HTML

HTML is the standard markup language for creating Web pages.

- HTML stands for Hyper Text Markup Language
- HTML is the standard markup language for creating Web pages
- HTML describes the structure of a Web page
- HTML consists of a series of elements
- HTML elements tell the browser how to display the content
- HTML elements label pieces of content such as "this is a heading", "this is a paragraph".

HTML Page Structure

Below is a visualization of an HTML page structure:

```
<html>
<head>
<title>Page title</title>
</head>
<body>
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
<p>This is another paragraph.</p>
</body>
</html>
```

HTML Documents

All HTML documents must start with a document type declaration: <!DOCTYPE html>.

The HTML document itself begins with <html> and ends with </html>.

The visible part of the HTML document is between <body> and </body>

The <!DOCTYPE> Declaration

The <!DOCTYPE> declaration represents the document type, and helps browsers to display web pages correctly.

It must only appear once, at the top of the page (before any HTML tags).

The <!DOCTYPE> declaration is not case sensitive

HTML Headings

HTML headings are defined with the <h1> to <h6> tags.

<h1> defines the most important heading. <h6> defines the least important heading:

```
<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<h3>This is heading 3</h3>
```

HTML Paragraphs

HTML paragraphs are defined with the <p> tag:

Example

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

HTML Links

HTML links are defined with the <a> tag:

Example

```
<a href="https://www.w3schools.com">This is a link</a>
```

The link's destination is specified in the href attribute.

Attributes are used to provide additional information about HTML elements.

HTML Images

HTML images are defined with the <img> tag.

The source file (src), alternative text (alt), width, and height are provided as attributes:

Example

```
<img src="w3schools.jpg" alt="W3Schools.com" width="104" height="142">
```

HTML Attributes

- All HTML elements can have **attributes**
- Attributes provide **additional information** about elements
- Attributes are always specified in **the start tag**
- Attributes usually come in name/value pairs like: **name="value"**

The href Attribute

The <a> tag defines a hyperlink. The href attribute specifies the URL of the page the link goes to:

Example

Visit W3Schools

The src Attribute

The <img> tag is used to embed an image in an HTML page. The src attribute specifies the path to the image to be displayed:

Example

<img src="img_girl.jpg">

There are two ways to specify the URL in the src attribute:

1. Absolute URL - Links to an external image that is hosted on another website. Example: src="https://www.w3schools.com/images/img_girl.jpg".

Notes: External images might be under copyright. If you do not get permission to use it, you may be in violation of copyright laws. In addition, you cannot control external images; it can suddenly be removed or changed.

2. Relative URL - Links to an image that is hosted within the website. Here, the URL does not include the domain name. If the URL begins without a slash, it will be relative to the current page. Example: src="img_girl.jpg". If the URL begins with a slash, it will be relative to the domain. Example: src="/images/img_girl.jpg".

Tip: It is almost always best to use relative URLs. They will not break if you change domain.

The width and height Attributes

The <img> tag should also contain the width and height attributes, which specify the width and height of the image (in pixels):

Example

<img src="img_girl.jpg" width="500" height="600">

The alt Attribute

The required alt attribute for the <img> tag specifies an alternate text for an image, if the image for some reason cannot be displayed. This can be due to a slow connection, or an error in the src attribute, or if the user uses a screen reader.

Example

<img src="img_girl.jpg" alt="Girl with a jacket">

Example

See what happens if we try to display an image that does not exist:

<img src="img_typo.jpg" alt="Girl with a jacket">

The HTML Style Attribute

Setting the style of an HTML element, can be done with the style attribute.

The HTML style attribute has the following syntax:

<*tagname* style="*property*:*value;*">

The ***property*** is a CSS property. The ***value*** is a CSS value.

Background Color

The CSS background-color property defines the background color for an HTML element.

Example

Set the background color for a page to powderblue:

```
<body style="background-color:powderblue;">
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
</body>
```

Set background color for two different elements:

```
<body>
<h1 style="background-color:powderblue;">This is a heading</h1>
<p style="background-color:tomato;">This is a paragraph.</p>
</body>
```

Cascading Style Sheets (CSS)

Cascading Style Sheets (CSS) is used to format the layout of a webpage.

