

Introduction to HTML and XHTML

HTML (HyperText Markup Language):

- **Hypertext Markup Language (HTML)** is the standard markup language for creating web pages and web applications.
 - With Cascading Style Sheets (CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web..
 - Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages.
 - HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.
 - HTML elements are the building blocks of HTML pages.
- HTML elements are represented by tags.
- Browsers do not display the HTML tags, but use them to render the content of the page.
- Hypertext Markup Language revision 5 (HTML5) is markup language for the structure and presentation of World Wide Web contents.
- HTML5 supports the traditional HTML and XHTML-style syntax and other new features in its markup. New APIs, XHTML and error handling.

Example: Basic syntax

```

<!DOCTYPE html>
<html>
  <head>
    <title>Basic structure of HTML</title>
  </head>
  <body>

    <h1>First Heading</h1>
    <p>This tag is used for paragraph.</p>

  </body>
</html>

```

- The `<!DOCTYPE html>` declaration defines this document to be HTML5
- The `<html>` element is the root element of an HTML page
- The `<head>` element contains meta information about the document
- The `<title>` element specifies a title for the document
- The `<body>` element contains the visible page content
- The `<h1>` element defines a large heading
- The `<p>` element defines a paragraph

HTML Tags:

- HTML tags are element names surrounded by angle brackets.
- HTML tags normally come in pairs like `<p>` and `</p>`.

- The first tag in a pair is the **start tag**, the second tag is the **end tag**.
- The end tag is written like the start tag, but with a forward slash inserted before the tag name.

<tagName> Content here.....</tagName>

The start tag is also called opening tag , and the end tag is also called closing tag.

XHTML

- Extensible Hypertext Markup Language (**XHTML**) is part of the family of XML markup languages.
- It mirrors or extends versions of the widely used Hypertext Markup Language (HTML), the language in which Web pages are formulated.
- XHTML is almost identical to HTML.
- XHTML is HTML defined as an XML application.
- XHTML is supported by all major browsers.

The Most Important Differences from HTML:

Document Structure

- XHTML DOCTYPE is **mandatory**
- The xmlns attribute in <html> is **mandatory**
- <html>, <head>, <title>, and <body> are **mandatory**

XHTML Elements

- XHTML elements must be properly nested
- XHTML elements must always be closed
- XHTML elements must be in lowercase
- XHTML documents must have one root element

XHTML Attributes

- Attribute names must be in **lower case**
- Attribute values must be **quoted**
- Attribute minimization is **forbidden**

Standard HTML Document Structure

- The first line of every HTML document is a DOCTYPE command , which specifies the particular SGML document -type definition (DTD) with which document complies.
- SGML is a system for defining markup languages. The DTD defines the syntax of markup constructs
- An HTML document must include the four tags <html> , <head> , <title> , and <body> .

- The `<html>` tag identifies the root element of the document ,
- So, HTML documents always have an `<html>` tag following the DOCTYPE command and they always end with the closing html tag, `</html>`.
- The `html` element includes an attribute , `lang` , which specifies the language in which the document is written. (`<html lang="en">` means English language)
- The `head` element is the first element to appear after the opening `html` tag.
- In the document `head` we place things like the page `title` and `meta` data, we add JavaScript to our page with the `script` tag, and we `link` to external stylesheets and other resources.
- All of the content that is visible on a web page is nested between opening and closing `body` tags. The body is the primary container of the content that makes up a web page.
- To display an HTML page correctly, a web browser must know the character set used in the page.
- The most popular international character set used for the web is the 8-bit Unicode Transformation Format (UTF-8) .
- This character set uses from one to six bytes to represent a character , but is backward compatible with the ASCII character set.
- The meta tags specifies the character set used to write the document. The following is the necessary meta element:
`(<meta charset = "utf-8" />)`

Basic Text Formatting

- HTML also defines some special elements for defining text with in the document with special meaning.
- ``- element defines **bold** text, without any extra importance.
- ``- element defines **strong** text, with added semantic "strong" importance.
- `<i>`- element defines *italic* text, without any extra importance.
- ``- element defines emphasized text, with added semantic importance.
- `<mark>`- element defines marked or highlighted text.
- `<small>`- element defines smaller text .
- ``- element defines deleted (removed) text.
- `<ins>`- element defines inserted (added) text.
- `<sub>`- element defines _{subscripted} text.
- `<sup>`- element defines ^{superscripted} text.

