

XHTML 1.0 Strict Reference Manual

Declarations

<xml> Defines the version of xml being used in the document.

Example:

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
```

Note: we will always use this statement in this course.

<DOCTYPE> Points to a specific DTD (document type definition) to be used by the document.

Example:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

Note: except for a few of the early examples, we will always use this doctype statement.

Core Attributes

Core attributes are common attributes found in most XHTML elements. The elements <base>, <html>, <meta>, <param>, <script>, <style>, and <title> only support some of the core attributes.

Attribute	Description
class	Indicates the class or classes to which a particular element belongs.
id	Specifies a unique identifier to be associated with an element.
style	Specifies an inline style for the element.
title	Supplies text that is often rendered as a tooltip when the mouse is over the element.

Index

Element	Description	Page
<a>	Anchor for hypertext link	6
<blockquote>	Used for large quotations	7
<body>	Defines the body of the document	8
 	A hard line break (a blank line)	8
<button>	Defines a push button	9
<caption>	Descriptive phrase for a table	10
<cite>	Specifies a reference to another source	11
<code>	Used to define computer code (font style change)	11
<col>	Defines attributes for table columns	12
<colgroup>	Defines groups of table column elements	13
<dd>	Marks the start of a term in a definition list	14
	Specifies deleted text (usually strikethrough)	15
<div>	Specifies a division or section in a document	15
<dfn>	Defines a definition within text (usually italics)	16
<dl>	Marks the start of a definition list	17
<dt>	Marks the start of a term in a definition list	18

Element	Description	Page
<code></code>	Places emphasis (usually italics or bold) on text	18
<code><fieldset></code>	Draws a box around the elements within it	19
<code><form></code>	Defines elements that make up a form	20
<code><h1></code> - <code><h6></code>	Header definitions	21
<code><head></code>	Information about the XHTML document	21
<code><hr /></code>	Inserts a horizontal line in the document	22
<code><html></code>	Parent section of all XHTML documents	22
<code></code>	Specifies an image placed in a document	23
<code><input></code>	Specifies start of a user input field	24
<code><ins></code>	Text has been inserted (usually underlined)	25
<code><kbd></code>	Defines keyboard input (font style change)	25
<code><label></code>	Labels a prompt and form control for interaction	26
<code><legend></code>	Defines a caption for a <code><fieldset></code> element	27
<code></code>	Defines a list element in <code></code> and <code></code> lists	28
<code><link></code>	The relationship between two linked documents.	29
<code><map></code>	Client-side image map with clickable regions	30

Element	Description	Page
<meta>	Describes the document	31
<noscript>	Specifies alternate content if no scripting	31
<object>	Describes an embedded object in the document	32
	Defines an ordered list of elements	33
<optgroup>	Defines an option group	34
<option>	Specifies options in a drop-down list	35
<p>	Defines a paragraph within the document	36
<param>	Specifies run-time values for objects	36
<pre>	Specifies preformatted text preserving structure	37
<q>	Defines a short quotation	37
<samp>	Describes sample output (usually font change)	38
<script>	Adds scripting capabilities to a document	38
<select>	Creates a drop-down list	39
	Groups inline elements in a document	40
<style>	Creates an internal style sheet (CSS)	41
<sub>	Creates subscripted text	42

Element	Description	Page
<code><sup></code>	Creates superscripted text	42
<code><table></code>	Defines a table	43
<code><tbody></code>	Defines the body of a table	45
<code><td></code>	Defines a cell in each row of a table	47
<code><textarea></code>	Allows for multiple lines of text input by user	48
<code><tfoot></code>	Describes a table footer	49
<code><th></code>	Describes a header row in a table	50
<code><thead></code>	Describes a table header	51
<code><title></code>	Defines the title of a document	52
<code><tr></code>	Describes a row in a table	52
<code></code>	Defines an unordered list	53
<code><var></code>	Defines a variable in text (usually a font change)	54
<code>, <big>, <i> <small>, <tt></code>	Legal in XHTML Strict DTD but typically done using CSS	54

<a>

The <a> element is used to specify an anchor within your document. You can use an anchor to link to another document or to a shortcut within the same document.

Required attributes: None

Optional attributes:

href: specifies the URL that you wish the browser to open when the user clicks on the link. This is the most commonly used attribute for the anchor tag.

rel: Specifies the relationship between the current document and the target URL. Possible values include *alternate*, *stylesheet*, *start*, *next*, *prev*, *contents*, *index*, *glossary*, *copyright*, *chapter*, *section*, *subsections*, *appendix*, *help*, and *bookmark*. The *stylesheet* value will be commonly used.

Examples:

absolute link

```
<a href="http://www.cnn.com"> click here for news </a>
```

relative link

```
<a href="home.html"> Home Page </a>
```

<blockquote>

The <blockquote> element specifies a long quotation. Typically, a blockquote renders in a browser by placing white space on either side of the text (both a left and right indent is used).

Required attributes: None

Optional attributes:

`cite`: if the quote is from an Internet source, a URL, will point to where the quote is from.

Examples:

```
<blockquote>
  <p> Four score and seven years ago our fathers
    brought forth on this...
  </p>
</blockquote>
```

```
<blockquote
  cite="http://en.wikipedia.org/wiki/Gettysburg_A
  ddress">
  <p> Four score and seven years ago, our fathers
    brought forth on this...
  </p>
</blockquote>
```

<body>

The <body> element defines the document's body. It contains all the content for the document.

Required attributes: None

Optional attributes: None

Example:

```
<body>
  <!-- content goes here -->
</body>
```


The
 element inserts a single line break. The break element is an empty element, which means it contains no content. Browser compatibility requires the blank space before the backslash character.

Required attributes: None

Optional attributes: None

Example:

```
<p>
  This line contains <br />
  a line break.
</p>
```


<button>

The <button> element specifies a push button within your document. A button element can contain text or images.

Required attributes: None

Optional attributes:

`disabled`: makes the button disabled and removes the control from tab navigation. The control's value won't be submitted when the form is submitted.

`name`: specifies the unique name for the button.

`type`: specifies the type of button. Valid values include `button`, `reset`, and `submit`. If no type is specified the default value is `submit`.

`value`: removes the control from tab navigation and won't submit the control's value when the form is submitted.

Example:

```
<div>
  <button type="submit" name="continue">
    <strong> Continue to next page</strong>
    
  </button>
</div>
```

<caption>

The <caption> element is used to place a caption above a table; it is not contained within a row or cell. The caption element is placed immediately after the start tag for the table. You can specify a single caption per table.

Required attributes: None

Optional attributes: None

Example:

```
<body>
  <table border="1" summary="Prices">
    <caption> Our Pizza Selections</caption>
    <tr>
      <td scope="col"> Pizza type</td>
      <td scope="col"> Medium</td>
      <td scope="col"> Large</td>
    </tr>
    <tr>
      <td>Thin crust</td>
      <td>$4.99</td>
      <td>$7.99</td>
    </tr>
    <tr>
      <td>Deep Dish</td>
      <td>$6.99</td>
      <td>$9.99</td>
    </tr>
  </table>
</body>
```

<cite>

The <cite> element specifies a reference to another resource.

Required attributes: None

Optional attributes: None

Example:

```
<p>
  <Restaurant critic <cite> Sam Waters </cite>
  had this to say about our eatery:</p>
  <blockquote cite="http://example.com/food/review"
    />
    <p> The atmosphere is cozy and food is terrific.
  </p>
  <p><cite>Bon Appetit, March 2008</cite></p>
</blockquote>
</p>
```

<code>

The <code> element is used to define text as computer code.

Required attributes: None

Optional attributes: None

Example:

