

9 Click the 'Use DDE' button.

10 Type `open %1` as the DDE Message.

11 Leave 'DDE Application Not Running' blank.

12 Type `Webber` as the Application.

13 Type `WebberCommand` as the Topic.

14 Enter the extensions to associate below. (i..e. htm, html)

15 Click the 'Ok' button to save the type.

16 Click 'Ok" to return to File Manager.

17 You will be able to open HTML files with Webber by double-clicking on them in File Manager.

Welcome to Webber™

Thank you for registering Webber™ - HTML Editor Extraordinaire! Webber has been designed to be a powerful yet fast and flexible tool for creating HTML documents for the World Wide Web. Webber's combination of features will suit both beginners and experts alike. From our validation feature and syntax highlighting to our tag help and tag assistants, and spell checking, all the features you need to create compliant, great looking documents are here.

What Your Registration Gets You!

You get to feel real good about yourself and...

- Your registration is making it possible for us to continue to upgrade and support Webber™. Even though Webber is an *HTML editor extraordinaire*, there are probably still features that you would like to see added! The money we make from registrations are put directly back into product development, so by registering you are helping to make Webber better for yourself.

- Registering also means that you get product support. We offer support to our customers through e-mail - on the Internet (at webber@csdcorp.com) and Compuserve (at 71554,1271), and by fax. (Sorry - we don't yet have the resources to offer phone support.)

- All those who register are entitled to free upgrades by e-mail until V2.0. We have lots of great plans for new versions - don't be left behind! We do not guarantee that we will always release an updated shareware version of Webber™.

In case you have a friend or colleague interested in getting their very own copy...
[**Registration Information**][**Cheque Order Form**][**Credit Order Form**]

Why Should I Validate my Documents?

Validating your document achieves two important goals. It checks that your document is not missing any structural elements such as angle brackets and end tags. It also ensures that your document is compliant with respect to the HTML dtd.

There is no golden rule that says all documents on the World Wide Web must be valid, compliant documents. In fact, if a document displays correctly in the browser it may not seem important for it to be valid. Browsers are usually very forgiving of mistakes in documents (as they should be), while Webber, especially in Recommended validation mode, is very strict.

So why put yourself through all that pain? In short because you will be ensuring that the documents you are creating will always be supported. The problem is that right now browsers are forgiving, so almost all documents will probably display just fine. However, as browser technology advances and incorporates new features, decisions will have to be made about which classes of existing documents will be supported. At the head of the list will be documents that match the HTML specification exactly - not because browser companies love specifications but because it's easy to support predictably structured documents. As soon as you move outside the specifications it becomes hard to predict what the document will look like. It is that variation that could cause large bodies of existing material to become obsolete. Also, if at some time in the future you want to move your material to PDF or some other format you will have a much easier time if it is all consistently and properly tagged.

So as your father used to say, "it builds character, so stop whining and do it!".

Please note: *The Netscape Extensions and HTML 3.0 tags supported by Webber™ have not yet been adopted by the standards body. Documents created using these features will be supported by many new browsers but are not officially compliant, and may never be. It is our feeling that most, if not all, of these features will eventually be adopted but there is no guarantee. Also many of these features have been accepted and are being widely used.*

Window Menu

Window|New Edit Window
Opens a new tabbed edit window. Select from the menu for a new blank edit window, a new HTML edit window, or to open an existing file in to the window.
Keyboard Equivalent: Alt + W, N

Window|Tile Horizontally
Arranges the open windows one above the other so that the maximum area of each is visible. This option only works if you have more than one edit window open.
Keyboard Equivalent: Alt + W, H

Window|Tile Vertically
Arranges the open windows beside each other so that the maximum area of each is visible. This option only works if you have more than one edit window open.
Keyboard Equivalent: Alt + W, V

Window|Cascade
Arranges the open windows so that the title bar of each is visible. This option only works if you have more than one edit window open.
Keyboard Equivalent: Alt + W, C

Window|Arrange Icons
Arranges the minimized windows so that their icons are lined up at the bottom of the application window. This option only works if you have more than one edit window open.
Keyboard Equivalent: Alt + W, A

Window|Close All
Closes all open windows.
Keyboard Equivalent: Alt + W, L

Window|Open Windows
Shows a list of all open windows, using the active file on the window for its title. Clicking on the title of an open document will bring its window to the top.
Keyboard Equivalent: Alt + W, [file #]

Word Break

compatibility

<wbr></wbr>

Usage:

The WBR element stands for word break. This tag does not force a line break as does <BR>, it only suggests where a line break is allowed to be inserted if needed.

There is no equivalent for this tag as of yet in HTML 3.0.

Attributes:

Required

Optional

Example:

```
<HTML><HEAD>
<TITLE>Information about HTML</TITLE>
</HEAD>
<BODY>
<H1><NOBR>Other information about HTML.</NOBR></H1>
<P>There is lots of information to be had about HTML on the
Internet.</P>
</BODY>
</HTML>
```

Typical Rendering (approximate):

(sorry - can't represent no breaks here!)

Other Information about HTML.

There is lots of information to be had about **HTML** on the Internet.

Word Wrap

Word wrap is available as a preference. When word wrap is enabled the text will automatically wrap to the size of its window. If word wrap is disabled the text will not wrap and carriage returns will be required to move text to the next line. On some computers resizing windows may be slow with word wrap enabled.

Please note: *When HTML documents are displayed by a browser they will automatically be wrapped. Hard returns are ignored. The word wrap feature affects only the editor window, not the display of the documents.*

align (for Tables)
LEFT - Flush left with the left text margin.
CENTER - The table is centered between the text margins and text flow around the table is disabled. This is the default setting for ALIGN.
RIGHT - Flush right with the right text margin.

align
BLEEDLEFT - Flush left with the left (window) border.
LEFT - Flush left with the left text margin.
CENTER - The figure is centered between the text margins and text flow around the figure is disabled. This is the default setting for ALIGN.
RIGHT - Flush right with the right text margin.
BLEEDRIGHT - Flush right with the right (window) border
JUSTIFY - When applicable the figure should be magnified or reduced to fill the space between the left and right text margins. Text flow around the figure is disabled for align=justify.