Example:

```

<!DOCTYPE html> <html>
<body>

```

```

<b>This text is bold</b>
<strong>This text is strong</strong>
<i>This text is italic</i>
<em>This text is emphasized</em>
<h2>HTML <small>Small</small> Formatting</h2> <h2>HTML
<mark>Marked</mark> Formatting</h2> <p>My favorite color is
<del>blue</del> red.</p> <p>My favorite <ins>color</ins> is
red.</p> <p>This is <sub>subscripted</sub> text.</p> <p>This is
<sup>superscripted</sup> text.</p>

</body>
</html>

```

Basic Text Markup

- *Paragraphs* – Text is normally organized into paragraphs in the body of a document. `<p>` elements defines paragraph.
- *Line Breaks* - Sometimes we need text explicit line break without the preceding blank line. `
` element defines line breaks. The slash indicates that the tag is both an opening and closing tag. The space before slash represents the absent content.
- *Preserving White Space* – Sometimes it is desirable to preserve the white space in text - .i.e. to prevent the browser from eliminating multiple spaces and ignoring embedded line breaks. `<pre>` tag is used `</pre>`
- *Headings* - There are seven level of headings (`h1`, `h2`, `h3`, `h4`, `h5`, & `h6`) that are defined by HTML. Where `<h1>` specifies highest – level heading and `<h6>` specifies lowest heading. The heading tags always break the current line , so their content always appears on a new line.
- *Block Quotations* – Sometimes we want a block of text to be set off from the normal flow of text in a document . The HTML `<q>` element defines a short quotation . The HTML `<blockquote>` element defines a section that is quoted from another source.
- *Horizontal Rules* - Two parts of a document can be separated from each other by placing a horizontal line between them, such lines are called horizontal rules.
- *Abbreviations* - `<abbr>` element defines an abbreviation or an acronym.
- *Address* - `<address>` element defines contact information (author/owner) of a document or an article.

Example:

```

<!DOCTYPE html>
<html>

```

```
<body>
<p>
    WWF's goal is to: <q>Build a future where people live in harmony with
    nature.</q>
</p>
<blockquote>
    For 50 years, WWF has been protecting the future of nature .The world's
    leading conservation organization, WWF works in 100 countries and is
    supported by 1.2 million members in the United States and close to 5
    million globally.
</blockquote>
<p>The <abbr title="World Health Organization">WHO</abbr> was
    founded in 1948.</p>
<address>
    Written by John Doe.<br>
    Visit us at:<br>
    Example.com<br>
    Box 564, Disneyland<br>
    USA
</address>
</body>
</html>
```

Images

- The inclusion of images in a document can dramatically enhance its appearance , although images slow the document-download process.
- The file in which the image is stored is specified in a tag.
- There are the two most common methods of representing images are the Graphic Inter-change Format(GIF ,pronounced like the first syllable of jiffy) and the Joint Photographic Experts Group(JPEG) formats.
- Files in both formats are compressed to reduce storage needs and allow faster transfer over the Internet.
- Files containing GIF images use the .gif (or .GIF) extension on their names.
- The JPEG image file use .jpg (or .jpeg or JPG) extension on their names.
- The image element , whose tag is , is an inline element that specifies an image that is to appear in a document .

```

```

Or

```

```

Where **src** is an attribute that identifies the location/path (also called web address) of the source. The **alt** attribute provides an alternate text for an image, if the user for some reason cannot view it.

