Chapter-2

## Introduction to HTML & XHTML

*HTML

- **HTML** stands for **Hyper Text Markup Language.**
- Html is introduced by **Tim Berners-Lee** in 1993
- It is used to design web pages and web related documents.
- HTML is the combination of Hypertext and Markup language.
  → Hypertext defines the link between the web pages.
  → A markup language is used to define the text document within **tag** which defines the structure of web pages. <html> <body>
- HTML is a markup language that is used by the browser to manipulate text, images, and other content to display it in the required format.
- HTML explains how a text & objects should appear on the web page.
- The use of **Tags** in HTML helps the Web page in a better design.
- It consists of a series if tags which is of short codes represented within < >.
- It is non-linear as we go to any page on the internet by clicking on the links which provides connectivity between the web pages that is known as **Hyperlinks.**
- It ensures universal compatibility & flexibility.
- The coding is done in notepad and the text file is saved as **.html** extension.
- A Web Browser like google chrome or an internet explorer is required to execute a HTML document.

### Basic format to create a simple web page using HTML.

```
<html>
        <head>                          ( Head section of website)
                        <title>         </title>
          </head>
          <body>        (Body section of website)
          </body>
 </html>
```

### Example of simple HTML page

```
<html>
<head>
        <title>Simple HTML Page</title>
</head>

<body>
```

```
    </body>

</html>
```

Different Versions of HTML

**1. HTML 1.0**

- This is the basic version of HTML has support for basic elements like **text controls and images**. This was the very basic version of HTML with less support for a wide range of HTML elements.
- **DisAdv**- It does not provide support for **tables**, **font** support, etc and also It does not have rich features like **styling** and other web features like **background colours, frames** etc, .

**2. HTML 2.0**

- HTML version 2.0 was developed in 1995 with basic intention of improving HTML version 1.0
- Forms and tables were developed.
- Background colour concept is introduced.
- **DisAdv-** It was expensive.
- Frames were not able to set.
- Forms became available but with limited fields like text boxes, buttons, etc.

**3. HTML 3.2**

- It was developed in 1997. After HTML 2.0 was developed, the next version of HTML was 3.2
- With version 3.2 of HTML, HTML tags were further improved.
- Another important feature what HTML 3.2 implemented was support for **CSS**. CSS stands for **Cascading Style Sheet**. It is CSS that provides features to make HTML tags look better. CSS helps to style HTML elements.
- It also started supporting **frame tags.**
- **DISADV - Expensive**

**4. HTML 4.0**

- It introduced the formatting styles such as **Bold, Italics, Underline.**
- More features of CSS is introduced.
- **Disadv –** It was expensive and Maintenance was tedious.

**5. HTML 4.01**

- It is a minor revision of 4.0 version.
- It was developed in 1999. It extended the support of cascading styling sheets.

- Concept of an external styling sheet emerged due to which external CSS file could be developed, and this external styling file could be included in HTML itself.

**6. XML 1.0**

- **Extensible Mark up Language -** considered as a version of HTML.

- Difference is that **XML** is used to **describe** the data whereas **HTML** is used to **display** the data.XML tags are not predefined..It consists of user created tags.

**7.XHTML –Extensible HTML**

- Is a Crossbreed of HTML & XML.
- It has the features of both HTML & XML
- It is considered as a general purpose mark up language.

**8. HTML 5**

- This is the latest version of HTML.
- Introduced in 2014
- It came up with lots of HTML tags support.
- Like **Email , Password etc..**
- It introduced the property of **interoperability – ie** the property of executing different operations at the same time.
- It reduces the development time.
- It explains how to recover from failures.
- Adding the features of audio, video and graphics in a web page was possible.
- Less Expensive

***XHTML - Features**

- It is a general purpose mark up language.
- **Extensible Hyper Text Markup Language** is the name for which XHTML stands for.
- It is a hybrid technology between HTML and XML that combines the functionalities of both to become powerful and efficient.