```
<pre><code>
  function helloWorld() {
    var button = document.getElementById("button");
    if (button) {
      button.onclick = function() {
        alert("Hello World");
      }
    }
  }
</code></pre>
```

<col>

The <col> element is used to define the attribute values for one or more columns in a table. You can use the col element to specify whatever attributes you want to be in common for each column in a table.

Required attributes: None

Optional attributes:

align: specifies the alignment of the text within a cell. Possible values include *center*, *char*, *justify*, *left*, and *right*.

char: specifies which character the text should be aligned on. This requires the use of the *align* attribute with the value set as *char*.

charoff: specifies in pixels or as a percentage how far the alignment should be adjusted to the first character to align on. This requires the use of the *align* attribute with the value set as *char*.

span: specifies the number of columns the *col* should occupy as a number.

valign: specifies the vertical alignment of the text within a cell. Possible values include *baseline*, *bottom*, *middle*, and *top*.

width: specifies the width of the table in pixels or as a percent.

Example:

```
<table>
  <col width="33" valign="middle" />
  <col width="50" valign="top" />
  <tr>
    <td> . . . </td>
    <td> . . . </td>
  </tr>
</table>
```

<colgroup>

The <colgroup> element is used to define a group of col elements. You should use this element only within a table element. Use this element to group columns for formatting.

Required attributes: None

Optional attributes:

align: specifies the alignment of the text within a cell. Possible values include center, char, justify, left, and right.

char: specifies which character the text should be aligned on. This requires the use of the align attribute with the value set as char.

charoff: specifies in pixels or as a percentage how far the alignment should be adjusted to the first character to align on. This requires the use of the align attribute with the value set as char.

span: specifies the number of columns the col should occupy as a number.

valign: specifies the vertical alignment of the text within a cell. Possible values include baseline, bottom, middle, and top.

width: specifies the width of the table in pixels or as a percent.

Example:

```
<table>
  <colgroup>
    <col width="33" valign="middle" />
    <col width="50" valign="top" />
  </colgroup>
  <tr>
    <td> . . . </td>
    <td> . . . </td>
  </tr>
</table>
```

<dd>

The <dd> element marks the start of a definition for a term within a definition list (see <dl> element). A definition term must be within a set of definition list (<dl>) elements.

Required attributes: None

Optional attributes: None

Example:

```
<dl>
  <dt>Pizza</dt>
  <dd>
    <p>A flat, open-faced baked pie of Italian
      origin, consisting of a layer of bread
      dough covered with tomato sauce, cheese,
      and a wide variety of optional
      toppings.</p>
  </dd>
  <dt>Pasta</dt>
  <dd>
    Unleavened dough that is molded into a
    variety of shapes and boiled.
  </dt>
</dl>
```


The element can be used to specify text that has been deleted in a document. It is often used when you're editing a document and trying to keep track of all the original content. Most web browsers display the text with a line through it.

Required attributes: None

Optional attributes:

`cite`: if the quote is from an Internet source, a URL will point to where the quote is from.

`datetime`: specifies the data and time that the text was deleted.

Example:

```
<p>  
  This text contains a <del> deleted</del> word.  
</p>
```

<div>

The <div> element specifies a division or section within a document. It has no default rendering. Typically, most browsers will induce a line return at the end of a division.

Required attributes: None

Optional attributes: None

Example:

```
<div>  
  <p>This text is contained in a division. </p>  
</div>
```

<dfn>

The <dfn> element is used to define a definition within text.

Required attributes: None

Optional attributes: None

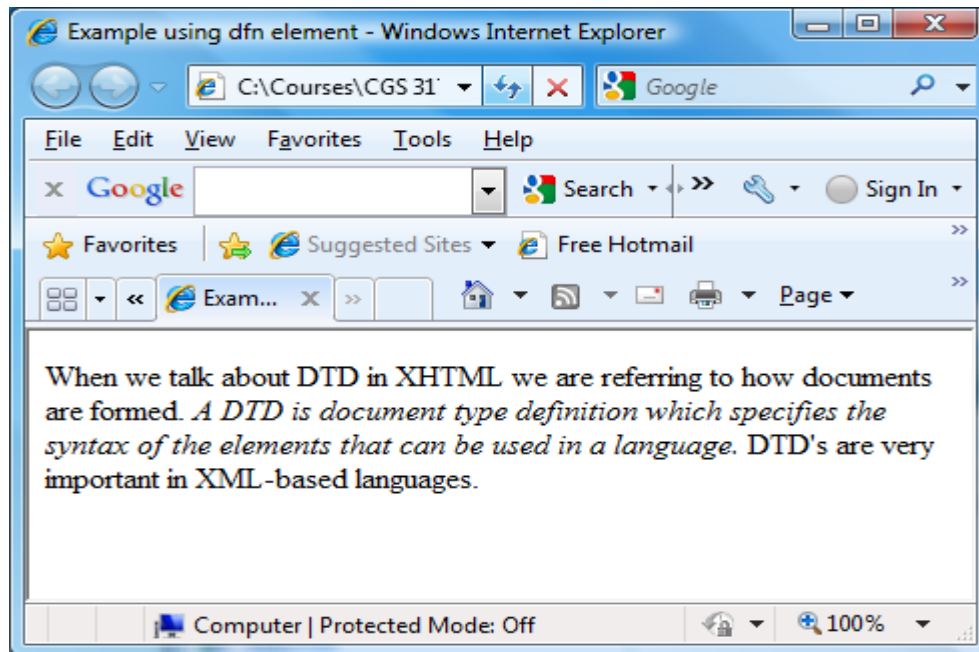
Example:

<p>When we talk about DTD in XHTML we are referring to how documents are formed.

<dfn> A DTD is document type definition which specifies the syntax of the elements that can be used in a language.

</dfn>DTD's are very important in XML-based languages.

</p>



<dl>

The <dl> element marks the start of a definition list. A definition list must be in the body of the document.

Required attributes: None

Optional attributes: None

Example:

```
<dl>
  <dt>Pizza</dt>
  <dd>
    <p>A flat, open-faced. . . .</p>
  </dd>
  <dt>Pasta</dt>
  <dd>
    Unleavened dough . . .
  </dt>
</dl>
```

<dt>

The <dt> element marks the start of a term within the definition list. A definition term must be within a set of <dl> elements.

Required attributes: None

Optional attributes: None

Example:

```
<dl>
  <dt>Pizza</dt>
  <dd>
    <p>A flat . . . </p>
  </dd>
  <dt>Pasta</dt>
  <dd>
    Unleavened . . .
  </dd>
</dl>
```


The element is used to define text that will be emphasized when rendered.

Required attributes: None

Optional attributes: None

Example:

```
<p>
  This text is <em> emphasized </em> here. </p>
```

<fieldset>

The <fieldset> element is used to draw a box around the element within it. It is often used to when grouping elements within a form to help nonvisual browsers.

Required attributes: None

Optional attributes: None

Example:

```
<form method="post" action="/cgi-bin/handler.cgi">
  <fieldset>
    <legend> Subscribe to our mailing list</legend>
    <label> for="email" Enter your email
      address</label>
    <input type="text" id="email" name="email" />
    <input type="submit" name="subscribe"
      value="Subscribe" />
  </fieldset>
</form>
```

<form>

The `<form>` element is used to contain other elements that make up a form used for user input. Forms are used to pass data from controls to a specified URL.

Required attributes:

`action`: specifies the URL where to send the data when the user clicks the Submit button on the form. It is also possible that the form was submitted using a script, in which case it will act as if the Submit button was clicked.

Optional attributes:

`accept`: specifies a comma-separated list of MIME types that the server accepts and processes correctly as files.

`accept-charset`: A comma-separated lists of possible character sets for the form data.

`enctype`: specifies the MIME type used to encode the data within the form when the method attribute is set to `post`.

`method`: specifies the HTTP method for sending the data to the URL specified by the `action` attribute. Possible values are `get` and `post`. Specifying `get` appends the data after the URL itself in the URL. Specifying `post` places the data within the request itself.

Example:

```
<form method="post" action="/cgi-bin/handler.cgi">
  <fieldset>
    <legend> Subscribe to our mailing list</legend>
    <label> for="email" Enter your email
      address</label>
    <input type="text" id="email" name="email" />
    <input type="submit" name="subscribe"
      value="Subscribe" />
  </fieldset>
</form>
```

<h1> - <h6>

The <h1> - <h6> elements define headers within the document. An <h1> element specifies the largest header while the <h6> element specifies the smallest header.

Required attributes: None

Optional attributes: None

Example:

```
<h1>
  This is a large header
</h1>
<h2>
  This is a smaller header
</h2>
```

<head>

The <head> element contains information about the HXTML document. Typically, nothing in the head section is displayed to the user in a browser.

Required attributes: None

Optional attributes:

profile: specifies a space-separated list of URLs that point to metadata profiles about the page.

Example:

```
<head>
  <title> My document </title>
</head>
```

<hr />

The <hr /> element is used to display a horizontal line when rendering the document. The <hr /> element is an empty element.

Required attributes: None

Optional attributes: None

Example:

```
<body>
  <p> This is a line of text. </p>
  <hr />
  <p> There is a horizontal line above this text.
  </p>
</body>
```

<html>

The <html> element tells a browser that the document is an HTML document and is the parent of all the other major sections within the document.

Required attributes:

`xmlns:` used to define the XML namespace attribute. However, most validations assume the value “`http://www.w3.org/1999/xhtml`” since it is a fixed value.

Optional attributes: None

Example:

```
<html xmlns="http://www.w3.org/1999/xhtml">
```


The element specifies an image within a document. The element is an empty element.

Required attributes:

alt: specifies brief description of the image.

src: specifies the URL of the image to be displayed.

Optional attributes:

height: specifies the height of the image.

ismap: specifies that the image should be used as a server-side image map.

longdesc: specifies a URL to a document that contains a long description of the image.

usemap: specifies the image as a client-side image map. The browser should look for the **map** and **area** elements for the specifics.

Example:

```

```

<input>

The <input> element specifies the start of an input field where the user can enter data.

Required attributes: None

Optional attributes:

`accept`: specifies a comma-separated list of MIME types that indicate the type used for a file transfer when using the `file` value for the `type` attribute.

`alt`: specifies a brief description of the image.

`checked`: specifies that the `input` element should be checked (selected). This is valid with the `checkbox` and `radio` types.

`disabled`: if set to the value `disabled`, it will make the <input> control disabled and not respond to user interaction.

`ismap`: specifies the image should be used as a server-side image map.

`maxlength`: specifies the maximum number of characters allowed in a text field when using the `text` value in the `type` attribute.

`name`: specifies the unique name for the `input` element.

`readonly`: specifies the value `readonly`. It indicates that the value of the field cannot be modified when using the `text` value in the `type` attribute.

`size`: specifies the size of the input element in characters.

`src`: specifies the URL of an image to be displayed when using the value `image` with the `type` attribute.

`type`: specifies the type of the input element. Valid values are `button`, `checkbox`, `file`, `hidden`, `image`, `password`, `radio`, `reset`, `submit`, and `text` (default).

`usemap`: specifies the image as a client-side image map. The browser should look for the `map` and `area` elements for the specifics.

`value`: the value attribute has different meaning based on the `type` attribute's value. For `button`, `reset`, and `submit` types, it specifies the text that appears on the button. For the `image` type, it specifies the symbolic result of the field passed to a script. For the `checkbox` and `radio` types, it specifies the result of the input element when clicked.

Example:

```
<p>
  <input type="submit" name="login" value="Log In"
        />
</p>
```


<ins>

The <ins> element is used to specify text that has been inserted into a document. It is often used when editing a document and trying to keep track of all the content. (See also the element.) In most web browsers, the displayed text is underlined.

Required attributes: None

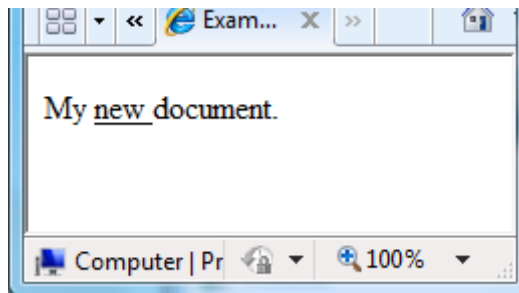
Optional attributes:

`cite`: if the quote is from an Internet source, a URL will point to where the quote is from.

`datetime`: specifies the data and time that the text was inserted.

Example:

```
<p>My <ins> new </ins> document. </p>
```



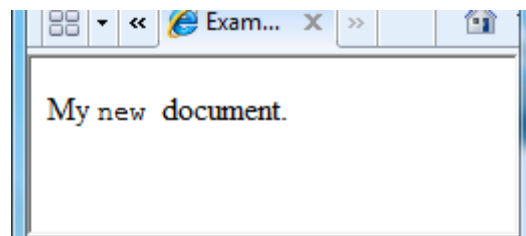
<kbd>

The <kbd> element is used to define text as being keyboard text. Typically this means a font style difference.

Required attributes: None

Optional attributes: None

Example: <p> My <kbd> new </kbd> document. </p>



<label>

The <label> element provides a programmatic association between the prompt and the form control, so that assistive technology can interact with the element effectively. In visual browsers, if the user clicks the text within the label element, the element will manipulate the control as if the user clicked on the control itself.

Required attributes: None

Optional attributes:

for: specifies which element within a form the label is attached to and will manipulate. This attribute should be set to the ID of another form element.

Example:

```
<form method="post" action="cgi-bin/handler.cgi">
  <fieldset>
    <legend> Subscribe to our mailing list</legend>
    <p>
      <label for="email">Your email
        address</label>
      <input type="text" name="email" id="email" />
    </p>
  </fieldset>
  <p><input type="submit" id="subscribe"
    value="Subscribe" /> </p>
</form>
```

<legend>

The <legend> element is used to define a caption for a fieldset.

Required attributes: None

Optional attributes: None

Example:

```
<form method="post" action="cgi-bin/handler.cgi">
  <fieldset>
    <legend> Subscribe to our mailing list</legend>
    <p>
      <label for="email">Your email
        address</label>
      <input type="text" name="email" id="email" />
    </p>
  </fieldset>
  <p><input type="submit" id="subscribe"
    value="Subscribe" /> </p>
</form>
```

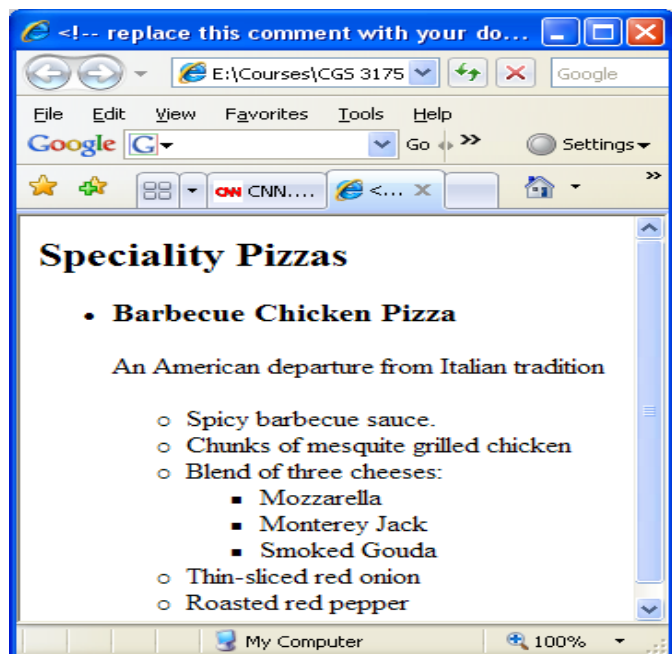

The element is used to indicate each item to be displayed in a list. The list item element is used in both ordered (ol) and unordered lists (ul).

Required attributes: None

Optional attributes: None

Example:

```
<h2>Specialty Pizzas</h2>
<ul>
  <li>
    <h3>Barbecue Chicken Pizza</h3>
    <p>An American departure from Italian tradition</p>
    <ul>
      <li>Spicy barbecue sauce.</li>
      <li>Chunks of mesquite grilled chicken</li>
      <li>Blend of three cheeses:
        <ul>
          <li>Mozzarella</li>
          <li>Monterey Jack</li>
          <li>Smoked Gouda</li>
        </ul>
      </li>
      <li>Thin-sliced red onion</li>
      <li>Roasted red pepper</li>
    </ul>
  </li>
</ul>
```



<link>

The <link> element is used to define the relationship between two linked documents. It is most often used to link external style sheets to the current document.

Required attributes: None

Optional attributes:

charset: specifies the character set being used by the document being linked.

href: specifies the URL pointing to the document that is being linked.

media: specifies the type of media that the document that is being linked is meant for. Common values include *all*, *braille*, *print*, *projection*, *screen*, and *speech*.

rel: specifies the relationship between the linked document and the current document. Common values include *alternate*, *appendix*, *bookmark*, *chapter*, *contents*, *copyright*, *glossary*, *help*, *index*, *next*, *prev*, *section*, *start*, *stylesheet*, and *subsection*.

rev: This is the opposite of *rel*. It defines the relationship between the current document and the linked document. Same possible values as the *rel* attribute.

type: This attribute specifies the MIME type of the target URL. The most common value is *text/css* for use with external style sheets, and *text/javascript* for use with JavaScript files.

Example:

```
<link rel="stylesheet" href="thirdCSS.css"
      type="text/css" />
```

<map>

The <map> element specifies a client-side image map with clickable regions.

Required attributes:

id: specifies a unique name for the map element.

Optional attributes:

name: specifies a unique name for the map element.

Example:

```
<body>
  <p>
    <map id="map" name="map">
      <area shape="rectangle" href="square.html"
        coords="20,27 97,76" alt="order square
        pizza" />
      <area shape="circle" href="circle.html"
        coords="167,51 35" alt="order round
        pizza" />
    </map>
    
  </p>
</body>
```

<meta>

The `<meta>` element provides information about the document. This information may be used by search engines that catalog pages on the Internet. You may use the `meta` element to provide keywords and descriptions for the search engines to use in order to catalog your document. You can also use the `meta` element to allow for automated refreshes using the `http-equiv` attribute. The term *meta* refers to metadata, which is a term that is often described as “data about data”. The `meta` element provides data about the data in the document.

Required attributes:

`content`: This is the value or data to be associated with a name or `http-equiv`.

Optional attributes:

`http-equiv`: This attribute is used to connect the `content` attribute value to a specific HTTP response header. You can use it to request the browser to do something or to reference information about the document from an external source.

`name`: This attribute is used to assign extra information to a document. The value of this attribute comes from the `content` attribute. Some common names include `author`, `keywords`, `description`, and `summary`.

`scheme`: This attribute is used to define a format that is used to interpret the value set in the `content` attribute.

Example:

```
<meta name="keywords" content="HTML, XHTML, CSS,  
      JavaScript." />
```

<noscript>

The `<noscript>` element is used to specify alternate content if scripting is not allowed on the client side.

Required attributes: None

Optional attributes: None

<object>

The <object> element specifies an object to be embedded into the document. Use the param element to specify any values that need to be passed.

Required attributes: None

Optional attributes:

archive: A spaced-separated list of URLs that contain any resources for the object.

classid: This attribute specifies the location of an object's implementation through a URL.

codebase: This attributed specifies the base path to be used when relative URLs are provided for classid, data, and archive attributes.

codetype: Specifies the MIME type of the code referred to in the classid attribute.

data: Specifies a URL that refers to any data for the object.

declare: Specifies that the object should only be declared and created when needed.

height: Specifies the height of the object in pixels.

name: Specifies a unique name for the object, which will be used in JavaScript to access the object. The name attribute is commonly used so that this control can be used in form submissions.

standby: Specifies the text to display when the object is being loaded.

type: Specifies the MIME type of data in the data attribute.

usemap: Specifies a client-side map in the form of a URL that will be used with the object.

width: Specifies the width of the object in pixels.

Example:

```
<p>A short video about making pizza</p>
<div>
  <object data="makingpizza.mpg" type="video.mpeg"
    width="370" height="275">
    <param name="src" value="makingpizza.mpg" />
  </object>
</div>
```

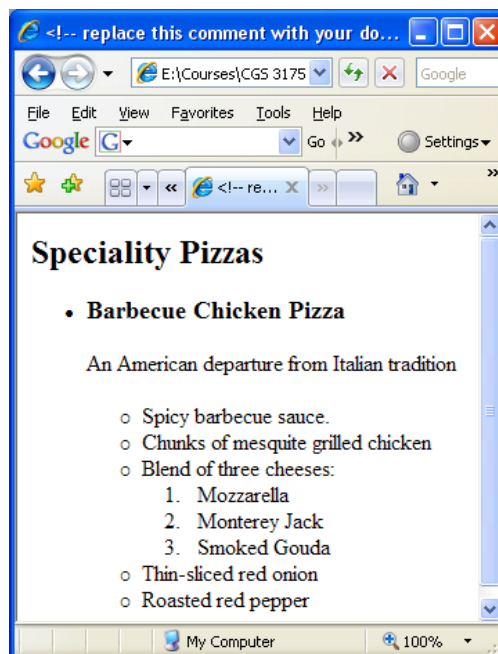

The element is used to signify a list. An ordered list must be within the body of the document.

Required attributes: None

Optional attributes: None

Example:

```
<h2>Specialty Pizzas</h2>
<ul>
  <li>
    <h3>Barbecue Chicken Pizza</h3>
    <p>An American departure from Italian tradition</p>
    <ul>
      <li>Spicy barbecue sauce.</li>
      <li>Chunks of mesquite grilled chicken</li>
      <li>Blend of three cheeses:
        <ol>
          <li>Mozzarella</li>
          <li>Monterey Jack</li>
          <li>Smoked Gouda</li>
        </ol>
      </li>
      <li>Thin-sliced red onion</li>
      <li>Roasted red pepper</li>
    </ul>
  </li>
</ul>
```



<optgroup>

The <optgroup> element is used to define an option group. It allows you to group choices together.

Required attributes:

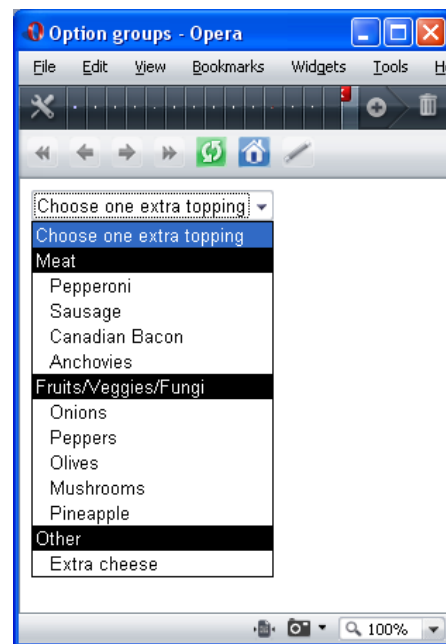
label: Specifies the label to be used for the option group.

Optional attributes:

disabled: specifies the option group as being disabled. The user won't be able to interact with the controls.

Example:

```
<select name="extra">
  <option value="" selected="selected">Choose one extra
    topping</option>
  <optgroup label="Meat">
    <option>Pepperoni</option>
    <option>Sausage</option>
    <option>Canadian Bacon</option>
    <option>Anchovies</option>
  </optgroup>
  <optgroup label="Fruits/Veggies/Fungi">
    <option>Onions</option>
    <option>Peppers</option>
    <option>Olives</option>
    <option>Mushrooms</option>
    <option>Pineapple</option>
  </optgroup>
  <optgroup label="Other">
    <option>Extra cheese</option>
  </optgroup>
</select>
```



<option>

The <option> element specifies an option in the drop-down or list control. It is common to have several option elements for each list.

Required attributes: None

Optional attributes:

disabled: sets the control as disabled.

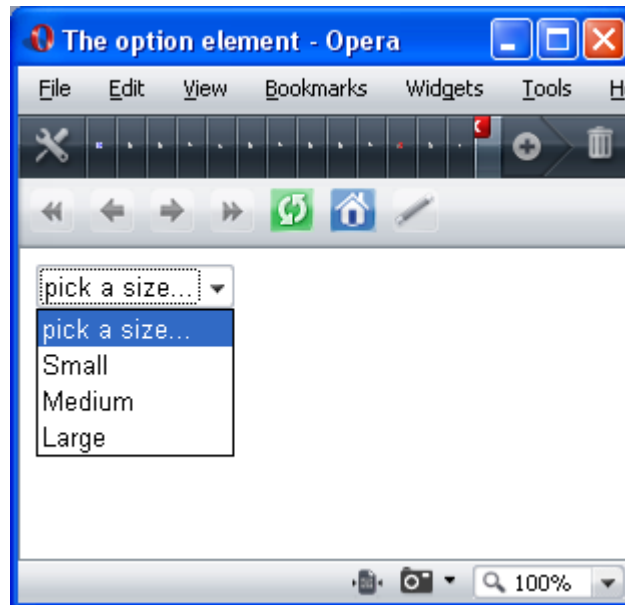
label: allows authors to provide a shorter label for the content of an option element.

selected: specifies the value that is selected by default.

value: specifies the value for the option that will be sent to the server.

Example:

```
<select name="size">
  <option value="" selected="selected">pick a
    size...</option>
  <option value="1">Small</option>
  <option value="2">Medium</option>
  <option value="3">Large</option>
</select>
```



<p>

The <p> element defines a paragraph within the document.

Required attributes: None

Optional attributes: None

Example:

```
<p>This is a paragraph of text. </p>  
<p>This is another paragraph of text.</p>
```

<param>

The <param> element allows you to set run-time values for objects that have been inserted into a document.

Required attributes: None

Optional attributes: None

<pre>

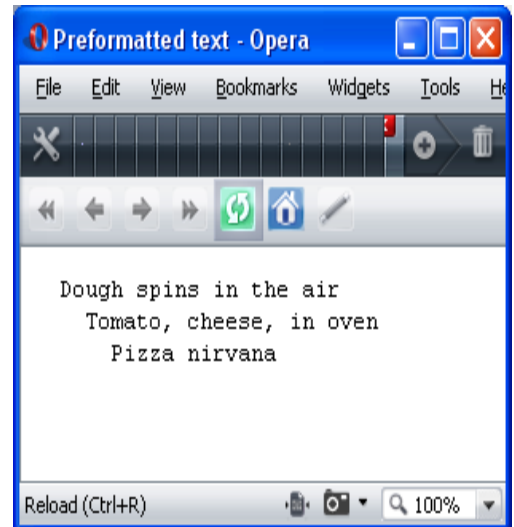
The <pre> element specifies the text as preformatted text and usually preserves spaces and line breaks.

Required attributes: None

Optional attributes: None

Example:

```
<body>
  <pre>
    Dough spins in the air
      Tomato, cheese, in oven
        Pizza nirvana
  </pre>
```



<q>

The <q> element defines a short quotation.

Required attributes: None

Optional attributes:

cite: If the quote is from an Internet source, a URL will point to where the quote is from.

Example:

```
<p>Norm Deplume, food critic for <cite>The Gotham Examiner</cite>, recently commended our geek-friendly attitude, even saying that we're  
<q  
  cite="http://example.com/food/reviews/SpaghettiCruft/  
  ">a popular  
spot for the neighborhood technophiles to linger with  
their laptops.</q></p>
```

< samp >

The `< samp >` element is used to specify sample output. It works similar to the `code`, `kdb`, and `var` elements.

Required attributes: None

Optional attributes: None

< script >

The `< script >` element plays a key role in making your website more dynamic and feature-rich. It allows you to add scripting languages to your XHTML documents that respond to user actions. We'll see this element when we get to JavaScript later in the semester.

Required attributes:

`type`: This attribute defines the MIME type of the script included. This is typically set as `text/javascript` when using JavaScript.

Optional attributes:

`charset`: defines the character encoding that is used in the script.

`defer`: tells the browser that the script won't generate any document content, so it can continue parsing and drawing the page.

`src`: specifies the URL to point to a document that contains the script.

Example:

```
<head>
  <title>JavaScript Basics</title>
  <script src="myscript.js"
    type="text/javascript">
  </script>
</head>
```

<select>

The <select> element is used to create a drop-down list.

Required attributes: None

Optional attributes:

`disabled`: makes the select element disabled and unresponsive to user interaction..

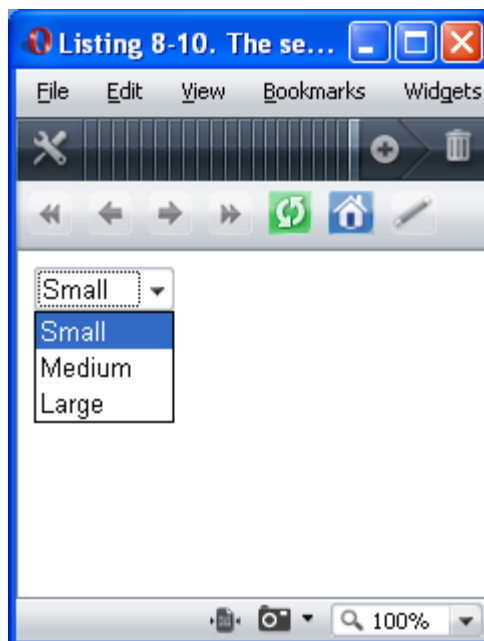
`multiple`: specifies that more than one item can be selected at a time.

`name`: specifies the unique name for the select element.

`size`: specifies the number of visible items in the drop-down list.

Example:

```
<body>
<form method="post" action="/cgi-
    bin/formhandler.cgi">
  <select name="size">
    <option>Small</option>
    <option>Medium</option>
    <option>Large</option>
  </select>
</form>
</body>
```

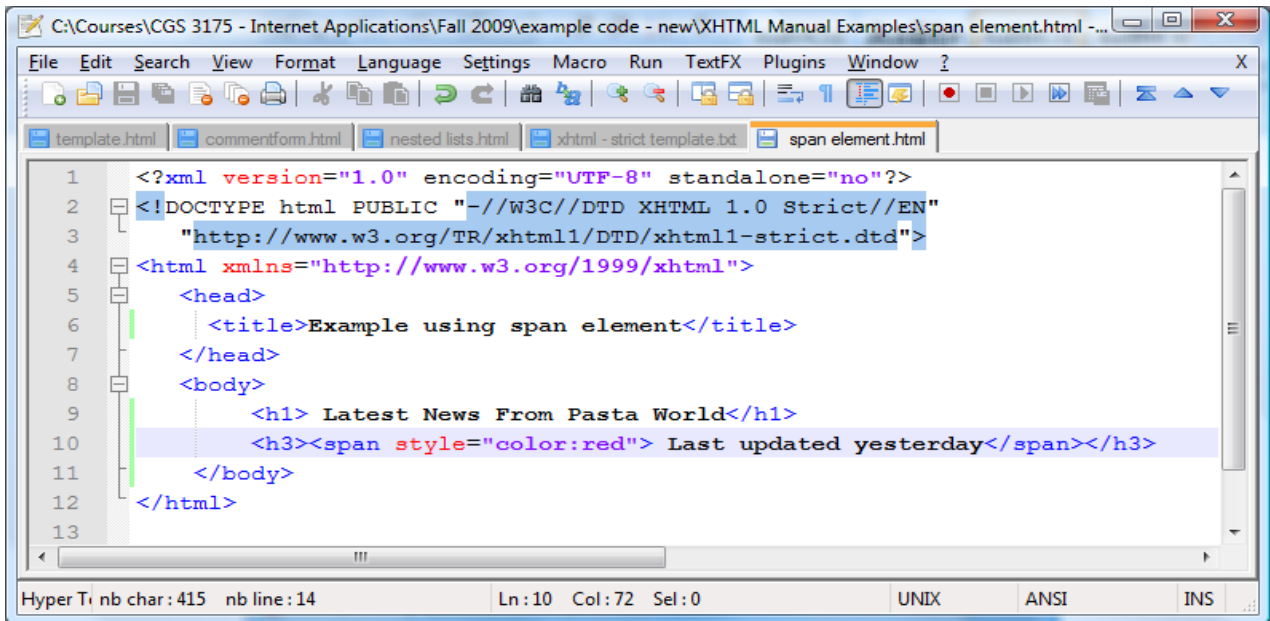


The element is used to group inline elements within a document.

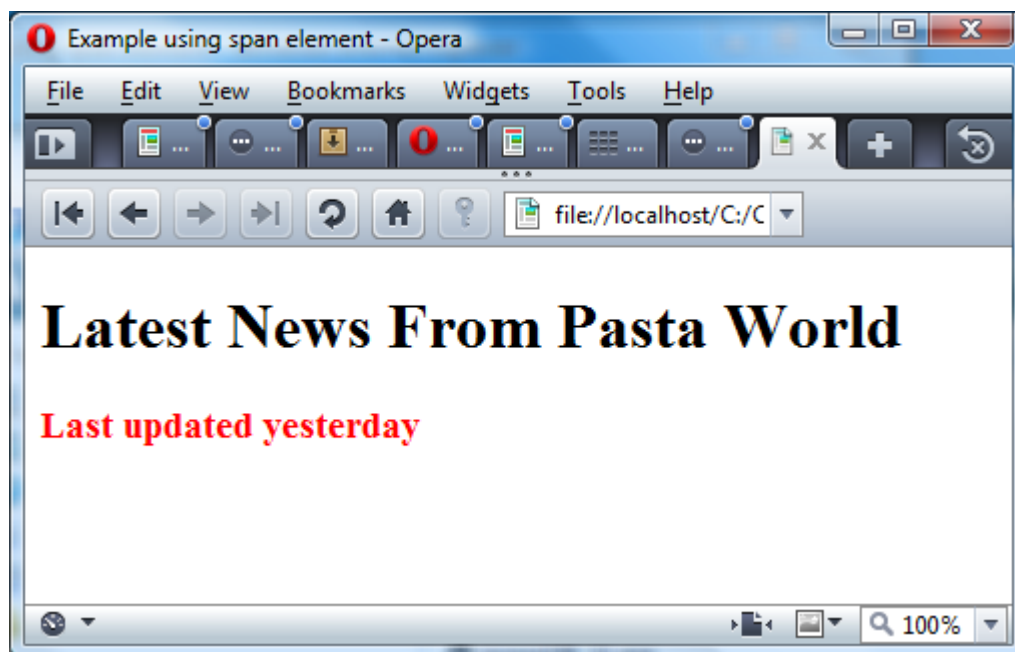
Required attributes: None

Optional attributes: None

Example:



```
1 <?xml version="1.0" encoding="UTF-8" standalone="no"?>
2 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
3 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
4 <html xmlns="http://www.w3.org/1999/xhtml">
5 <head>
6 <title>Example using span element</title>
7 </head>
8 <body>
9 <h1> Latest News From Pasta World</h1>
10 <h3><span style="color:red"> Last updated yesterday</span></h3>
11 </body>
12 </html>
13
```



<style>

The <style> element's sole purpose is to allow you to create internal style sheets for your documents. The style element provides a placeholder for internal styles that you can use throughout the document. The style element must be contained within the head element.

Required attributes:

type: defines the style type. It is normally set to `text/css`.

Optional attributes:

media: for the specifics on the media types.

Example:

```
<head>
<style type="text/css">
    <!-- style definitions go here
    -->
</style>
</head>
```


The <sub> element specifies text that is subscripted text. Subscript text is typically rendered in a smaller font below the normal text.

Required attributes: None

Optional attributes: None

Example:

```
<h1>Latest News  
  from Pasta  
  World </h1>  
<h3><span  
  style="color:red ">Last updated on  
  <sup>yesterday</sup>  
</span></h3>
```



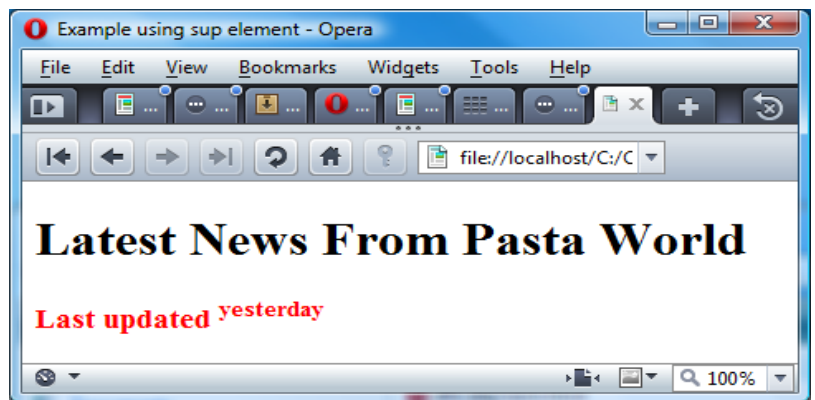
The <sup> element specifies text that is superscripted text. Superscript text is typically rendered in a smaller font raised above the normal text.

Required attributes: None

Optional attributes: None

Example:

```
<h1>Latest  
  News from  
  Pasta  
  World </h1>  
<h3><span style="color:red ">Last updated on  
  <sup>yesterday</sup>  
</span></h3>
```



<table>

The <table> element defines where a table will exist. A table is used to hold tabular data. You can place table headers, rows, cells, and other tables within a table.

Required attributes: None

Optional attributes:

`border`: specifies the width of the border for a table in pixels.

`cellpadding`: specifies the amount of space between the cell walls and the contents in pixels or as a percentage.

`cellspacing`: specifies the amount of space between cells in pixels or as a percentage.

`frame`: specifies how the outer borders of a table should be displayed.

This attribute is used along with the `border` attribute. Possible values are `above`, `below`, `border`, `box`, `hsides`, `lhs`, `rhs`, `void`, and `vsides`.

`rules`: specifies the divider lines used for horizontal and vertical lines.

This attribute is used along with the `border` attribute. Possible values are `all`, `cols`, `group`, `none`, and `rows`.

`summary`: specifies the summary of what the table is for browsers that provide speech synthesizing and nonvisual capabilities.

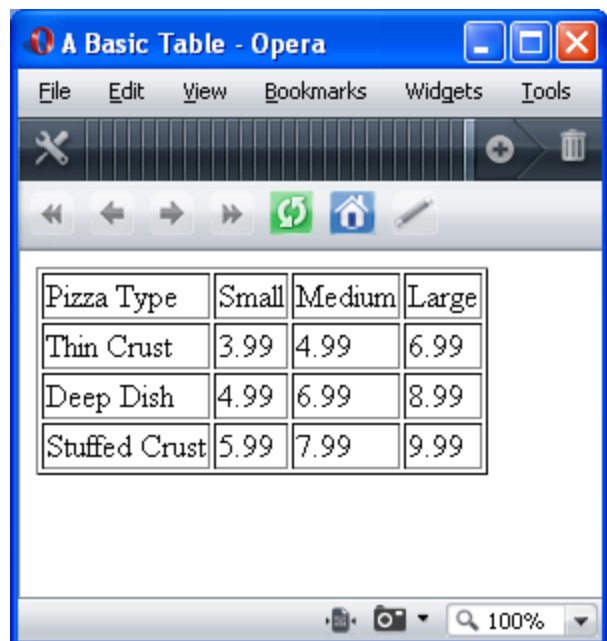
`width`: specifies the width of the table in pixels or as a percentage.

Example:

```

<body>
  <table border="1" summary="Prices for types of pizza
    by size">
    <tr>
      <td scope="col">Pizza Type</td>
      <td scope="col">Small</td>
      <td scope="col">Medium</td>
      <td scope="col">Large</td>
    </tr>
    <tr>
      <td scope="row">Thin Crust</td>
      <td>3.99</td>
      <td>4.99</td>
      <td>6.99</td>
    </tr>
    <tr>
      <td scope="row">Deep Dish</td>
      <td>4.99</td>
      <td>6.99</td>
      <td>8.99</td>
    </tr>
    <tr>
      <td scope="row">Stuffed Crust</td>
      <td>5.99</td>
      <td>7.99</td>
      <td>9.99</td>
    </tr>
  </table>
</body>

```



<tbody>

The <tbody> element is used to define a table body section. It is contained within a <table> element and you may have multiple tbody elements. The <thead>, <tbody>, and <tfoot> elements allow you to group rows within a table easily.

Required attributes: None

Optional attributes:

align: specifies the alignment of the text within a cell. Possible values are center, char, justify, left, and right.

char: specifies which character the text should be aligned on. This requires the use of the align attribute with the value set as char.

charoff: specifies in pixels or as a percentage how far the alignment should be adjusted to the first character to align on. This requires the use of the align attribute with the value set as char.

valign: specifies the vertical alignment of cell content. Possible values are baseline, bottom, middle, and top.

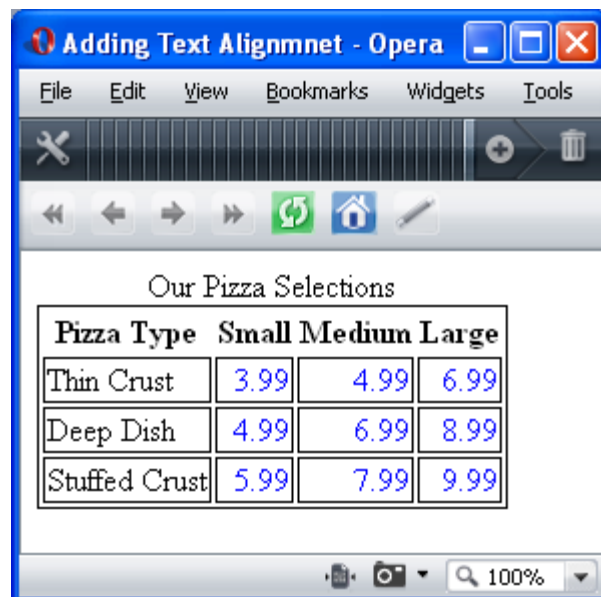
Example:

```
<head>
  <title>Adding Text Alignment</title>
  <style type="text/css">
    td {border:1px solid black;}
    table {border:1px solid black;}
    td.number{text-align: right;color: blue;}
    td.text{text-align: left;}
  </style>
</head>
<body>
  <table summary="This table provides the pricing
    information for pizzas">
    <caption>Our Pizza Selections</caption>
    <thead>
      <tr>
        <th scope="col">Pizza Type</th>
        <th scope="col">Small</th>
```