Hypertext links

- A hypertext link in an HTML document , which we simply call a *link* here , acts as a pointer to some particular place in some Web resource.
- That resources can be HTML document anywhere on the Web , or it may be the document currently being displayed .
- Most of the web sites consist of many different documents , all logically linked together.
- A link that points to a different resource specifies the address of that resource.
- Links are specified in an attribute of an anchor element , a, which is an inline element.

```
<a href="https://www.google.com">Visit our HTML tutorial</a>
```
- Where (*Visit our HTML tutorial*) link text is the visible part on the browser. Clicking on the link text sending you to specified address.
- **Local links** - A local link (link to the same web site) is specified with a relative URL (without https://www....).

```
<p><a href="html_images.asp">HTML Images</a> </p>
```
- **HTML Links - The target Attribute** - The target attribute specifies where to open the linked document.

The target attribute can have one of the following values:

- blank - Opens the linked document in a new window or tab
- _self - Opens the linked document in the same window/tab as it was clicked (this is default)
- _parent - Opens the linked document in the parent frame
- _top - Opens the linked document in the full body of the window
- framename - Opens the linked document in a named frame

```
<a href="https://www.google.com/" target="_blank">Visit!</a>
```

- **Image as Link** - Images are commonly used as link.

```
<a href="https://www.facebook.com">
```

```
</a>
```

- **Link Titles** - The title attribute specifies extra information about an element. The information is most often shown as a tooltip text when the mouse moves over the element.

```
<a href="https://www.gmail.com" title="Go to gmail">visit for mail to  
gmail</a>
```

- **HTML Link Colors** - By default, a link will appear like this (in all browsers):

- An unvisited link is underlined and blue
- A visited link is underlined and purple
- An active link is underlined and red

You can change the default colors, by using CSS:

```
<!DOCTYPE html>  
<html>  
<head>  
    <style>  
        a.link {  
            color: green;  
            background-color: transparent;  
            text-decoration: none;  
        }  
        a:visited {  
            color: pink;  
            background-color: transparent;  
            text-decoration: none;  
        }  
        a:hover {  
            color: red;  
            background-color: transparent;  
            text-decoration: underline;  
        }  
        a:active {  
            color: yellow;  
            background-color: transparent;  
            text-decoration: underline;  
        }  
    </style>  
    <title>Color links</title>  
</head>  
<body>  
    <h2>Link Colors</h2>  
    <p>You can change the default colors of links</p>  
    <a href="html_images.asp" target="_blank">HTML Images</a>
```

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

Lists

- There are mainly three sorts of html lists.(Unordered Lists, Ordered List and Definition Lists).

1. Unordered Lists:

- The `` tag , which is a block tag , creates an unordered list.
- Each item in a list is specified with an `` tag.
- Any tag can appear in list item , including nested lists with in the `` tag.
- Items are displayed , each list item is implicitly preceded by a bullet.
- Also we can change type of Unordered list .(.i.e. `type=square or circle or disc or none`)

2. Ordered Lists:

- The `` tag , which is a block tag , creates an ordered list.
- Each item in a list is specified with an `` tag.
- Any tag can appear in list item , including nested lists with in the `` tag.
- Items are displayed , each list item is implicitly preceded by a numbers(1, 2, 3....).
- Also we can change type of Unordered list .(.i.e. `type= i / a/A/I`)

3. Definition Lists:

- Definition lists are used to specify lists of terms and their definitions.
- A definition list is given as the content of a `dl` element , which is a block element.
- Each term to be defined in the definition list is given as the content of a `dt` element.
- Definitions themselves are specified as the content of `dd` elements .

Example: Unordered lists , Ordered lists and Definition lists.

```
<!DOCTYPE html>

<html>
  <head>
    <title>About Lists</title>
  </head>
  <body>
    <h2>Unordered List with Bullets(as default)</h2>
```

<p>You can change type as needed. Unordered types (square, circle, disc and none)</p>

```
<ul style="list-style-type:disc">
    <li>Coffee</li>
    <li>Tea</li>
    <li>Milk</li>
</ul>
```

circle
square
none

<h2>Ordered List with Numbers(as default)</h2>

< p > You can change type as needed. </ p >

```
<ol type="">
    <li>Coffee</li>
    <li>Tea</li>
    <li>Milk</li>
</ol>
```

<h3>Definition Lists</h3>

just simple order \ll

```
<dt>122</dt>
<dd>C Programming language</dd>
<dt>122</dt>
<dd>C Programming language</dd>
```

</body>

</html>

Tables

- ☐ Tables provide a highly effective way of presenting many kinds of information.
 - ☐ A table is a matrix of cells.
 - ☐ The cells in the top row often contain column labels , those in the leftmost column often contain row labels , and most of the rest of the cells contain the data of the table.
 - ☐ The content of a cell can be almost any document element , including text, a heading , a horizontal rule , an image , or a nested table.