- The XHTML was developed by World Wide Web Consortium (W3C), which is an international organization that sets standards for the World Wide Web (WWW). It was designed to help web developers .
- It helps to create a web page in an easy manner because of its précised (accurate)structure.
- It is a compatible software and easy to maintain.
- It is the follow-on version of HTML, which means we can do everything using XHTML that HTML can do.

***Advantages**

1. **Extensibility:** we can define and use our own tags.
2. **Portability:** we can also use it in mobile devices and small devices which contain small processors with less power. Ie it is feasible for desktops

,laptops etc.

3. **Easy to Maintain:** As the rules are clear in XHTML, the margin for errors is less. The structure is more apparent, and problem syntax is easier to spot; therefore, XHTML is easy to author and maintain.
4. **Ready for the future:** The documents will be easily upgraded to the new version to take advantage of new features.
5. **Compatibility:** we can use with all type of systems & OS.

## **Basic Tags in HTML/XHTML

| Tag | Description |
|---|---|
| <!DOCTYPE> | Defines the document type |
| <html> | Defines an HTML document |
| <head> | Contains metadata/information for the document |
| <title> | Defines a title for the document |
| <body> | Defines the document's body |
| <h1> to <h6> | Defines HTML headings |
| <p> | Defines a paragraph |
| <br> | Inserts a single line break |
| <hr> | Defines a thematic change in the content |
| <!--...--> | Defines a comment |

## <!DOCTYPE>

- All XHTML documents must start with a **<!DOCTYPE>** declaration.
- It is an "information" to the browser about what document type to expect.
- The <!DOCTYPE> declaration should mention in capital letters only.

## <html> Tag

- The <html> tag represents the root of an HTML document.
- The <html> tag is the container for all other HTML elements
- It represents the starting of your document.
- All the html tags should have a closing tag**</html>**

## <head> Tag

- The <head> tag in HTML is used to define the head portion of the document which contains information related to the document.
- The <head> tag contains other head elements such as <title>, <meta>, <link>, <style> etc..
- </head>

## <title> Tag

- It defines a title for your HTML document.
- The title must be text-only.
- The <title> tag is required in HTML documents!
- It sets the title in the browser toolbar.
- **Syntax:**

  <title> Title name </title>

## <body> Tag

- The <body> tag defines the document's body.
- That means it is used to define the main content present inside an HTML page.
- It is always enclosed within <html>tag. The <body> tag is the **last child of <html> tag**. A body tag contains starting as well as an ending tag </body>.

- The <body> element contains all the contents of an HTML document, such as **headings, paragraphs, images, hyperlinks, tables, lists, etc**.
- **Syntax:**

  <body> Body Contents... </body>

  ****The Heading Tags**

  <h> <h1> to <h6>
- There are six levels of headings defined by HTML.
- These six heading elements are h1, h2, h3, h4, h5, and h6; with **h1** being the biggest font level and **h6** the smallest.
- Use one <h1> per page - this should represent the main heading/subject for the whole page.
- All the heading tag should be closed

  Eg: <h1>.....  </h1>

```
    <body>
        <h1>This is heading 1</h1>
        <h2>This is heading 2</h2>
        <h3>This is heading 3</h3>
        <h4>This is heading 4</h4>
        <h5>This is heading 5</h5>
        <h6>This is heading 6</h6>
    </body>
</html>
```

This will produce the following result –

**This is heading 1**

**This is heading 2**

**This is heading 3**

**This is heading 4**

**This is heading 5**

**This is heading 6**

#  Tag

- The ** ** tag in HTML defines a paragraph. These have both opening and closing tag. So anything mentioned within ** ** and ** ** is treated as a paragraph.
- **Syntax:**

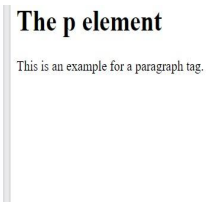
   Content 

```
<html>
<head>
<title>  </title>
<body>

<h1>The p element</h1>

<p>This is an example for a  paragraph tag.</p>


</body>
</html>
```




# **<br> Tag (break rule)

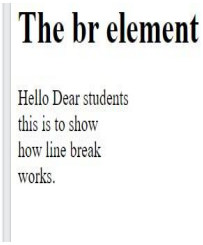
- The <br> tag inserts a **single line break**.
- The <br> tag is useful when we wants to use a newline.
- The <br> tag is an **empty tag** which means that it has no end tag.

```
<html>
<body>

<h1>The br element</h1>

<p>Hello Dear students <br> this is to show <br> how line break <br> works. <br> </p>

</body>
</html>
```



**<hr> Tag - horizontal rule

- The <hr> tag in HTML stands for horizontal rule and is used to insert a horizontal rule or a thematic break in an HTML page to divide or separate document sections.
- The <hr> tag is an **empty tag** and it does not require an end tag.

