



Copyright©, 2008 BrandsPatch LLC

<http://www.explainth.at>

Color key overleaf

Code Structure

```
<!DOCTYPE html PUBLIC DTD>
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    [<base href="url" [id="id" target="target"]>
     <meta name="value" content="text" />
     <script type="text/javascript" [src="url" id="id"
      defer="defer"]>
    </script>
    <style type="text/css" [id="id" media="media"
     title="title"]>
    </style>
    <link rel="value" href="url" type="type"
     target="target" media="media"/>
    <!--comment-->
    <!--#SSI Directive -->
  ]
  <title>Document Title</title>
</head>
<body>
  [
    <!--comment-->
    <!--#SSI Directive -->
    <script type="text/javascript" [src="url" id="id"
     defer="defer"]>
    </script>
  ]
</body>
</html>
```

For valid XHTML documents the xmlns attribute is required. The title tag is required for validation & is advisable for SEO.

Document Type Declaration (DTD)

"-//W3C//DTD a b" "http://www.w3.org/TR/c" where

	HTML	XHTML
a	HTML	XHTML
b	4.01/EN	1.1/EN
c	html4/strict.dtd	xhtml11/DTD/xhtml11.dtd

Other HTML DTDs exist.

<base >

<base href="url" [id="id" target="target"]>
 Specifies the base path name for all relative urls in the document. url is usually absolute. In theory it is possible to respecify the base path relative to a previously specified absolute url – but not with Internet Explorer!

Use target to indicate the target window for clickable links in the document. See the anchor, <a>, tag for a detailed description of the target attribute.

<meta>

```
<meta name="value" content="text" [id="id"] />
<meta http-equiv="value" content="text" />
```

Name	Content
description	Page Description
robots	[options]

Other name values exist but are of limited utility. The robots tag provides one or more instructions to search engine robots.

Option	Meaning
noindex	Don't index this page
nofollow	Don't follow links on this page
nooodp	Don't use DMOZ descriptions
noarchive	Don't archive this page
noimageindex	Index page text only, not images
noimageclick	Image searches link to page, not image.

http-equiv is intended to supply the browser with information that would otherwise appear in server response headers. Not all browsers support this feature. One notable exception

```
<meta http-equiv="refresh" content="n;url="url" />
<script>
```

<script type="text/javascript" [src="url" id="id" defer="defer"]></script>
 Use this to insert JavaScript code directly into the document or specify the src attribute to link to an external script. Executable code – i.e. not wrapped inside a function block that needs to be explicitly called - in scripts in the body part of the document will be executed as soon as it is loaded. In theory, with defer="defer" the script is not interpreted until the document has been loaded.

<style>

```
<style type="text/css" [id="id" media="media"
title="title"]>
media = [all,aural,braille,handheld,print,projection,
screen,ty,tv]
```

<link>

```
<link rel="stylesheet" type="text/css" href="url"
media="media" />
<link rel="shortcut icon" href="url" type="image/x-
icon" />
media as described in <style>.
```

Other values for rel exist but are not well supported.

Comments

```
<!--comment--> where comment is --comment text--.
Comments are delimited by two – characters so it is in
principle possible to write
<!--commentA---commentB---commentC-->
```

SSI Directives

Server Side Include (SSI) directives are written as comments in HTML but never make it to the client – provided that the server has been configured to parse and the document and interpret them. For more details see <http://www.explainth.at/en/gr/ssiqr.shtml>

<body>

<body>
 ...
 </body>
 There can be only one <body>...</body> tag pair in the document. All viewer content goes between these tags. To pass validation use container elements – div, p, pre, h# etc – to hold all content placed inside the document body. Other elements and/or bare character data are not legal but accepted by most browsers.

<a>

```
<a href="url" [id="id" name="name"
target="target"]>Anchor Text</a>
Anchors define clickable links in the document. target
can be a named frame, e.g.
<iframe href="url" name="name"> </iframe>
<a href="url" target="name">Anchor Text</a>
```

or one of the predefined value tabulated below

target	Meaning
_self	Current window
_blank	New window

_parent	Parent frame
_top	Unframed window

Frames should be avoided and forcing a new browser instance/tab on the viewer is impolite. Consequently XHTML bans the use of this attribute.

<button>

```
<button type="type" name="name" value="value"
[id="id" disabled="disabled"]>innerHTML</button>
innerHTML can be plain text or an <img> element.
```

type = submit|reset|button
 The implementation of the default value of type is not consistent across browsers. When type="button", or when used in a form without an action attribute the onclick event handler must be assigned.