Nalign

ALIGN=left - image will float down and over to the left margin (into the next available space there), and subsequent text will wrap around the right hand side of that image.
ALIGN=right the - image aligns with the right margin, and the text wraps around the left.
ALIGN=top - image aligns itself with the top of the tallest item in the line.
ALIGN=texttop - image aligns itself with the top of the tallest text in the line (this is usually but not always the same as ALIGN=top).
ALIGN=middle - middle of the image aligns with the baseline of the current line.
ALIGN=absmiddle - middle of the image aligns the middle of the current line.
ALIGN=baseline - bottom of the image aligns with the baseline of the current line.
ALIGN=bottom - identical to ALIGN=baseline but baseline is a better name.
ALIGN=absbottom - aligns the bottom of the image with the bottom of the current line.

align (insert object element)
The following values are chosen for their ease of implementation, and their independence of other graphics occurring earlier on the same line:
ALIGN=TEXTTOP, the top of the object is vertically aligned with the top of the current font.
ALIGN=MIDDLE, the middle of the object is vertically aligned with the baseline.
ALIGN=TEXTMIDDLE, the middle of the object is vertically aligned with the position midway between the baseline and the x-height for the current font. The x-height is defined as the top of a lower case x in western writing systems. If the text font is an all-caps style then use the height of a capital X. For other writing systems, align the middle of the object with the middle of the text.
ALIGN=BASELINE, the bottom of the object is vertically aligned with the baseline of the text line in which the object appears.
ALIGN=TEXTBOTTOM, the bottom of the object is vertically aligned with the bottom of the current font.
Note the proposed Netscape extensions for the align attribute of the IMG element are context sensitive, as are some of the implementations of ALIGN=TOP. See the test page at:
http://www.w3.org/pub/WWW/MarkUp/Group/imgtest.html
The following alignment values allow the object to float rather than being treated as part of the current line:
ALIGN=LEFT, the object is floated down and over to the current left margin. Subsequent text is flowed past the right hand side of the visible area of the object.
ALIGN=CENTER, the object is floated to after the end of the current line and centered between the left and right margins. Subsequent text starts at the beginning of the next line.
ALIGN=RIGHT, the object is floated down and over to the current right margin. Subsequent text is flowed past the left hand side of the visible area of the object.

align
align=left - rendered flush left (the default).
align=center - centered. (default for table headers)
align=right - rendered flush right.
align=justify - Text lines are justified where practical, otherwise this gives the same effect as the default align=left setting.

For Table Col, Table Rows, Table Headers and Table Data
align=char - Text lines are indented such that the first occurrence of a character such as a decimal point on each line are aligned vertically. The character is specified by the CHAR attribute.

tab align
align=left - Following text starts immediately after the designated tab stop (the default).

align=center - Following text up to next tab or line break is centered on the designated tab stop. If the TO attribute is missing, it centers the text between the current left and right margins.

align=right - Following text up to the next tab or line break is rendered flush right to the designated tab stop. If the TO attribute is missing, it renders the text flush right against the current right margin.

align=decimal - The following text is searched for the first occurrence of the character representing the decimal point. The text up to the next tab or line break is then aligned such that the decimal point starts at the designated tab stop. If the TO attribute is missing, the tab element is treated as a single space character.

attribute
An attribute is a named, typed value associated with a tag.

BACKGROUND
This attribute can be used with the BODY tag to specify an image tile to cover the document background. Browsers may ignore this attribute. It is included here for the benefit of browsers that do not yet support style sheets. Note that the text color may need to be adjusted to show an adequate contrast with the background.

browser
Used in this help file to refer to software used for viewing HTML documents on the World Wide Web.

TYPE=CHECKBOX
Form input field that appears as a check box and can be either checked or not. Only buttons that have been checked will have their name/value pair sent with a submitted form.

J This input type has javascript events associated with it. For details see the description of the Javascript event handlers.

CLASS
Used to create more specific types (or classes) of an existing generic tag. The class naming convention is a list of SGML NAME tokens where each class consists of a hierarchical set of names separated by periods. For example, <P CLASS="LESSON.QUESTION"> would mean a paragraph that represents a question within a lesson. It is envisioned that this attribute will be used to enable searching documents for elements that fall into specific classes.

classid
This could be a DCE universally unique object identifier (uuid), or another type of class name as appropriate to the object system, e.g. Java or Corba. This allows effective use of caching, as the user agent can use simple string comparison to check whether two objects are the same independent of their location.

The CLASSID attribute value takes the form of a URL scheme prefix separated by a colon from the character string defining the class identifier. The prefix is used to identify the object system for the class identifier, for example classid="uuid:{663C8FEF-1EF9-11CF-A3DB-080036F12502}" gives the uuid for a Microsoft COM object, using the UUID name space, while classid="java:Animator.class" gives the class name for Java applet.

CLASSID may be sufficient for the user agent to locate the code implementing the object. However, the CODE attribute can also be used with CLASSID to provide a hint as to where to look for this code. The search mechanism will in general depend on the object system the identifier belongs to. Note that the value specified with CLASSID takes precedence over a class identifier derived from the object's data stream.

When searching for the implementation of an object, the CLASSID attribute takes precedence over the CODE attribute which in turn takes precedence over the DATA attribute. A decision tree giving further details on this resolution procedure appears later on in this specification. In the absence of CLASSID a value for the class identifier may be derivable from the CODE or DATA attributes, for instance the Internet media type for the DATA may sufficient, e.g. when the data is for a GIF encoded image.

CLEAR

This attribute is common to all tags that contain blocks of text. When text flows around a figure or table in the margin, you sometimes want to start below the figure rather than alongside it. The CLEAR attribute allows you to move down unconditionally:

clear=left - move down until left margin is clear
clear=right - move down until right margin is clear
clear=all - move down until both margins are clear

Alternatively, you can decide to place the element alongside the figure just so long as there is enough room. The minimum width needed is specified as:

clear="40 en" - move down until there is at least 40 en units free
clear="100 pixels" - move down until there is at least 100 pixels free

The style sheet (or browser defaults) may provide default minimum widths for each class of block-like elements.

clickable
Text and images (including imagemaps) that are clickable will follow a hyperlink when clicked on with a mouse. Browsers will usually change the colour of clickable text and sometimes underline it to indicate that it is clickable. Clickable images will often be outlined with a box of the same colour as the clickable text.

client-side image map
The USEMAP attribute can be used with the ISMAP attribute to indicate that the the image can be processed as either a client-side or server-side image map. The argument to USEMAP specifies which map to use with the image, in a format similar to the HREF attribute on anchors. If the argument to USEMAP starts with a '#', it is assumed to be in the same document as the IMG tag.