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

<!-- This is a comment -->
<p>This is a paragraph.</p>

</body>
</html>
```

# *(<!– Comment –>) Tag

•The comment tag (<!– Comment –>) is used to insert comments in the HTML code.

•Comments are non-executable statements in HTML.

•Comments are not displayed in the browsers.

**Syntax:**

<!-- Comments here -->

**Types of HTML Comments:**

- There are two types of comments in HTML they are:
- Single-line comment
- Multi-lines comment

**Single-line comment:**

```
<body>
   <!--   This is valid comment -->
   <p>Document content goes here.....</p>
</body>
```

**Multi-line comment:**

```
<body>
   <!--
      This is a multiline comment and it can
      span through as many as lines you like.
   -->

   <p>Document content goes here.....</p>
</body>
```

# *Hyperlinks

- **Link**
  It is a connection from one web resource to another.
  User can click on a link and jump to another document.
  A link has two ends ,
  *→An **anchor - <a>**
  →**direction or path**

The link starts at the “source” anchor and points to the “destination” anchor.

# *HTML Links - Syntax

• The HTML **<a>** tag defines a hyperlink. • It has the following **syntax**:

• **<a href="*url or the path*"> *link text* </a>**

The most important attribute of the **<a>** element is the **href** attribute(hypertext reference), which indicates the link's destination address or the path.

**Rules for XHTML Documents**

- XHTML - <!DOCTYPE ....> Is Mandatory
- XHTML Elements Must be Properly Nested
- XHTML Elements Must Always be Closed
- XHTML Empty Elements Must Always be Closed
- XHTML Elements Must be in Lowercase
- XHTML Attribute Names Must be in Lowercase
- XHTML Attribute Values Must be Quoted
- XHTML Documents should always be well formed with root elements and sub elements.

**XHTML - <!DOCTYPE ....> Is Mandatory**

- XHTML document must have an XHTML **<!DOCTYPE>** declaration.

- The <html>, <head>, <title>, and <body> elements must also be present.

**XHTML Elements Must be Properly Nested**

- In XHTML, elements must always be properly nested within each other.
- Eg:-

