

XHTML

- XHTML stands for **Extensible HyperText Markup Language**. It is a cross between HTML and XML language.
- XHTML is almost identical to HTML but it is stricter than HTML. XHTML is HTML defined as an XML application. It is supported by all major browsers.
- Although XHTML is almost the same as HTML but It is more important to create your code correctly, because XHTML is stricter than HTML in syntax and case sensitivity. XHTML documents are well-formed and parsed using standard XML parsers, unlike HTML, which requires a lenient HTML-specific parser.

Why use XHTML

XHTML was developed to make HTML more extensible and increase interoperability with other data formats. There are two main reasons behind the creation of XHTML:

- It creates a stricter standard for making web pages, reducing incompatibilities between browsers. So it is compatible for all major browsers.
- It creates a standard that can be used on a variety of different devices without changes.

Advantages of XHTML

Here are the following advantages of XHTML, such as:

- While using XHTML, the code of web applications becomes more stylish and easy to reuse.
- It can help the developer create more advanced web projects due to the compatibility with various devices, and it also supports self-created markups like SVG (scalable vector graphics).
- XHTML code can easily be converted to PDFs, RSS, and RFT, which allows the developer to work with a vast range of files.
- XHTML reduce the loading time required by the browser to load an event which can result in overall speedy development, thus reducing time and energy
- It contains closing tags which is an advantage for beginners, and this also makes the code look clean and easy to reuse.

Disadvantages of XHTML

XHTML also has some disadvantages, such as:

- Very few browsers use XHTML.
- Case sensitive as every part of code should be in lowercase.
- It is mandatory to write < DOCTYPE > declaration.
- And all the tags must be closed in the necessary order.

Difference between HTML and XHTML

HTML and XHTML are both markup languages used to create web pages and applications. HTML and XHTML have some key differences that set them apart. Here are the following major differences between HTML and XHTML:

S.No.	HTML	XHTML
1.	Hypertext mark-up language - - > HTML	Extensible Hypertext Mark-up Language - - > XHTML.
2.	Tim Berners created in 1991	World wide web consortium or W3C created in 2000
4.	It is an extension of standard generalized markup language or SGML	It is a combination of extensible markup language XML and hypertext markup language HTML
5.	It stored in a document file format	It stored as a markup language format
6.	It is not case sensitive as there is no mandatory rule to write the entire mark up in uppercase or lower case. It can also be a combination of both.	It is case-sensitive, and every tag and attribute used inside must be in lowercase.
7.	It is not mandatory to add document label < DOCTYPE >at the top of every page. We can even skip it.	It is mandatory to add a document label < DOCTYPE > at the beginning of the page.
8.	We can close any tag anytime and anywhere as per our needs	It is mandatory to close all the tags in strict residing order as they were declared.
9.	We can add attributes without any quotes.	It is mandatory to add quotes on every attribute we declare
10.	.html and .htm are the extensions used by HTML	.xhtml, .xml and .xht are the file extensions used by XHTML

11 Low structure is used

It contains a very strict structure, and the developer cannot go out of the bounds of these structures.

XHTML Syntax

XHTML syntax is very similar to HTML syntax and all the valid HTML elements are also valid in XHTML. But XHTML is case sensitive so you have to pay a bit extra attention while writing an XHTML document to make your HTML document compliant to XHTML.

You must remember the following important points while writing a new XHTML document or converting existing HTML document into XHTML document:

- All documents must have a DOCTYPE.
- All tags must be in lower case.
- All documents must be properly formed.
- All tags must be closed.
- All attributes must be added properly.
- The name attribute has changed.
- Attributes cannot be shortened.
- All tags must be properly nested.

DOCTYPE Declaration

All XHTML documents must contain a DOCTYPE declaration at the start. There are three types of DOCTYPE declarations:

- Strict
- Transitional
- Frameset

Here is an example of using DOCTYPE.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
```

Tags must be in lower case

XHTML is case-sensitive markup language. So, all the XHTML tags and attributes must be written in lower case.

```
<!-- Invalid in XHTML -->
```

```
<A Href="/xhtml/xhtml_tutorial.html">XHTML Tutorial</A>
```

```
<!-- Valid in XHTML -->
```

```
<a href="/xhtml/xhtml_tutorial.html">XHTML Tutorial</a>
```

Closing Tags are mandatory

An XHTML must have an equivalent closing tag. Even empty elements should also have closing tags. Let's see an example:

```
<!-- Invalid in XHTML -->
```

```
<p>This paragraph is not written according to XHTML syntax.
```

```
<!-- Invalid in XHTML -->
```

```

```

```
<!-- Valid in XHTML -->
```

```
<p>This paragraph is not written according to XHTML syntax.</p>
```

```
<!-- Valid in XHTML-->
```

```

```

Attribute Quotes

All the XHTML attribute's values must be quoted. Otherwise, your XHTML document is assumed as an invalid document.

See this example:

1. <!-- Invalid in XHTML -->
2.
3. <!-- Valid in XHTML -->
4.

Attribute Minimization

XHTML doesn't allow you to minimize attributes. You have to explicitly state the attribute and its value.

See this example:

1. <!--Invalid in XHTML -->
2. <option selected>

3. <!-- valid in XHTML-->
4. <option selected="selected">

A list of minimized attributes in HTML and the way you need to write them in XHTML.

HTML Style	XHTML Style
compact	compact="compact"
checked	checked="checked"
declare	declare="declare"
readonly	readonly="readonly"
disabled	disabled="disabled"
selected	selected="selected"
defer	defer="defer"
ismap	ismap="ismap"
nohref	nohref="nohref"
noshade	noshade="noshade"
nowrap	nowrap="nowrap"
multiple	multiple="multiple"

noresize noresize="noresize"

The id Attribute

The id attribute is used to replace the name attribute. Instead of using name = "name", XHTML prefers to use id = "id".

See this example:

1. <!-- Invalid in XHTML -->
2.
3. <!-- Valid in XHTML -->
4.

The language attribute

In XHTML, the language attribute of script tag is deprecated so you have to use type attribute instead of this.

See this example:

1. <!-- Invalid in XHTML -->
2. <script language="JavaScript" type="text/JavaScript">
3. document.write("Hello XHTML!");
4. </script>
5. <!-- Valid in XHTML -->
6. <script type="text/JavaScript">
7. document.write("Hello XHTML!");
8. </script>

XHTML Events

When you visit a website, you do things like click on text, images and hyperlinks, hover-over things, etc. These are examples of what JavaScript calls events.

We can write our event handlers in JavaScript or VBScript and can specify these event handlers as a value of event tag attribute. The XHTML 1.0 has a similar set of events which is available in HTML 4.01 specification.

The <body> and <frameset> Level Events

There are only two attributes which are used to trigger any JavaScript or VBScript code, when any event occurs at document level.

Attribute	Value	Description
Onload	Script	Script runs when a XHTML document loads.
onunload	Script	Script runs when a XHTML document unloads.

The <form> Level Events

There are six attributes which are triggered when any event occurs at form level.

Attribute	Value	Description
onchange	Script	It is executed when the element changes.
onsubmit	Script	It is executed when the form is submitted.
onreset	Script	It is executed when the form is reset.
onselect	Script	It is executed when the element is selected.
onblur	Script	It is executed when the element loses focus.
onfocus	Script	It is executed when the element gets focus.

Keyboard Events

There are three events which are generated by keyboard. The keyboard events are not valid in base, bdo, br, frame, frameset, head, html, iframe, meta, param, script, style, and title elements.

Attribute	Value	Description
Onkeydown	Script	This is executed when the user press the keyboard button.
Onkeypress	Script	This is executed when the user press and release the keyboard button.
onkeyup	Script	This is executed when the user release the keyboard button.

Mouse Events

There are some mouse generated events which executes when it comes in contact with any HTML tag. These events are not valid in base, bdo, br, frame, frameset, head, html, iframe, meta, param, script, style, and title elements.

Attribute	Value	Description
OnClick	Script	It is executed on a mouse click.
Ondblclick	Script	It is executed on a mouse double-click.
Onmousedown	Script	It is executed when mouse button is pressed.
Onmousemove	Script	It is executed when mouse pointer moves.
onmouseout	Script	It is executed when mouse pointer moves out of an element.
onmouseover	Script	It is executed when mouse pointer moves over an element.

onmouseup Script It is executed when mouse button is released.

XHTML Doctypes

There are three types of Document Type Definitions (DTDs). The easiest and most commonly used is the XHTML Transitional document.

A list of the XHTML Doctypes:

- Strict
- Transitional
- Frameset

You should be very careful while writing XHTML document because the few XHTML elements and attributes, which are available in one DTD but not available in another DTD. So, you should select your XHTML elements or attribute carefully.

XHTML 1.0 Strict DTD

It is recommended to use when you want to use Cascading Style Sheet (CSS) strictly and avoiding to write most of the XHTML attributes.

Add the following DTD at the top of your XHTML document.

Syntax:

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

XHTML 1.0 Transitional DTD

It is recommended to use when you want to use many XHTML attributes as well as few Cascading Style Sheet (CSS) properties.

Add the following DTD at the top of your XHTML document.

Syntax:

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

XHTML 1.0 Frameset DTD

It is recommended to use when you want to use HTML Frames to partition the browser window into two or more frames.

Add the following DTD at the top of your XHTML document.

Syntax:

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

XHTML Attributes

There are some XHTML attributes which are standard and associated to all XHTML tags. The XHTML attributes can be divided in 3 types:

- Core Attributes
- Language Attributes
- Microsoft Proprietary Attributes

Core Attributes

The core attribute is not valid in base, head, html, meta, param, script, style, and title elements.

Attribute	Value	Description
-----------	-------	-------------

Class	class_rule or style_rule	It specifies the class of the element.
Id	id_name	It specifies a unique id for the element.
Style	style_definition	It specifies an inline style definition.
Title	tooltip_text	It specifies a text to display in a mouse tip.

Language Attributes

The lang attribute specifies the language which is used for the enclosed content. It uses the ISO standard language abbreviations like fr for French, en for English, and so on. The language attribute is not valid in base, br, frame, frameset, hr, iframe, param, and script elements.

Attribute	Value	Description
Dir	ltr rtl	It is used to set the text direction.
Lang	language_code	It is used to set the language code.

Microsoft Proprietary Attributes

There are some proprietary attributes introduced by Microsoft with the Internet Explorer 4 and higher versions.

Attribute	Value	Description
accesskey	character	It is used to set a keyboard shortcut to access an element.
language	String	This attribute is specifies the scripting language which is used with the script bound to the element, typically through an event handler attribute. Its possible values might include JavaScript, jScript, VBS, and VBScript.
tabindex	Number	It is used to set the tab order of an element.
contenteditable	Boolean	It allows users to edit content rendered in internet explorer 5.5 or greater. Its possible values are true or false.
disabled	Boolean	The disabled attribute makes the element appear faded and will not respond to user input. Its possible values are true or false.
hidefocus	On or Off	This proprietary attribute, introduced with internet explorer 5.5, hides focus on an element's content. It specifies that focus must be applied to the element using the tabindex attribute.
unselectable	On or Off	It is used to prevent content displayed in internet explorer 5.5 from being selected.