With CSS, you can control the color, font, the size of text, the spacing between elements, how elements are positioned and laid out, what background images or background colors are to be used, different displays for different devices and screen sizes, and much more!

Tip: The word **cascading** means that a style applied to a parent element will also apply to all children elements within the parent. So, if you set the color of the body text to "blue", all headings, paragraphs, and other text elements within the body will also get the same color (unless you specify something else)!

CSS can be added to HTML documents in 3 ways:

- **Inline** - by using the style attribute inside HTML elements
- **Internal** - by using a <style> element in the <head> section
- **External** - by using a <link> element to link to an external CSS file

The most common way to add CSS, is to keep the styles in external CSS files. However, in this tutorial we will use inline and internal styles, because this is easier to demonstrate, and easier for you to try it yourself

Inline CSS

An inline CSS is used to apply a unique style to a single HTML element.

An inline CSS uses the style attribute of an HTML element.

The following example sets the text color of the <h1> element to blue, and the text color of the <p> element to red:

Example

```
<h1 style="color:blue;">A Blue Heading</h1>

<p style="color:red;">A red paragraph.</p>
```

Internal CSS

An internal CSS is used to define a style for a single HTML page.

An internal CSS is defined in the <head> section of an HTML page, within a <style> element.

The following example sets the text color of ALL the <h1> elements (on that page) to blue, and the text color of ALL the <p> elements to red. In addition, the page will be displayed with a "powderblue" background color:

Example

```
<!DOCTYPE html>
<html>
<head>
<style>
body {background-color: powderblue;}
h1   {color: blue;}
p    {color: red;}
</style>
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

External CSS

An external style sheet is used to define the style for many HTML pages.

To use an external style sheet, add a link to it in the <head> section of each HTML page:

Example

```
<!DOCTYPE html>
<html>
<head>
  <link rel="stylesheet" href="styles.css">
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

The external style sheet can be written in any text editor. The file must not contain any HTML code, and must be saved with a .css extension.

Here is what the "styles.css" file looks like:

"styles.css":

```
body {
  background-color: powderblue;
}
h1 {
  color: blue;
}
p {
  color: red;
}
```

HTML5

Introduction: HTML stands for Hyper Text Markup Language. It is used to design web pages using a markup language. HTML is an abbreviation of Hypertext and Markup language. Hypertext defines the link between the web pages. The markup language is used to define the text document within the tag which defines the structure of web pages. HTML 5 is the fifth and current version of HTML. It has improved the markup available for documents and has introduced application programming interfaces (API) and Document Object Model (DOM).

Features:

- It has introduced new multimedia features which supports both audio and video controls by using <audio> and <video> tags.

- There are new graphics elements including vector graphics and tags.
- Enrich semantic content by including <header> <footer>, <article>, <section> and <figure> are added.
- Drag and Drop- The user can grab an object and drag it further dropping it to a new location.
- Geo-location services- It helps to locate the geographical location of a client.
- Web storage facility which provides web application methods to store data on the web browser.
- Uses SQL database to store data offline.
- Allows drawing various shapes like triangle, rectangle, circle, etc.
- Capable of handling incorrect syntax.
- Easy DOCTYPE declaration i.e., <!doctype html>
- Easy character encoding i.e., <meta charset="UTF-8″>

New Added Elements in HTML 5:

- **<article>:** The <article> tag is used to represent an article. More specifically, the content within the <article> tag is independent from the other content of the site (even though it can be related).
- **<aside>:** The <aside> tag is used to describe the main object of the web page in a shorter way like a highlighter. It basically identifies the content that is related to the primary content of the web page but does not constitute the main intent of the primary page. The <aside> tag contains mainly author information, links, related content and so on.
- **<figcaption>:** The <figcaption> tag in HTML is used to set a caption to the figure element in a document.
- **<figure>:** The <figure> tag in HTML is used to add self-contained content like illustrations, diagrams, photos or codes listing in a document. It is related to main flow, but it can be used in any position of a document and the figure goes with the flow of the document and if it is removed it should not affect the flow of the document.
- **<header>:** It contains the section heading as well as other content, such as a navigation links, table of contents, etc.
- **<footer>:** The <footer> tag in HTML is used to define a footer of HTML document. This section contains the footer information (author information, copyright information, carriers etc.). The footer tag is used within body tag. The <footer> tag is new in the HTML 5. The footer elements require a start tag as well as an end tag.
- **<main>:** Delineates the main content of the body of a document or web app.