<div>

<div>...</div>
 The workhorse of any well designed HTML document. Acts as a container for other HTML elements. Generally used with the optional attributes

```
[id="id" class="class" style="style"]
```

Beware of using div where span, pre or p elements would be more appropriate.

<fieldset>

```
<fieldset>
<legend [accesskey="key"]>Legend</legend>
<!--form controls here-->
</fieldset>
```

Used to demarcate related controls in a form. legend is optional in XHTML.

<form>

```
<form action="url" [enctype="enctype"
method="method">...</form>
Used to group interactive controls whose content can be
submitted for remote processing to url. For validation
form controls must be wrapped in a container tag – e.g.
fieldset, p, div etc. Form controls without a name
attribute are not submitted to url.
enctype = application/x-www-form-urlencoded |
multipart/form-data
method = get|post (N.B. - lower case!)
Use multipart/form-data with post when using file input
elements.
```

<h#>

```
<h#>Heading</h#>
# = 1..6
Judicious & ordered – i.e. h1 before h2 etc – use of
headings may help with SEO. Headings are container
elements – i.e. they can be used to contain other HTML
elements. To place other containers - such as div, p etc
– inside headings wrap them in an <object>...</object>
tag pair.
```

<iframe>

iframe is not legal in strict (X)HTML. However, browsers will accept iframes and they do have a certain utility – e.g. for inline display of a PDF document.
<iframe src="url" [frameborder="fb" height="h" width="w" scrolling="scroll" margin#="n"]>...</iframe>
 fbr = 1|0 scroll = yes|no|auto
 Frame height and width can be specified in pixels. margin# (# = height|width) in pixels.

```

height & width enable page rendering to continue while
large images are being downloaded.
```

<input>

```
<input type="type" name="name" value="text"
[disabled="disabled"]>
```

This interactive HTML element can take on a multitude of guises depending on the assignment to the type attribute.

```
type = text|password|checkbox|radio|file|hidden|submit|
reset|button
```

- ♦ text & password input elements use two additional attributes – **maxlength** & **size**. The former controls the maximum number of characters that can be typed while the latter (specified as number of characters) controls the screen width of the element. Use CSS styling when **size** fails to deliver the desired result
- ♦ checkbox & radio input elements take the additional attribute **checked**="checked". In fact all browsers treat the mere presence of **checked** as a directive to check the control. **value** is not relevant here. Use the same **name** attribute for radio controls that should be mutually exclusive – i.e. only one checked at any given time.
- ♦ In the interests of security file inputs are not populated with **value** text. Neither do they respond well to styling via CSS. This is to prevent rogue websites from masking a file input by making it look like something else. For information on styling these controls see www.explainth.at/en/html/fileupstyle.shtml
- ♦ hidden input elements can be used to store user specific information – for instance, when cookies are disabled. It should be borne in mind though that such information is easily altered.
- ♦ clicking on submit input buttons inside forms causes the form data – the information provided in all form controls with a **name** attribute – to be shipped to the **url** supplied in the **action** attribute of the form.
- ♦ clicking on reset input buttons inside forms causes all form controls to revert to their initial state.
- ♦ button input controls require an **onclick** event handler

The default appearance of buttons is browser and OS dependent and not easily amenable to CSS styling. This tends to spoil the look and feel of an otherwise well designed site. A technique for creating "buttons" with a more consistent look and feel is described here www.explainth.at/en/tricks/cssbut.shtml

```
<label>Text<input type="text" name="name"></label>

or <label for="id">Name</label><input id="id"
name="name" type="text">
```

Labels are used to attach descriptions to **<input>** elements & to transfer focus to the input when the label text is clicked. This latter feature is particularly useful with radio buttons and checkboxes where users expect such functionality. Note that the first format does not make labels clickable in **IE6**.

```
<map>
<map name="name" id="name">
<area shape="rect" href="url" coords="X1,Y1,X2,Y2">
<area shape="circle" href="url" coords="X,Y,r">
<area shape="poly" href="url" coords="X1,Y1...Xn,Yn">
</map>

```

Client-side image maps are used to create images with distinct clickable areas that can be rectangles, circles or polygons. To establish the X,Y coordinates of the vertices/center of the area/circle examine the image using an application such as Windows Paint at a high level of zoom. Note that there is **no comma** prior to the **r** part of the **coords** specification for circles. In the interests of compliance – and to make the document more accessible to the visually impaired – the **alt** attribute should be specified for each clickable area.

```
<noscript>
<noscript>...</noscript>
The content within this tag pair is only displayed if JavaScript has been disabled or is not supported. However, bear in mind that search engine robots will always examine these contents.
```

```
<?l?>
<?l?>
<li>Item 1</li>
...
<li>Item n</li>
</?l?>
```

? = u for "unordered", i.e. bullet, lists
 ?=o for "ordered", i.e. number, lists
 The CSS attributes list-style-type and list-style-image should be used to alter the default bullet/number style. See www.explainth.at/en/qr/cssqr.shtml for details.