This is what Netscape Communications Corp has to say about this addition (taken from their web pages):
Image maps are an important feature of the point-and-click interface which makes the World Wide Web so popular. The most common use of image maps is to allow users to access different documents by clicking on different areas in an image.

There are several limitations of the current image map implementation as it applies to this use. First, it only works over the HTTP protocol, making it unusable for reading local files or files accessed via alternate protocols. Second, a server transaction is required merely to determine where the link is directed. This can degrade performance noticeably when accessing distant sites. Third, unlike for normal links, there is no way for a browser to provide visual feedback to the user showing where a portion of an image map leads before the user actually clicks on it. Lastly, the implementation of image maps is server-dependent, compromising portability of documents.

While HTML+[3] contains provisions for "hypertext buttons" on images via use of the FIG element, this method is an unworkable short-term solution for several reasons. First, complete support of the FIG element requires significant additional processing by the browser. Second, it cannot degrade gracefully on browsers that do not support it. Third, it requires the map description to be specified when the image appears, which is inappropriate for some applications. The extension to support client-side image maps addresses these issues.

The syntax of the client-side image maps provides maximum flexibility to the document author for dealing with browsers which do not support this extension, since such browsers will ignore the MAP and AREA elements. If the document resides on an HTTP server, the server can still provide ISMAP-style support. Otherwise, the author can choose to have the image not appear as an anchor at all, or have a click anywhere within it lead to a another page, perhaps providing an equivalent textual list of options.

The ability to have the map description reside in an different file provides additional flexibility. A common use of image maps is a button bar which appears at the bottom of every document. The map description could be specified in one file, such as the server's home page, and referenced from each document. Thus, the map could be modified by changing a single map description rather than having to modify every file on the server. There is also the possibility of advanced applications with servers dynamically generating map descriptions, similar to the way that some servers currently dynamically generate image files.

The demand for a non-HTTP based mechanism for image maps will also increase as archives of material in HTML format begins to appear on CD-ROM. The expected increase in pay-per-access servers will also lead to users saving copies of documents locally, which they would then expect to function identically to those on the original server. The extensions described here could serve as a basis to satisfy these needs.

comment

Comments can be stored in the source of a document but are not displayed to a user by browsers. A comment is noted by the character sequence `<!-- comment text -->`. The stamp feature uses comments to identify and refresh date and time updates. These stamp comments look like this `<!--Webber_Auto_StampDNR--><!--DS-->23.04.95<!----><!--TS-->4:29:10 PM<!---->`. It is very important not to remove these comments or Webber will not be able to refresh this information properly.

HTML 2	HTML 3	HTML 3.2	Explorer	Netscape
Y	Y			

more...

HTML 2	HTML 3	HTML 3.2	Explorer	Netscape
	Y		Y	

more...

HTML 2	HTML 3	HTML 3.2	Explorer	Netscape
	Y	Y	Y	

more...

HTML 2	HTML 3	HTML 3.2	Explorer	Netscape
	Y			

more...

HTML 2	HTML 3	HTML 3.2	Explorer	Netscape
	Y	Y		

more...

HTML 2	HTML 3	HTML 3.2	Explorer	Netscape
Y	Y	Y	Y	Y

more...

HTML 2	HTML 3	HTML 3.2	Explorer	Netscape
			Y	Y

more...

HTML 2	HTML 3	HTML 3.2	Explorer	Netscape
			Y	

more...

HTML 2	HTML 3	HTML 3.2	Explorer	Netscape
				Y

more...

HTML 2	HTML 3	HTML 3.2	Explorer	Netscape
	Y	Y	Y	Y

more...

HTML 2	HTML 3	HTML 3.2	Explorer	Netscape
Y		Y	Y	Y

more...

HTML 2	HTML 3	HTML 3.2	Explorer	Netscape
		Y	Y	Y

more...

compliance
A document is compliant if it meets all the rules as set out in the HTML specification (dtd). You can make sure your document is compliant by using our validation feature. Only documents checked against the HTML 2.0 dtd will be truly compliant as HTML 3.0 and the Netscape Extensions have not yet been approved.

data
This could be a GIF file or the pickled data representing an object's state. In many cases the media type or the data itself contains sufficient information to identify what code is needed to initialize the object. Note that an object's data can even be included inline for super efficient loading. This specification proposes a new URL scheme "data:". The rest of the URL is a base64 encoded character string that specifies the object's data as an opaque byte stream.

On its own, this would be meaningless. If the DATA attribute appears without a CODE or CLASSID attribute, then a TYPE attribute may be sufficient to interpret the data. For instance a Microsoft COM object can be asked to write its state using the WriteClassStream procedure. This inserts the object's class id as the first 16 bytes of the stream. If the TYPE attribute indicates that the data is in the COM persistent stream format, then the class id can be retrieved from the DATA attribute and used to find the code implementing the object's behaviour.

The CLASSID or CODE attributes can be used to override the default implementation as implied by the DATA attribute. For example, you may have the pickled data for an Excel spread sheet but want to view it with the "SuperGraph" package. You would then use the DATA attribute to point to the Excel spreadsheet data, and the CLASSID or CODE attribute to point to the SuperGraph plug-in.

The CLASSID, CODE and DATA attributes specify URLs. Any fragment identifier included as part of these URLs should be passed to the object, either directly, or by callback.

Deprecated
Uses the deprecated aspects of the HTML dtd when validating a document. Deprecated features of the dtd will become obsolete and unsupported in future versions of the dtd. If you use deprecated then your document will validate even if you have use deprecated features. This setting is meant to provide backwards compatibility with old documents. New documents should probably be created with the recommended setting.

Examples of deprecated features are certain tags (PLAINTEXT, XMP, LISTING - all deprecated in HTML 2 and obsolete in HTML 3), and placement of text/tags within a document (both dtd's have deprecated the placement of text inside the body of the document - it must be located inside a para, list, heading etc.).

dingbat
Specifies an iconic image to appear preceding the header or list item. The icon is specified as an entity name. Refer to the list of standard icon entity names for HTML 3.0. Implementation of this feature by browsers has not yet been determined.

Standard ISO/WWW icons courtesy of Bert Bos and Kevin Hughes (from draft HTML 3.0 dtd).