- An HTML table is defined with the `<table>` tag.
- Each table row is defined with the `<tr>` tag.
- A table header is defined with the `<th>` tag.
- By default, table headings are bold and centered.
- A table data/cell is defined with the `<td>` tag.

Example: `<!DOCTYPE html> <html>`

```

<head>
    <title>Creating table</title> </head>
<body>
    <h2>Basic HTML Table</h2> <table style="width:100%">
        <tr>
            <th>Firstname</th> alignment = "center" border = 1px
            <th>Lastname</th> cellspadding = 15px
            <th>Age</th> colspan = 2
            <td colspan = 2>
                <td>Jill</td> tr / th
                <td>Smith</td>
                <td>50</td>
                <td>Go</td>
                <td>Lifas</td>
            </tr>
            <tr>
                <td>Eve</td>
                <td>Jackson</td>
                <td>94</td>
                <td>10</td>
                <td>14</td>
            </tr>
            <tr>
                <td>John</td>
                <td>Doe</td>
                <td>80</td>
                <td>90</td>
                <td>16</td>
            </tr>
        </table>
    </body>
</html>
```

Add following properties too:

```

<style>
table, th, td {
    border: 1px solid black;
    border-collapse: collapse;
}
```

Diagram illustrating the structure of the table defined in the code. The table has 3 rows and 5 columns. The first row contains three `th` cells and two `td` cells. The second row contains two `td` cells. The third row contains five `td` cells. The `th` cells are labeled 'Firstname', 'Lastname', and 'Age'. The `td` cells are labeled 'Jill', 'Smith', '50', 'Go', 'Lifas', 'Eve', 'Jackson', '94', '10', '14', 'John', 'Doe', '80', '90', and '16'. The `td` cells in the first row have a colspan of 2, merging them into a single column. The `th` cells in the first row have a text-align of left, and the first `td` cell has a colspan of 2. The `td` cells in the first row have a width of 100% and a border of 1px. The `td` cells in the second row have a width of 100% and a border of 1px. The `td` cells in the third row have a width of 100% and a border of 1px.

Firstname	Lastname	Age	Jill	Smith
John	Doe	80	90	16
Eve	Jackson	94	10	14

```

th, td {
    padding: 15px;
}
th {
    text-align: left;
}
</style>

```

- Use the CSS border-spacing property to set the spacing between cells
- Use the colspan attribute to make a cell span many columns
- Use the rowspan attribute to make a cell span many rows
- Use the id attribute to uniquely define one table

Frames

- The <frame> tag is not supported in HTML5.
- The <frame> tag defines one particular window (frame) within a <frameset>.
- Each <frame> in a <frameset> can have different attributes, such as border, scrolling, the ability to resize, etc.
- Instead of using <frame> tag in HTML5 we uses <iframe> tag.
- <iframe> tag is used to define inline frame. Example:

```

<!DOCTYPE html> <html>

    <head> <title>How to define iframe</title> </head> <body>

        <h2>Iframe - Target for a Link</h2>

        <iframe height="300px" width="100%" src="frame.htm"
        name="iframe_1"></iframe>

        <p><a href="https://www.w3schools.com" target="iframe_1">
        W3Schools.com </a> </p>

        <p>When the target of a link matches the name of an iframe, the link will
        open in the iframe. </p>
    </body>
</html>

```

frameset cols="25%, 75%.">
 <frame src="frame1.htm" />
 <frame src="frame2.htm" />

<frameset rows="33.3%, 66.7%">
 <frame src="top.htm" name="top" />
 <frame src="left.htm" />
 <frame src="content.htm" />