```
<body>
        <bold>
                <italics>Hello All
                </italics>
        <
/bold> </body>
```

**XHTML Elements Must Always be Closed**

- In XHTML, elements must always be closed, like this:
- This is a paragraph
  <u>This is another paragraph</u>

**XHTML Empty Elements Must Always be Closed**

- In XHTML, empty elements must always be closed.
- Eg:
- A **break**: <br />
  A **horizontal rule**: <hr />
  An **image**: <img src="happy.gif“ />

**XHTML Elements Must be in Lowercase**

- In XHTML, element names must always be in lowercase.
- Eg: <body>
  This is a paragraph
  </body>

invalid

- <BODY>
  <P>This is a paragraph</P>
  </BODY>

**XHTML Attribute Names Must be in Lowercase**

- In XHTML, attribute names must always be in lowercase

- Eg: <a **href**=“image.html/">**click** </a>
  <img **src**=“ flower.jpeg”>

**XHTML Attribute Values Must be Quoted**

- In XHTML, attribute values must always be quoted.
- Eg:
- <a href=**“**image.html**"**>**click** </a>

  <img src=**“** flower.jpeg**”**>
- XHTML Documents should always be well formed with root elements and sub elements.
- There should be only one root element and sub tags should be mentioned within the root tag.

## The Most Important Differences from HTML

- <!DOCTYPE> is **mandatory**
- The xmlns attribute in <html> is **mandatory**
- <html>, <head>, <title>, and <body> are **mandatory**
- Elements must always be **properly nested**
- Elements must always be **closed**
- Elements must always be in **lowercase**
- Attribute names must always be in **lowercase**
- Attribute values must always be **quoted**

## Basic structure of an XHTML Document

**<?xml version**="1.0“ encoding= “UTF-8”**?>**

**<!DOCTYPE ……>**

<**html xmlns**……..> **<head>**

**<title>** Title of the Document**</title> </head>**

**<body>** ...your content goes here... **</body> </html>**

**XML VERSION**

- The first line of a XHTML document always starts with the xml version.
- It must include the initial xml declaration.

**XML Declaration syntax**

<?xml version =
"version_numb
er"
encoding =
"encoding_declaration" ?>

***DOCTYPE**

- The second line of the document starts with **DOCTYPE:** It is used to declare a DTD (**Document Type Definition**).
- **DTD** defines the document **structure** with a list of validated elements and attributes. Its purpose is to describe in precise terms , the **language** and the **syntax** allowed in XHTML.

| Parameter | Parameter_value | Parameter_description |
|---|---|---|
| Version | 1.0 | Specifies the version of the XML standard used. |
| Encoding | UTF-8, UTF-16, ISO-10646-UCS-2, ISO-10646-UCS-4, ISO-8859-1 to ISO-8859-9, ISO-2022-JP, Shift_JIS, EUC-JP | It defines the character encoding used in the document. UTF-8 is the default encoding used. |

- **XML** documents contains a set of rules to be followed , these rules are stored in DTD.
-

**html xmlns – xml name space**

- **XML namespaces** helps to make use of wide range of tags available in xml documents.

- They are used for providing unique named elements and attributes in an XML document.
- It will avoid name conflicts.
- The html xmlns is followed by the **<head>,<title> and <body>** where all the tags must be properly nested and closed.

Example

```
<?xml version="1.0" encoding="UTF-8"?>

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

<html xmlns="http://www.w3.org/TR/xhtml1" xml:lang="en" lang="en">
   <head>
      <title>Every document must have a title</title>
   </head>

   <body>
      ...your content goes here...
   </body>
</html>
```

## **DOCTYPES or DTD

- **Document Type Definition** , also known as **DOCTYPE** , is an important code required in every XHTML document.
- The <!**DOCTYPE>** is not an **element or tag**, it lets the browser know about the **version** or **standard** of HTML or any other **markup language** that is being used in the document.
- It is an "information" to the browser about what **document type** to expect.
- It defines the document structure with a legal set of elements & attributes.

**Types of**

| Doc Type | Example |
|---|---|
| XHTML 1.0 **Strict** | If we are planning to write strictly CSS and avoiding to write most of the XHTML attributes, then it is recommended to use this DTD. A document conforming to be this DTD will be of the best quality. |
| XHTML 1.0 **Transitional** | If we are planning to use many XHTML attributes as well as few CSS properties, then we should use adopt this DTD and we should write XHTML document according to DTD. |
| XHTML 1.0 **Frameset** | We can use this when we want to use HTML frames to partition the browser window into two or more frames |

**SYNTACTIC DIFFERENCES BETWEEN HTML AND XHTML**

| XHTML | HTML |
|---|---|
| The tag names and attribute names are case sensitive. | The tag names and attribute names are case insensitive. |
| Every element we use in XHTML must have both an opening and closing tag. **Ex:** hello | Not required in HTML. **Ex** :< p>Hello, this is all we need in HTML no closing tag is required. |
| In XHTML ,empty elements must use either empty element syntax (<br/>) or have an end tag immediately follow the start tag(<br></br>) | It is optional. We can just write as <br>. |
| In XHTML, all tags must be written in lowercase. | In HTML we can use capital letters for elements, lowercase letters for attributes, or Whatever convention we like. |
| All attribute values in XHTML must be quoted. **Ex:**<a href="path"> </a> | Not required. **Ex:**<a href="path"/> </a> |
| In XHTML an opening<html> tag requires an **xmlns** attribute. | It is not required in HTML. |