These can be used in place of default symbols for list items or as part of hypertext links, and save time needed to download images. Browsers can define them in terms of library images or as URL/URNs. Implementation of this feature by browsers has not yet been determined.

```
<!ENTITY ftp SDATA "ftp" -- ftp server -->
<!ENTITY gopher SDATA "gopher" -- gopher server -->
<!ENTITY telnet SDATA "telnet" -- telnet connection -->
<!ENTITY archive SDATA "archive" -- archive server -->
<!ENTITY filing.cabinet SDATA "filing.cabinet" -- filing cabinet -->
<!ENTITY folder SDATA "folder" -- folder or directory -->
<!ENTITY fixed.disk SDATA "fixed.disk" -- fixed media drive -->
<!ENTITY disk.drive SDATA "disk.drive" -- removeable media drive -->
<!ENTITY document SDATA "document" -- unspecified document type -->
<!ENTITY unknown.document SDATA "unknown.document" -- unrecognised document type
-->
<!ENTITY text.document SDATA "text.document" -- text/plain, text.html etc. -->
<!ENTITY binary.document SDATA "binary.document" -- binary data -->
<!ENTITY binhex.document SDATA "binhex.document" -- binhex format -->
<!ENTITY audio SDATA "audio" -- audio sequence -->
<!ENTITY film SDATA "film" -- film or animation, such as an MPEG movie --> <!ENTITY image SDATA "image" --
photograph, drawing or graphic of any kind -->
<!ENTITY map SDATA "map" -- geographical or a schematic map -->
<!ENTITY form SDATA "form" -- fill-out form -->
<!ENTITY mail SDATA "mail" -- email messages -->
<!ENTITY parent SDATA "parent" -- parent of current document -->
<!ENTITY next SDATA "next" -- next document in current sequence -->
<!ENTITY previous SDATA "previous" -- previous document in current sequence -->
<!ENTITY home SDATA "home" -- home document -->
<!ENTITY toc SDATA "toc" -- table of contents -->
<!ENTITY glossary SDATA "glossary" -- glossary of terms etc. -->
<!ENTITY index SDATA "index" -- searchable index -->
<!ENTITY summary SDATA "summary" -- summary -->
<!ENTITY calculator SDATA "calculator" -- A calculator -->
<!ENTITY caution SDATA "caution" -- Warnign sign -->
<!ENTITY clock SDATA "clock" -- A clock -->
<!ENTITY compressed.document SDATA "compressed.document">
<!ENTITY diskette SDATA "diskette" -- A diskette -->
<!ENTITY display SDATA "display" -- A computer screen -->
<!ENTITY fax SDATA "fax" -- A fax machine -->
<!ENTITY mail.in SDATA "mail.in" -- mail-in tray -->
<!ENTITY mail.out SDATA "mail.out" -- mail-out tray -->
<!ENTITY mouse SDATA "mouse" -- mouse/pointing device -->
<!ENTITY printer SDATA "printer" -- hardcopy device -->
<!ENTITY tn3270 SDATA "tn3270" --tn3270 terminal session -->
<!ENTITY trash SDATA "trash" -- waste paper basket -->
<!ENTITY uuencoded.document SDATA "uuencoded.document" -- uuencoded data -->
```

dtd
Document Type Definition - a set of SGML definitions (rules) that define the structure of a document of a particular type. HTML documents use a dtd (i.e. HTML 2.0 or HTML 3.0) which defines the structure that all HTML documents must follow.

dtd Quick Reference

HTML 2.0

Tag Name / *Attributes*

<a>
Required
Optional
href
name
rel
rev
urn
title
methods

<address></address>
Required
Optional

HTML 3.0

Tag Name / *Attributes*

<a>
Required
Optional
id -ID
lang
class
href
md
name
shape
rel
rev
title
methods

<abbrev></abbrev>
Required
Optional
id -ID
lang
class

<above></above>
Required
Optional
sym

<acronym></acronym>
Required
Optional
id -ID
lang
class

<address></address>
Required
Optional
id -ID
lang
class
clear

<array></array>
Required
Optional
align [TOP | MIDDLE
| BOTTOM]
coldef
ldelim
rdelim
labels [LABELS]

<atop>
Required
Optional

<au></au>
Required
Optional
id -ID
lang
class

****
Required
Optional

****
Required
Optional
id -ID
lang
class

<banner></banner>
Required
Optional
id -ID
lang
class

<bar></bar>
Required
Optional

<base>
Required
href
Optional

<base>
Required
href
Optional
id -ID

<below></below>
Required
Optional
sym

<big></big>
Required
Optional
id -ID
lang
class

<blockquote></blockquote>
Required
Optional

<blockquote></blockquote>
Required
Optional
id -ID
lang
class
clear
nowrap [NOWRAP]

<body></body>
Required
Optional

<body></body>
Required
Optional
id -ID
lang
class
background

<box></box>

Required
Optional
size [NORMAL | MEDIUM
| LARGE | HUGE]

<bq></bq>
Required
Optional
id -ID
lang
class
clear
nowrap [NOWRAP]

**
**
Required
Optional

**
**
Required
Optional
id -ID
lang
class
clear

<bt></bt>
Required
Optional
class

<caption></caption>
Required
Optional
id -ID
lang
class
align [TOP | BOTTOM
| LEFT | RIGHT]

<choose>
Required
Optional

<cite></cite>
Required
Optional

<cite></cite>
Required
Optional
id -ID
lang
class

<code></code>
Required
Optional

<code></code>
Required
Optional
id -ID
lang
class

<col>
Required
Optional
id -ID
lang
class
style
dir

span
width
align [LEFT | CENTER
| RIGHT | JUSTIFY | CHAR]
valign [TOP | MIDDLE
| BOTTOM | BASELINE]
char
charoff

<colgroup></colgroup>
Required
Optional
id -ID
lang
class
style
dir
span
width
align [LEFT | CENTER
| RIGHT | JUSTIFY | CHAR]
valign [TOP | MIDDLE
| BOTTOM | BASELINE]
char
charoff

<credit></credit>
Required
Optional
id -ID
lang
class

<dd></dd>
Required
Optional

<dd></dd>
Required
Optional
id -ID
lang
class
clear

<ddot></ddot>
Required
Optional

<del></del>
Required
Optional
id -ID
lang
class

<dfn></dfn>
Required
Optional
id -ID
lang
class

<dir></dir>
Required
Optional

<dir></dir>
Required
Optional

compact [COMPACT]

compact [COMPACT]