Lists can be nested inside one another. Best practice – and XHTML compliance – requires that a nested list be wrapped correctly inside the containing **...** pair. Not all browsers handle bullet lists nested inside number lists in a satisfactory manner

```
<p>Contents</p>
The lowly paragraph container element has many of the same capabilities as its better known sibling the div.
```

```
<pre>
By default HTML normalizes white space characters – it treats repeated sequences of 0x20(space)s as one character and ignores CR/LF characters all together. Wrap text – and other HTML elements – in a <pre>...</pre> tag pair to suppress this behavior.
```

pre also suppresses wrapping of text as would occur normally in a **p** or **div** element. This can be prevented by using the following CSS styling

```
white-space:pre-wrap;white-space:-pre-wrap;white-space:-o-pre-wrap;white-space:-moz-pre-wrap;word-wrap:break-word;
```

```
<select>
<select name="name" [disabled="disabled"
multiple="multiple" size="size">
[<optgroup label="text">]
<option [value="1" selected="selected"
disabled="disabled"]>One </option>
<option>Two</option>
</optgroup>
<option>A</option>
</select>
```

size="1" displays this element as a combobox. Other values for **sizes** display a listbox. With **multiple**="multiple" **size** is assumed to be 4 (browser dependent) unless it has been specified. **disabled** causes the select to be disabled. Disabling individual options does not work in **IE6**. **optgroup** is used to separate the selection into logical subgroups.

The response of **select** elements to CSS styling leaves much to be desired – especially in **IE6**. For a partial solution see www.explainth.at/en/html/selstyle.shtml

```
<span>
<span [id="id" class="class" style="style"...</span>
spans are generic inline elements used to apply specific styling effects to their contents and/or make the contents available for programmatic manipulation.
```

```
<table>
<table border="0" cellpadding="0" cellspacing="0">
<thead>
<tr><th>...</th><th>...</th>...</tr>
</thead>
<tfoot>
<tr><td>...</td><td>...</td>...</tr>
</tfoot>
<tbody>
<tr><td>...</td><td>...</td>...</tr>
...
<tr><td>...</td><td>...</td>...</tr>
</tbody>
</table>
```

HTML table attributes provide some intrinsic styling capabilities. However, for consistent results it is better to use CSS styling for tables. For an example of what can be achieved see www.explainth.at/en/css/tabdem.shtml

While it is possible to build tables with nothing but **<tr>** and **<td>** elements, the use of header and footer elements makes tabulated information more accessible to the visually impaired – and can help robots to better index tabular information. XHTML requires **<tfoot>** to be specified **before** **<tbody>**. Note that **<th>** table cell elements stray into CSS territory and introduce visual effects.

Styling/Formatting Tags

Good practice requires a strict separation of styling from content. CSS is the right place to define visual effects with the document body only containing pure content.

In practice this breaks down somewhat since inline styling can be applied to all HTML elements. In addition, there are a small number of HTML tags that stray into styling.

Tag	Effect
bold	Bold
<big>big</big>	big

	Line break
emphasized	emphasized²
<hr>	Horizontal line
<i>italics</i>	<i>italics</i>
<small>small</small>	small
<strike>strike</strike>	strike
<sub>subscript</sub>	Normal _{subscript}
<sup>superscript</sup>	Normal ^{superscript}
<u>underline</u>	underline

Miscellanea

Use the comments
 <!-- google_ad_section_start -->
 <!-- google_ad_section_end -->
 to wrap document content that you would like the Google AdSense robot to use for indexing and delivering contextual advertising.

Notes

- Color Coding**
 (X)HTML Tag Deprecated Tag [Optional]
 Common Attribute HTML Attribute
 XHTML Attribute Comment | XHTML only
 context specific replacement CSS Attribute
- default value **onclick** (events) **0xnn** (hexadecimal)
 [a,b,...] Comma separated list of one or more of a,b,... etc.
 a is the default.
a|b|c a or b or c. a is the default.
1 # is 1..6
2 Interpretation depends on the browser.
 [id="id" class="class" style="style"] can be used with virtually all elements to help locate them (id) or to alter their appearance (class & style).
- CSS cheatsheet - www.explainth.at/en/qr/cssqr.shtml
 Tested with Internet Explorer 6+, Firefox 2+, Opera 9.1+ and Safari 3.0.3 (Windows).