<frameset border="1" rows="50%, 50%">
 <frame src="top.htm" />
 <frame src="bottom.htm" />

<frameset border="1" cols="33.33333333333333%, 33.33333333333333%, 33.33333333333333%">
 <frame src="left.htm" />
 <frame src="center.htm" />
 <frame src="right.htm" />

</frameset> Forms

- Form is used to communicate information from a web browser to the server.

name:

`<input type="text" name="extname" maxLength="8"
placeholder="Max 8 character" value="Rimel" />`

- Also form is a window or screen i.e. used to gathering information(data) and also visualized with different , numerous field or spaces to enter data.
- Generally , form is used to enter data.
- In HTML5 form is defined within the `<form>` tag and having different control attributes.

Example:

`<html>`

`<head> <title>Designing Form</title> </head>`

`name
"textarea"`

`"checkbox"`

`"input type="text"`

`"password"`

`"number"`

`"date"`

`"range"`

`"file"`

`"radio"`

`"Reset"`

`dropdownList" = "submit"`

`= color`

`<select> option value="`

`"cricket" cricket`

`option value="football" football`

`football`

`<input type="submit" value="Submit" />`

`</form>`

`<p>If you click the "Submit" button, the form-data will be sent to a page called "firstPage.php".</p>`

`<select>`

`<select name="sports" size="3" multiple>`

Note: `<input type="submit">` defines the button that is used to submitting the form data to form handler. The form handler is typically a server page with script for processing input data. The form handler is specified in the form's action attributes.

Note: The page that response when button is clicked is `firstPage.php` i.e. given below.

`<!-- This is firstPage.php file -->`

`<!DOCTYPE html>`

`<html>`

`<head>`

```

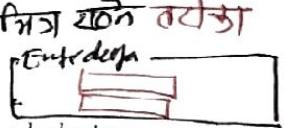
<title>Response on clicking submit button</title>
</head>
<body>
    Country Name:<?php echo $_POST["countryName"]; ?><br>
    <hr>      <!-- define horizontal ruler -->
    Capital Name:<?php echo $_POST["capitalName"]; ?>
</body>
</html>

```

Attributes that are used to control form:

- action* – is used to define action to be performed when the form is submitted.
- method* – is used to specifies HTTP method(post and get) when submitting the form.
- name* – is used to sent data to all (where needed). Also used to identify the form.
- target* – is used to specifies the target address.

Some elements that are used within the form:

- input element* – is used to take input from user on browser i.e. it provides input field. Input fields are defined by an attribute called *type*(text, checkboxes, radio buttons, passwords, email, reset buttons, submit button, image, buttons, color, etc).
 - fieldset* – is used to group related data in a form. *It has first two as best*
 - legend* – is used to define caption for *<fieldset>* element.
- 

Example: Radio button – let user to chose / select one of a limited choice

```

<form>
    <input type="radio" name="gender" value="male" checked>
    Male<br>
    <input type="radio" name="gender" value="female"> Female <br>
    <input type="radio" name="gender" value="other"> Other </form>

```

Example: Input type password – defines password field

```

<form action="">
    User name:<br>
    <input type="text" name="userName">
    <br>

```

<textarea name='message' rows='10' cols='30'>
<button type='button' onclick='alert('Hello student') click me!</button>

User password:


```
<input type="password" name="psw">
</form>
```

Example: Input type reset – defines reset button that is used to reset form and set all form values to default.

```
<form action="firstPage.php">
Country Name:<br>
<input type="text" name="countryName" required>
<br>
Capital Name:<br>
<input type="text" name="capitalName" >
<br><br>
<input type="submit" value="Submit">
<input type="reset">
</form>
```

Example: Input type checkbox – defines checkbox let a user select ZERO or MORE options of a limited number of choices.

```
<form action="secondPage.php" method="POST">
<input type="checkbox" name="nation[]" value="Nepal">My county
name is Nepal.
<br>
<input type="checkbox" name="nation[]" value="Kathmandu"> Our
capital is Kathmandu.
<br><br>
<input type="submit" value="Submit">
</form>
```

Note: <input type="submit"> defines the button that is used to submitting the form data to form handler. The form handler is typically a server page with script for processing input data. The form handler is specified in the form's action attributes.