** **

Required

Optional

id -ID

lang

class

clear

align [LEFT | CENTER | RIGHT]

nowrap [NOWRAP]

<dl></dl>

Required

Optional

compact [COMPACT]

<dl></dl>

Required

Optional

id -ID

lang

class

clear

compact [COMPACT]

<dot></dot>

Required

Optional

<dt></dt>

Required

Optional

<dt></dt>

Required

Optional

id -ID

lang

class

clear

****

Required

Optional

****

Required

Optional

id -ID

lang

class

<fig></fig>

Required

src

Optional

id -ID

lang

class

clear

md

align [BLEEDLEFT | LEFT | CENTER | RIGHT | BLEEDRIGHT | JUSTIFY]

noflow [NOFLOW]

width

height

units [EN | PIXELS]

imagemap

<fn></fn>

Required

Optional
id -ID
lang
class

<form></form>
Required
Optional
action
method [GET | POST]
enctype

<form></form>
Required
action
Optional
method [GET | POST]
enctype
script

<h1></h1>
Required
Optional

<h1></h1>
Required
Optional
id -ID
lang
class
align [LEFT | CENTER
| RIGHT | JUSTIFY]
clear
seqnum
skip
dingbat
src
md
nowrap [NOWRAP]

<h2></h2>
Required
Optional

<h2></h2>
Required
Optional
id -ID
lang
class
align [LEFT | CENTER
| RIGHT | JUSTIFY]
clear
seqnum
skip
dingbat
src
md
nowrap [NOWRAP]

<h3></h3>
Required
Optional

<h3></h3>
Required
Optional
id -ID
lang
class
align [LEFT | CENTER
| RIGHT | JUSTIFY]
clear
seqnum
skip
dingbat
src
md
nowrap [NOWRAP]

<h4></h4>

<h4></h4>

Required
Optional

Required
Optional
id -ID
lang
class
align [LEFT | CENTER
| RIGHT | JUSTIFY]
clear
seqnum
skip
dingbat
src
md
nowrap [NOWRAP]

<h5></h5>
Required
Optional

<h5></h5>
Required
Optional
id -ID
lang
class
align [LEFT | CENTER
| RIGHT | JUSTIFY]
clear
seqnum
skip
dingbat
src
md
nowrap [NOWRAP]

<h6></h6>
Required
Optional

<h6></h6>
Required
Optional
id -ID
lang
class
align [LEFT | CENTER
| RIGHT | JUSTIFY]
clear
seqnum
skip
dingbat
src
md
nowrap [NOWRAP]

<hat></hat>
Required
Optional

<head></head>
Required
Optional

<head></head>
Required
Optional

<hr>
Required
Optional

<hr>
Required
Optional
id -ID
lang
class
src

	md clear
<html></html> *Required* *Optional* version	**<html></html>** *Required* *Optional* version urn role
**** *Required* *Optional*	**** *Required* *Optional* id -ID lang class
<img> *Required* src *Optional* alt align [TOP | MIDDLE | BOTTOM] ismap [ISMAP]	**<img>** *Required* src *Optional* id -ID lang class md alt align [TOP | MIDDLE | BOTTOM | LEFT | RIGHT] width height units [EN | PIXELS] ismap [ISMAP]
<input> *Required* *Optional* type [TEXT | PASSWORD | CHECKBOX | RADIO | SUBMIT | RESET | IMAGE | HIDDEN] name value src checked [CHECKED] size maxlength align [TOP | MIDDLE | BOTTOM]	**<input>** *Required* *Optional* type [TEXT | PASSWORD | CHECKBOX | RADIO | SUBMIT | RESET | RANGE | FILE | SCRIBBLE | HIDDEN | IMAGE] name id -ID lang class value src md checked [CHECKED] size maxlength align [TOP | MIDDLE | BOTTOM | LEFT | RIGHT] disabled accept error max min

<ins></ins>
Required
Optional
id -ID
lang
class

<isindex>
Required
Optional

<isindex>
Required
Optional
href
prompt

<item></item>
Required
Optional
align
colspan
rowspan

<kbd></kbd>
Required
Optional

<kbd></kbd>
Required
Optional
id -ID
lang
class

<lang></lang>
Required
Optional
id -ID
lang
class

<left>
Required
Optional

<lh></lh>
Required
Optional
id -ID
lang
class

**
**
Required
Optional

**
**
Required
Optional
id -ID
lang
class
clear
dingbat
src
md
skip

<link>
Required
href
Optional
rel

<link>
Required
href
Optional
rel

rev
urn
title
methods

rev
title
methods

$$
Required
Optional
id -ID
class

<listing></listing>
Required
Optional

<menu></menu>
Required
Optional
compact [COMPACT]

<menu></menu>
Required
Optional
compact [COMPACT]

<meta>
Required
content
Optional
http-equiv
name

<meta>
Required
content
Optional
http-equiv
name

<nextid>
Required
n
Optional

<nextid>
Required
n
Optional

<note></note>
Required
Optional
id -ID
lang
class
src
md
clear

<of></of>
Required
Optional

**
**
Required
Optional
compact [COMPACT]

**
**
Required
Optional
id -ID
lang
class
clear
continue [CONTINUE]
seqnum
compact [COMPACT]

<option></option>
Required
Optional
selected [SELECTED]
value

<option></option>
Required
Optional
id -ID
lang

class
selected [SELECTED]
value
shape
disabled
error

<over>
Required
Optional

<overlay>
Required
src
Optional
md
units [EN | PIXELS]
x
y
width
height
imagemap

** **
Required
Optional

** **
Required
Optional
id -ID
lang
class
align [LEFT | CENTER
| RIGHT | JUSTIFY]
clear
nowrap [NOWRAP]

<person></person>
Required
Optional
id -ID
lang
class

<plaintext></plaintext>
Required
Optional

<pre></pre>
Required
Optional
width

<pre></pre>
Required
Optional
id -ID
lang
class
width
clear

<q></q>
Required
Optional
id -ID
lang
class

<range>

Required
from -IDRef
until -IDRef
Optional
id -ID
class

<right>
Required
Optional

<root></root>
Required
Optional

<row></row>
Required
Optional

<s></s>
Required
Optional
id -ID
lang
class

<samp></samp>
Required
Optional

<samp></samp>
Required
Optional
id -ID
lang
class

<select></select>
Required
name
Optional
size
multiple [MULTIPLE]