- **<mark>:** The <mark> tag in HTML is used to define the marked text. It is used to highlight the part of the text in the paragraph.
- **<nav>:** The <nav> tag is used to declaring the navigational section in HTML documents. Websites typically have sections dedicated to navigational links, which enables user to navigate the site. These links can be placed inside a nav tag.
- **<section>:** It demarcates a thematic grouping of content.
- **<details>:** The <details> tag is used for the content/information which is initially hidden but could be displayed if the user wishes to see it. This tag is used to create interactive widget which user can open or close it. The content of details tag is visible when open the set attributes.
- **<summary>:** The <summary> tag in HTML is used to define a summary for the <details> element. The <summary> element is used along with the <details> element and provides a summary visible to the user. When the summary is clicked by the user, the content placed inside the <details> element becomes visible which was previously hidden. The <summary> tag was added in HTML 5. The <summary> tag requires both starting and ending tag.
- **<time>:** The <time> tag is used to display the human-readable data/time. It can also be used to encode dates and times in a machine-readable form. The main advantage for users is that they can offer to add birthday reminders or scheduled events in their calendars and search engines can produce smarter search results.
- **<bdi>:** The <bdi> tag refers to the Bi-Directional Isolation. It differentiates a text from other text that may be formatted in different direction. This tag is used when a user generated text with an unknown direction.
- **<wbr>:** The <wbr> tag in HTML stands for word break opportunity and is used to define the position within the text which is treated as a line break by the browser. It is mostly used when the used word is too long and there are chances that the browser may break lines at the wrong place for fitting the text.
- **<datalist>:** The <datalist> tag is used to provide autocomplete feature in the HTML files. It can be used with input tag, so that users can easily fill the data in the forms using select the data.
- **<keygen>:** The <keygen> tag in HTML is used to specify a key-pair generator field in a form. The purpose of <keygen> element is to provide a secure way to authenticate users. When a form is submitted then two keys are generated, private key and public key. The private key stored locally, and the public key is sent to the server. The public key is used to generate client certificate to authenticate user in future.
- **<output>:** The <output> tag in HTML is used to represent the result of a calculation performed by the client-side script such as JavaScript.
- **<progress>:** It is used to represent the progress of a task. It also defines how much work is done and how much is left to download a task. It is not used to represent the disk space or relevant query.

- **<svg>:** It is the Scalable Vector Graphics.
- **<canvas>:** The <canvas> tag in HTML is used to draw graphics on web page using JavaScript. It can be used to draw paths, boxes, texts, gradient and adding images. By default, it does not contain border and text.
- **<audio>:** It defines the music or audio content.
- **<embed>:** Defines containers for external applications (usually a video player).
- **<source>:** It defines the sources for <video> and <audio>.
- **<track>:** It defines the tracks for <video> and <audio>.
- **<video>:** It defines the video content.

Example 1:

```
<!DOCTYPE html>
<html>
<head>
   <title>HTML 5</title>
   <style>
      h1 {
         font-size:50px;
      }
   </style>
</head>
<body>
   <h1>GeeksforGeeks</h1>
</body>
</html>
```

Output:

Example 2:

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

```
<head>
   <title>HTML 5 Demo</title>

   <style>
      .GFG {
         font-size:40px;
         font-weight:bold;
         color:green;
      }
      body {
         text-align:center;
      }
   </style>
</head>

<body>
   <div class = "GFG">GeeksforGeeks</div>
   <aside>
      <div>A computer science portal for geeks</div>
   </aside>
</body>
</html>
```

Output:

XHTML

XHTML or **EXtensible HyperText Markup Language** is a mix of HTML and XML, very similar to HTML but stricter. It's like a rulebook for creating web pages that browsers easily understand. Unlike HTML, you have to be careful and follow the rules exactly. Most browsers support it. Just think of it as a more precise way to write web code.