Note: The page that response when button is clicked is secondPage.php i.e. given below.

```
<?php
```

```

if (isset($_POST["submit"])) {
    if (!empty($_POST["nation"])) {
        echo '<h3> you have selected following vehicle </h3>';
        foreach ($_POST["nation"] as $nation) {
            echo '<p>' . $nation . '</p>';
        }
    } else {
        echo "<br>Please select at least one";
    }
}
?>

```

Multimedia in HTML

- Multimedia is content that uses the combination of different contents form such as audio , video , text, animation , graphic etc .
- Also different contents form have different formats.
- Almost anything that can be heard or seen called multimedia.
- Different types and formats of multimedia forms are often used by web page.

The Video Element

- There are 3 supported video format in HTML 5: MP4 , WebM , and Ogg.
- The browser support for the different formats is:

Browser	MP4	WebM	Ogg
Internet Explorer	Yes	No	No
Chrome	Yes	Yes	Yes
FireFox	Yes	Yes	Yes
Safari	Yes	No	No
Opera	Yes	Yes	Yes

- The HTML5 <video>element specifies a standard way to embed a video in a web page.
- HTML video-media types

File Format	Media Types
Mp4	video/mp4
Ogg	video /ogg
WebM	video /webm

Example:

```
<!DOCTYPE html>
<html>
<body>

<video width="320" height="240" controls>
  <source src="movie.mp4" type="video/mp4">
  <source src="movie.ogg" type="video/ogg">
  Your browser does not support the video tag.
</video>
</body>
</html>
```

How it works:

- The *controls* attribute adds video controls, like play, pause, and volume.
- It is good idea to include *height* and *width* attributes . If height and width are not set, the page might flicker when video loads.
- The *<source>* element allows you to specify alternative video files which the browser may choose from. The browser will use the first recognized format.
- The text between the *<video>* and *</video>* tags will only be displayed in browsers that do not support the *<video>* element.
- HTML5 defines DOM methods, properties, and events for the *<video>* element.

The audio Element

- The HTML5 *<audio>* element specifies a standard way to embed audio in a web page.
- There are 3 supported audio format in HTML5 : MP3, OGG, WAV
- The browser support for the different formats is:

Browser	MP3	WAV	Ogg
Internet Explorer	Yes	No	No
Chrome	Yes	Yes	Yes
FireFox	Yes	Yes	Yes
Safari	Yes	Yes	No
Opera	Yes	Yes	Yes

- HTML audio-media types

File Format	Media Types
Mp3	audio/mpeg
Ogg	audio/ogg
Wav	audio/wav

Example:

```
<!DOCTYPE html>
<html>
    <title> Audio Element</title>
    <body>
        <audio controls autoplay>
            <source src="horse.ogg" type="audio/ogg">
            <source src="horse.mp3" type="audio/mpeg">
                Your browser does not support the audio element.
        </audio>
    </body>
</html>
```

How it works:

- The *controls* attribute adds audio controls, like play, pause, and volume.
- The *<source>*element allows you to specify alternative audio files which the browser may choose from. The browser will use the first recognized format.

Some attributes/methods/properties used in audio/video element:

There are many attributes/properties that are used in audio/video element . Some of them are given below:

- *autoplay* – it start playing audio/video as soon as it is loaded.
- *audioTracks* – returns an Audio TrackList object representing available audio tracks.
- *buffered* – returns a TimeRanges object representing the buffered parts of the audio/video.
- *currentSrc* – returns the URL of the current audio/video.
- *currentTime* – sets or returns the current playback position in the audio/video(in seconds).
- *duration* – returns the length of the current audio/video(in seconds).
- *paused* – returns whether the audio/video is paused or not.
- *muted* – sets or returns whether the audio/video is muted or not.

- play(event)* – fires when the audio/video has been started or is no longer paused.
- loop* – sets or returns whether the audio/video should start over again when finished.