<select></select>
Required
name
Optional
id -ID
lang
class
multiple [MULTIPLE]
error
disabled [DISABLED]
src
md
width
height
units [EN | PIXELS]
align [TOP | MIDDLE
| BOTTOM | LEFT
| RIGHT]

<small></small>
Required
Optional
id -ID
lang
class

<spot>
Required
id -ID

Optional

<sqrt></sqrt>
Required
Optional

****
Required
Optional

****
Required
Optional
id -ID
lang
class

<style></style>
Required
notation
Optional

****
Required
Optional
id -ID
lang
class
align [LEFT | CENTER
| RIGHT]

****
Required
Optional
id -ID
lang
class
align [LEFT | CENTER
| RIGHT]

<t></t>
Required
Optional
class

<tab>
Required
Optional
id -ID
indent
to -IDRef
align [LEFT | CENTER
| RIGHT | DECIMAL]
dp

<table></table>
Required
Optional
id -ID
lang
class
style
dir
border [BORDER]
cols
width

align [LEFT
| CENTER | RIGHT]
cellpadding
cellspacing

<tbody></tbody>

Required
Optional
id -ID
lang
class
style
dir
align [LEFT | CENTER
| RIGHT | JUSTIFY | CHAR]
valign [TOP | MIDDLE
| BOTTOM | BASELINE]
char
charoff

<td></td>

Required
Optional
id -ID
lang
class
style
dir
colspan
rowspan
nowrap
align [LEFT | CENTER
| RIGHT | JUSTIFY | CHAR]
valign [TOP | MIDDLE
| BOTTOM | BASELINE]
char
charoff
axis
axes

<textarea></textarea>

Required
name
rows
cols
Optional

<textarea></textarea>

Required
name
rows
cols
Optional
id -ID
lang
class
align [TOP | MIDDLE
| BOTTOM | LEFT
| RIGHT]

<tfoot></tfoot>

Required
Optional
id -ID
lang
class
style
dir
align [LEFT | CENTER

| RIGHT | JUSTIFY | CHAR]
valign [TOP | MIDDLE
| BOTTOM | BASELINE]
char
charoff

<th></th>

Required
Optional
id -ID
lang
class
style
dir
colspan
rowspan
nowrap
align [LEFT | CENTER
| RIGHT | JUSTIFY | CHAR]
valign [TOP | MIDDLE
| BOTTOM | BASELINE]
char
charoff
axis
axes

<thead></thead>

Required
Optional
id -ID
lang
class
style
dir
align [LEFT | CENTER
| RIGHT | JUSTIFY | CHAR]
valign [TOP | MIDDLE
| BOTTOM | BASELINE]
char
charoff

<tilde></tilde>

Required
Optional

<title></title>

Required
Optional

<title></title>

Required
Optional

<tr></tr>

Required
Optional
id -ID
lang
class
style
dir
align [LEFT | CENTER
| RIGHT | JUSTIFY | CHAR]
valign [TOP | MIDDLE
| BOTTOM | BASELINE]
char

charoff

<tt></tt>
Required
Optional

<tt></tt>
Required
Optional
id -ID
lang
class

<u></u>
Required
Optional
id -ID
lang
class

**
**
Required
Optional
compact [COMPACT]

**
**
Required
Optional
id -ID
lang
class
clear
wrap [VERT | HORIZ | NONE]
plain [PLAIN]
dingbat
src
md
compact [COMPACT]

<var></var>
Required
Optional

<var></var>
Required
Optional
id -ID
lang
class

<vec></vec>
Required
Optional

<xmp></xmp>
Required
Optional

element
An element is a generic term referring to part of a document, such as its title, its body, a paragraph, a list, etc. It can also refer to tags, for example, isindex may be referred to as an element.

empty tag
Empty tags are those that do not contain any content, such as line breaks <BR>, and horizontal rules <HR>. Empty tags have only start tags and no end tag.

TYPE=FILE
This input type allows users to attach one or more files to be submitted with the form's contents. The ACCEPT attribute can be used to specify a comma separated list of MIME content types. These are used to restrict the kinds of files that can be attached to the form.

frame
VOID - Don't render any sides of the frame.
ABOVE - The top side of the frame
BELOW - The bottom side of the frame
HSIDES - The top and bottom sides of the frame
LHS - The left hand side of the frame
RHS - The right hand side of the frame
VSIDES - The left and right sides of the frame
BOX - All four sides of the frame
BORDER - All four sides of the frame

TYPE=HIDDEN
This attribute makes the input field a hidden field. The user does not interact with this field; instead, the VALUE attribute can be used to specify a value which is sent with the submitted form. This value may be used to transmit state information about client/server interaction, for instance a transaction identifier.

hyperlink
Where text or a graphic in one document is 'linked' to another document or other file such as a sound, graphic, or binary file. Clicking on the link loads the object that is linked to. Most browsers allow you to set preferences for handling the links. Hyperlinks pointing to HTML documents and some types of graphics files will be loaded by the browser. For other types of files, external viewers can be launched or the file can be saved to the local hard disk.

TYPE=IMAGE

The image input specifies an image resource to display. This allows you to mimic the behaviour of a SUBMIT button using an image and to retrieve the x and y co-ordinates of the click as form data. The NAME attribute is returned twice, as *name.x* and *name.y* with the value being the x & y co-ordinates chosen.

LANG
Used in HTML 3.0, this attribute can be used by parsers to select language specific choices for quotation marks, ligatures and hyphenation rules etc. The value is one of the ISO standard language abbreviations, i.e. "en.uk" for the variation of English spoken in the United Kingdom. The language attribute is composed from the two letter language code from ISO 639, optionally followed by a period and a two letter country code from ISO 3166.

md
Specifies a message digest or cryptographic checksum for the associated graphic specified by the SRC attribute. It is used when you want to be sure that a linked object is indeed the same one that the author intended, and hasn't been modified in any way. The MD attribute is generally allowed for all elements which support URI based links.

named
As in, it has a name. For example: HREF is the name of an attribute associated with the anchor tag.

nested
Nesting is when a tag is placed between the start tag and end tag of another tag. This is a useful technique for multi-level lists, word level emphasis, etc.