```

        <th scope="col">Medium</th>
        <th scope="col">Large</th>
    </tr>
</thead>
<tbody>
    <tr>
        <td class="text" scope="row">Thin Crust</td>
        <td class="number">3.99</td>
        <td class="number">4.99</td>
        <td class="number">6.99</td>
    </tr>
    <tr>
        <td class="text" scope="row">Deep Dish</td>
        <td class="number">4.99</td>
        <td class="number">6.99</td>
        <td class="number">8.99</td>
    </tr>
    <tr>
        <td class="text" scope="row">Stuffed
            Crust</td>
        <td class="number">5.99</td>
        <td class="number">7.99</td>
        <td class="number">9.99</td>
    </tr>
</tbody>
</table>
</body>

```



<td>

The <td> element is used to hold each cell within a row in a table.

Required attributes: None

Optional attributes:

abbr: specifies a shortened version of the content in a cell in text.

align: specifies the alignment of cell content. Possible values are center, char, justify, left, and right.

axis: associates a cell with a conceptual category.

char: specifies which character the text should be aligned on. This requires the use of the align attribute with the value set as char.

charoff: specifies in pixels or as a percentage how far the alignment should be adjusted to the first character to align on. This requires the use of the align attribute with the value set as char.

colspan: specifies the number of columns this cell should occupy as a number.

rowspan: specifies the number of rows this cell should occupy as a number.

valign: specifies the vertical alignment of cell content. Possible values are baseline, bottom, middle, and top.

Example:

see <tbody> element

<textarea>

The <textarea> element allows multiple lines of text input. It provides for unlimited amount of text. Mostly used in forms.

Required attributes:

cols: A number that specifies the number of columns that are visible in the text area.

rows: A number that specifies the number of rows that are visible in the text area.

Optional attributes:

disabled: makes the text area control disabled and unresponsive to the user.

name: specifies the unique name for the text area control.

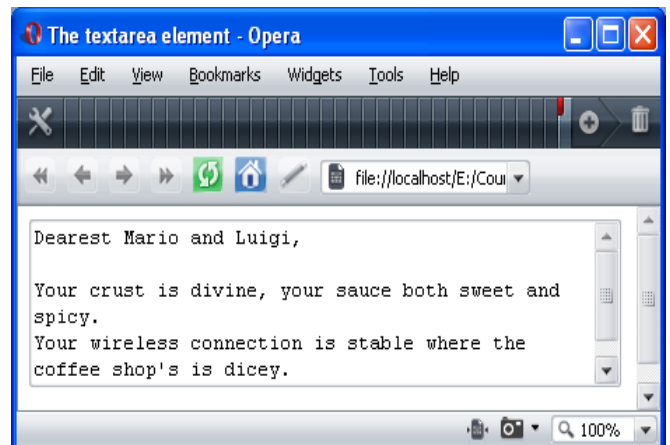
readonly: sets the text area control so that the user cannot change its content.

Example:

```
<body>
<form method="post" action="/cgi-bin/formhandler.cgi">
<textarea name="message" rows="6" cols="50">
  Dearest Mario and Luigi,

  Your crust is divine, your sauce both sweet and
  spicy.

  Your wireless connection is stable where the
  coffee shop's is dicey.
</textarea>
</form>
</body>
```



<tfoot>

The <tfoot> element is contained within a table element and specifies a table footer. The <thead>, <tbody>, and <tfoot> elements allow you to group rows within a table easily.

Required attributes: None

Optional attributes:

`align`: specifies the alignment of cell content. Possible values are `center`, `char`, `justify`, `left`, and `right`.

`char`: specifies which character the text should be aligned on. This requires the use of the `align` attribute with the value set as `char`.

`charoff`: specifies in pixels or as a percentage how far the alignment should be adjusted to the first character to align on. This requires the use of the `align` attribute with the value set as `char`.

`valign`: specifies the vertical alignment of cell content. Possible values are `baseline`, `bottom`, `middle`, and `top`.

Example:

see <tbody> element

<th>

The <th> element can be used in place of the <td> element to mark a row as a header. A header row is normally bolded and centered within a cell.

Required attributes: None

Optional attributes:

abbr: specifies a shortened version of the content in a cell in text.

align: specifies the alignment of cell content. Possible values are *center*, *char*, *justify*, *left*, and *right*.

axis: associates a cell with a conceptual category.

char: specifies which character the text should be aligned on. This requires the use of the `align` attribute with the value set as `char`.

charoff: specifies in pixels or as a percentage how far the alignment should be adjusted to the first character to align on. This requires the use of the `align` attribute with the value set as `char`.

colspan: specifies the number of columns this cell should occupy as a number.

rowspan: specifies the number of rows this cell should occupy as a number.

valign: specifies the vertical alignment of cell content. Possible values are *baseline*, *bottom*, *middle*, and *top*.

Example:

see <tbody> element

<thead>

The <thead> element is contained within a table element and specifies a table header. The <thead>, <tbody>, and <tfoot> elements allow you to group rows within a table easily.

Required attributes: None

Optional attributes:

`align`: specifies the alignment of cell content. Possible values are `center`, `char`, `justify`, `left`, and `right`.

`char`: specifies which character the text should be aligned on. This requires the use of the `align` attribute with the value set as `char`.

`charoff`: specifies in pixels or as a percentage how far the alignment should be adjusted to the first character to align on. This requires the use of the `align` attribute with the value set as `char`.

`valign`: specifies the vertical alignment of cell content. Possible values are `baseline`, `bottom`, `middle`, and `top`.

Example:

see <tbody> element

<title>

The <title> element allows you to provide a title to your document. Browsers typically display this value in the title bar of the browser. This element must appear within the <head> element.

Required attributes: None

Optional attributes: None

Example:

```
<head>
  <title>The textarea element</title>
</head>
```

<tr>

The <tr> element describes a row in a table.

Required attributes: None

Optional attributes:

align: specifies the alignment of cell content. Possible values are *center*, *char*, *justify*, *left*, and *right*.

char: specifies which character the text should be aligned on. This requires the use of the `align` attribute with the value set as `char`.

charoff: specifies in pixels or as a percentage how far the alignment should be adjusted to the first character to align on. This requires the use of the `align` attribute with the value set as `char`.

valign: specifies the vertical alignment of cell content. Possible values are *baseline*, *bottom*, *middle*, and *top*.

Example:

see <tbody> element

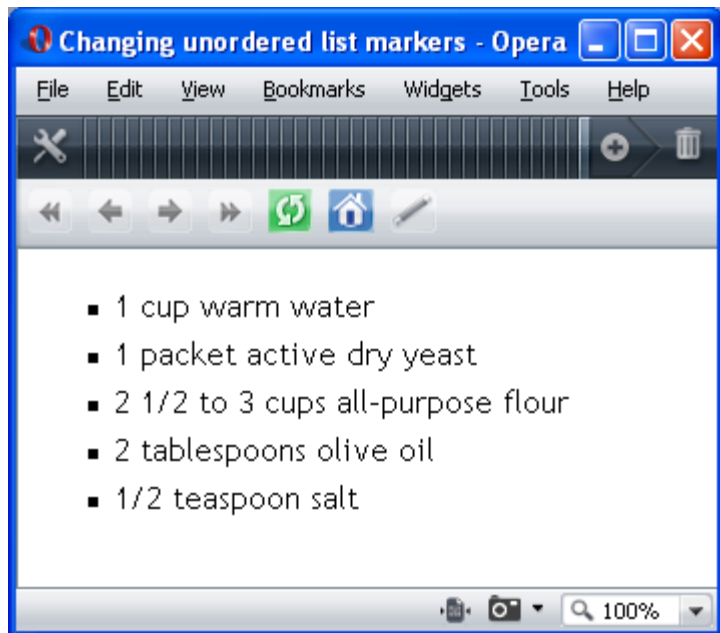
The element is used to specify content as an unordered list. An unordered list must be in the body of the document.

Required attributes: None

Optional attributes: None

Example:

```
<body>
  <ul>
    <li>1 cup warm water</li>
    <li>1 packet active dry yeast</li>
    <li>2 1/2 to 3 cups all-purpose flour</li>
    <li>2 tablespoons olive oil</li>
    <li>1/2 teaspoon salt</li>
  </ul>
</body>
```



<var>

The <var> element is used to define a variable within text.

Required attributes: None

Optional attributes: None

Example:

see code, kbd, and samp

NOTE: The following elements are legal in XHTML Strict, however, they are style elements and as such using CSS is the preferred method instead of using these elements in XHTML.

 - for rendering bold text.

<big> - for rendering larger than normal text.

<i> - for rendering italic text.

<small> - for rendering smaller than normal text.

<tt> - for rendering monospaced text.