Example:

```
<UL>
<LI>Section A
<UL><LI>Chapter 1
</LI><LI>Chapter 2
</LI></UL></LI>
<LI>Section B
<UL><LI>Chapter 1
</LI><LI>Chapter 2
</LI></UL>
</UL>
```

Typical Rendering (approximate):

Index

- Section A
  - Chapter 1
  - Chapter 2
- Section B
  - Chapter 1
  - Chapter 2

noflow
The presence of this attribute disables text flow around the table or figure. If you use this attribute you will not need to use the CLEAR or NEEDS attributes on the the tag which follows the table or figure.

NOWRAP
The NOWRAP attribute is used to prevent the browser from automatically wrapping lines. You can then specify line breaks within text blocks using the BR tag.

TYPE=PASSWORD
Form input field similar to the TEXT attribute, except that the value is obscured as it is entered.

TYPE=RADIO
Form input field that appears as a radio button. Several radio buttons with the same NAME attribute will work as a unit - turning one on will turn the others in the unit off. Only the value of the selected button is sent when the form is submitted. The VALUE attribute is required for radio inputs.

J This input type has javascript events associated with it. For details see the description of the Javascript event handlers.

TYPE=RANGE

This attribute allows the user to pick a numeric value in between a lower and an upper bound. The range is specified with the MIN and MAX attributes, as in: `<input name=rating type=range min=1 max=10>`

relative
When a destination document is linked without using the full url.
For example:
```
<A HREF="index.html">Index</A>
```
Relative urls are used so that a series of related documents can be easily moved from one location to another without 'breaking' the anchors within them. This method is especially useful in combination with the BASE tag.

TYPE=RESET
Form input that appears as button and clears the information from the form without posting it, and resets all fields to their initial states. The value of the VALUE attribute will appear as the text or caption on the button.

J This input type has javascript events associated with it. For details see the description of the Javascript event handlers.

Right-Click Menu (Structure Assistant)

Right-click menu options for the structure assistant are as follows:

Link to... :	this option is enabled when you select either an anchor or an image tag in the structure. When selected, a dialog will open that lists images linked to recently, as well as those used in the current document. For anchors, the dialog will show the files linked to recently, as well as internal targets available to link to. Select an image or link from the list, and click 'Ok'. The link will be added to the tag in your document.
Follow link :	this option is enabled when you select an anchor in the structure. When selected, it will take you to the correct place in the current document (for internal links), or it will open the file that is linked to (if the file is a local one and the path used is correct on the local system).
Tag Assistant... :	select any tag in the structure and choose this option to open the tag assistant with the selected tag and attributes, ready for editing.
Expand All :	this option will expand all levels of the current document structure.
Collapse All :	This option will collapse all levels of the current document structure.
Show Id or Name :	when enabled, this option will show the id or name associated with all applicable tags in square brackets beside the tag name, i.e. "H1[first target]".
Show Link :	when enabled, this option will show the file that is linked to the anchor or image tag. The link is shown as following a dash and angle bracket, i.e. "A->file.htm" or "IMG->graphic.gif".
Show Content :	when enabled, this option will show the first 20 characters of content for any applicable tag. The content is shown following a dash, i.e. "P - This is the first pa...".

rules
NONE - Suppresses internal rulings.
BASIC - The THEAD, TFOOT and TBODY elements divide the table into groups of rows. This choice places a horizontal rule between each such group.
ROWS - Place horizontal rules between all rows. User agents may choose to use a heavier rule between groups of rows for emphasis.
COLS - Place vertical rules between groups of columns as defined by COLGROUPS and COL elements, plus horizontal rules between row groups (see rules=basic).
ALL - Place rules between all rows and all columns. User agents may choose to use a heavier rule between groups of rows and columns for emphasis.

TYPE=SCRIBBLE

This input field type allows the user to scribble with a pointing device (such as a mouse or pen) on top of a predefined image. The image is specified as a URI with the SRC attribute. If the browser can't display images, or can't provide a means for users to scribble on the image, then the field should be treated as a text field. The VALUE attribute can be used to initialize the text field for these users. It is ignored when the browser provides scribble on image support.

seqnum
A sequence number is associated with each level of header from the top level (H1) to the bottom level (H6). This attribute is used to set the sequence number associated with the header level of the current element to a given number, e.g. SEQNUM=10. Normally, the sequence number is initialized to 1 at the beginning of the document and incremented after each header element. It is reset to 1 by any header element of a higher level, e.g. an H1 header resets the sequence numbers for H2 to H6. The style of header numbering is controlled by the style sheet.

shape

This attribute is used within figures to define shaped hotzones for graphical hypertext links. The attribute value is a string taking one of the following forms:

"default" - Used to define a default link for the figure background.

"circle x, y, r" - Where x and y define the center and r specifies the radius.

"rect x, y, w, h" - Where x, y define the upper left corner and w, h define the width and height respectively

"polygon x1, y1, x2, y2, ..." - Given n pairs of x, y coordinates, the polygon is closed by a line linking the n'th point to the first. Intersecting polygons use the non-zero winding number rule to determine if a point lies inside the polygon.

If a pointer event occurs in a region where two or more shapes overlap, the distance from the point to the center of gravity of each of the overlapping shapes is computed and the closest one chosen. This feature is useful when you want lots of closely spaced hotzones, for example over points on a map, as it allows you to use simple shapes without worrying about overlaps.

Note: The x coordinate increases to the right, and the y coordinate increases downwards in the same way as IMG and image maps. If both numbers are integers, the coordinates are interpreted as pixel offsets from the upper left corner of the figure. Otherwise, the coordinates are interpreted as scaled values in the range 0.0 to 1.0 across the figure. Note the syntax is tolerant of repeated white space characters between tokens.

skip
Increments the sequence number before rendering the element. It is used when headers or list items have been left out of the sequence. For instance, SKIP=3 advances the sequence number past 3 omitted items.

src
Specifies an image as a URI. Depending on the tag it is used with, this attribute can be used to place a custom graphic at the start of the heading or list item to act as a bullet, or to appear as the horizontal rule, etc.

standard units
Several attributes specify widths as a number followed by an optional suffix (i.e. `2cm`) The units for widths are specified by the suffix:
`pt` = points
`pi` = picas
`in` = inches
`cm` = centimeters
`mm` = millimeters
`em` = em units (equal to the height of the default font)
`px` = screen pixels.
The default units are screen pixels (chosen for backwards compatibility). The number is an integer value or a real valued number such as "2.5". Exponents, as in "1.2e2", are not allowed. White space is not allowed between the number and the suffix.

The above set of suffices is augmented for certain elements: "%" is used for the WIDTH attribute for the TABLE element. It indicates that the attribute specifies the percentage width of the space between the current left and right margins, i.e. width="50%". For the COL element, "*" is used with the the WIDTH attribute to specify relative column widths, i.e. width="3*", using the same representation as the CALS table model.

Structure Assistant

The structure assistant displays the structure of the active document in a collapsible outline format. Viewing the structure of your document gives you a quick way to navigate, as well as being a quick way to ensure that elements in the document are correctly layed out. For example, you could quickly tell whether a list item was located inside the list, or that a table cell is inside a table row.

The structure assistant also includes many useful features which are accessible from its right-click menu.

To Use:

1 Select Options|Show structure from the menu. This will open the structure assistant which will display an outline of the current document. The structure assistant can float so you can leave it open while you work. When you make changes to your document, or change to a new document, and click back on the structure assistant it will refresh itself accordingly.

2 Collapse and expand portions of the structure by double-clicking on the tag names. Or you can use the right-click menu to collapse or expand all levels.

3 As you click on tag names in the structure, the associated start tag will be highlighted in your document, allowing you to easily navigate through your document.

4 Other right-click menu options allow you to display the id or name of a tag, any linked files, and/or the first 20 characters of content for a tag. You can also navigate by following links where appropriate, and even opening linked files (if they are local).

TYPE=SUBMIT
Form input that appears as button and submits the information in the form as defined by the action and method attributes. The value of the VALUE attribute will appear as the text or caption on the button. If the NAME attribute is present, the name/value pair will be submitted with the form information. If the NAME attribute is not present, this element does not return any value. The NAME attribute is useful if there are many buttons and it is desirable to know which one was pushed.

J This input type has javascript events associated with it. For details see the description of the Javascript event handlers.

Getting Support

You can get support for Webber in the following ways (in the order of our preference!):

- fill out our customer feedback form at : http://www.csdcorp.com/websupt.htm
- send e-mail to *webber@csdcorp.com* or to 71554,1271 on Compuserve
- send snail-mail to CSD Corp. 499 Rushton Road, Toronto, Ontario, Canada M6C 2Y4
- send a fax to CSD Corp. at +1 416.651.7685

Support is offered to registered users of Webber only, so be sure to give us your name, organization name, and e-mail address for reference.

Things you should do before contacting us:

- try to reproduce the problem and record all details (even those that might not seem important to you).
- refer to our help file, readme.txt file and our release notes. The latest version of our release notes will be at our Web site at : http://www.csdcorp.com/relnotes.htm
- if you are sending e-mail, snail mail or a fax, be sure to give us all the details of your system - machine type, processor speed, RAM, Windows/DOS versions, browser used, Webber version, other software/hardware configuration. Also give us a detailed problem description - the more detailed it is the more likely we are to able to help you promptly. Especially important are details regarding the conditions under which the problem occurs (repeatable?).

sym
Used with some math tags to specify the symbol to be used. If this attribute is not used it defaults to "line". Other choices include: "cub" - for a curly bracket (brace); "larr" - (left arrow); "rarr" - (right arrow); "hat" and "tilde".

tag
Tags are used to mark the start and end of headings, paragraphs, lists, character formatting and of course, hyperlinks. Usually there will be a start tag and an end tag for each element. Tags are enclosed by angle brackets <*tagname*> and end tags also include a forward slash </*tagname*>.

Text formatting tags in the HTML 2.0 dtd include:
BR, EM, STRONG, B, I, CITE, VAR, SAMP, CODE, KBD, TT

TYPE=TEXT
Form input field for single line text entry fields. The SIZE and MAXLENGTH attributes may be used to constrain the input or layout of the field. Use the <TEXTAREA> element for multi-line text fields.

J This input type has javascript events associated with it. For details see the description of the Javascript event handlers.

type

The following grammar for media types is a superset of that for MIME because it does not restrict itself to the official IANA and x-token types.

```
media-type      = type "/" subtype *(";" parameter )
type            = token
subtype         = token
```

where token is defined by:

```
token           =   1*<any (ASCII) CHAR except SPACE, CTLs, or tspecials>
tspecials       = <one of the set>    ( ) < > @ , ; : \ " / [ ] ? =
```

Parameters may follow the type/subtype in the form of attribute/value pairs.

```
parameter       = attribute "=" value
attribute       = token
value           = token | quoted-string
```

The type, subtype, and parameter attribute names are case-insensitive. Parameter values may or may not be case-sensitive, depending on the semantics of the parameter name. White space characters must not be included between the type and subtype, nor between an attribute and its value.

If a given media-type value has been registered by the IANA, any use of that value must be indicative of the registered data format. Although HTML allows the use of non-registered media types, such usage must not conflict with the IANA registry. Data providers are strongly encouraged to register their media types with IANA via the procedures outlined in RFC 1590.

All media-type's registered by IANA must be preferred over extension tokens. However, HTML does not limit applications to the use of officially registered media types, nor does it encourage the use of an "x-" prefix for unofficial types outside of explicitly short experimental use between consenting applications.

typed
Defined as having an SGML defined type which restricts the possible values an attribute can contain. For example: the NAME attribute of the anchor tag is of type ID, which means that it must be unique within the document, begin with a letter and contain no more than a fixed number of alpha-numeric characters.

units

units=en

Specifies en units (a typographical unit equal to half the point size). This is the default setting for tables and allows user agents to render the table a row at a time without waiting until all of the table's data has been received.

units=relative

Used to set the relative width of columns. The user agent sums the values to determine the proportional width of each column.

units=pixels

Specifies pixels as the unit to be used. Pixels are the default for images and figures. The other two choices are more useful for tables.

url
Universal Resource Locator - the address or location of a document.

valign
align=top - The cell contents appear at the top of each cell (the default).
valign=middle - Cell contents are centered vertically in each cell.
valign=bottom - The cell contents appear at the bottom of each cell.
valign=baseline - This is used when you want to ensure that all cells in the row with valign=baseline share the same baseline. This constraint only applies to the first text line for each cell.

Note: In the absence of the VALIGN attribute, the default can be overridden by the presence of a VALIGN attribute on the parent TR element.