Elements of XHTML:

When creating an XHTML web page, it is necessary to include a DTD (Document Type Definition) declaration. There are three types of DTD which are discussed below:

Transitional DTD:

It is supported by the older browsers which do not have inbuilt cascading style sheets supports. Several attributes are enclosed in the body tag which are not allowed in strict DTD.

Syntax:

```
<!DOCTYPE html
PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"DTD/xhtml1-transitional.dtd">

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

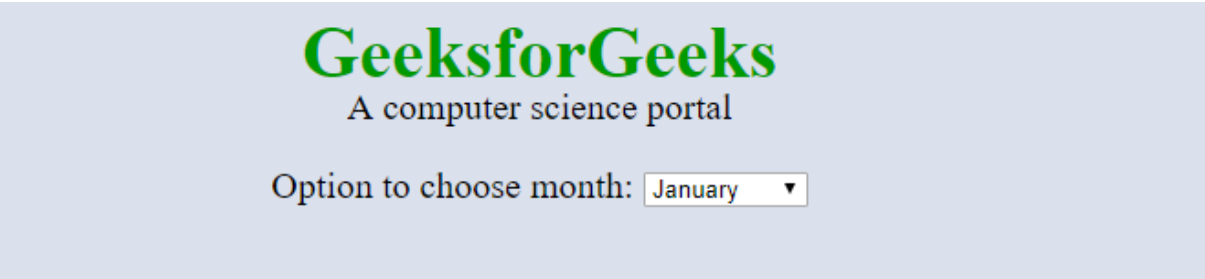
Example: In this example we will see the code for writing an XHTML document with an example.

- html

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">

<head>
   <title>Transitional DTD XHTML</title>
</head>

<body bgcolor="#dae1ed">
   <div style="color:#090;font-size:40px;
            font-weight:bold;text-align:center;
            margin-bottom:-25px;">GeeksforGeeks</div>
   <p style="text-align:center;font-size:20px;">
```

```
    A computer science portal</p>
  <p style="text-align:center;font-size:20px;">
    Option to choose month:
    <select name="month">
      <option selected="selected">January</option>
      <option>February</option>
      <option>March</option>
      <option>April</option>
      <option>May</option>
      <option>June</option>
      <option>July</option>
      <option>Augusy</option>
      <option>September</option>
      <option>October</option>
      <option>November</option>
      <option>December</option>
    </select>
  </p>
</body>

</html>
```

Output:



Strict DTD:

Strict DTD is used when XHTML page contains only markup language. Strict DTD is used together with cascading style sheets, because this attribute does not allow CSS property in body tag.

Syntax:

```
<!DOCTYPE html
PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
```

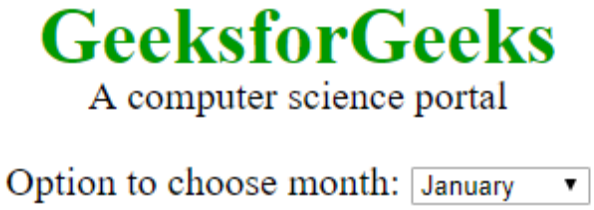
Example 2: In this example we will see the code for writing an XHTML document with an example for strict DTD.

- html

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">

<head>
   <title>Strict DTD XHTML</title>
</head>

<body>
   <div style="color:#090;font-size:40px;
            font-weight:bold;text-align:center;
            margin-bottom:-25px;">GeeksforGeeks</div>
   <p style="text-align:center;font-size:20px;">
      A computer science portal</p>
   <p style="text-align:center;font-size:20px;">
      Option to choose month:
      <select name="month">
         <option selected="selected">January</option>
```

```
        <option>February</option>
        <option>March</option>
        <option>April</option>
        <option>May</option>
        <option>June</option>
        <option>July</option>
        <option>Augusy</option>
        <option>September</option>
        <option>October</option>
        <option>November</option>
        <option>December</option>
      </select>
    </p>
</body>
</html>
```

Output:



Frameset DTD

The frameset DTD is used when XHTML page contains frames. This DTD is identical to the HTML 4.01 Transitional DTD except for the content model of the HTML element.

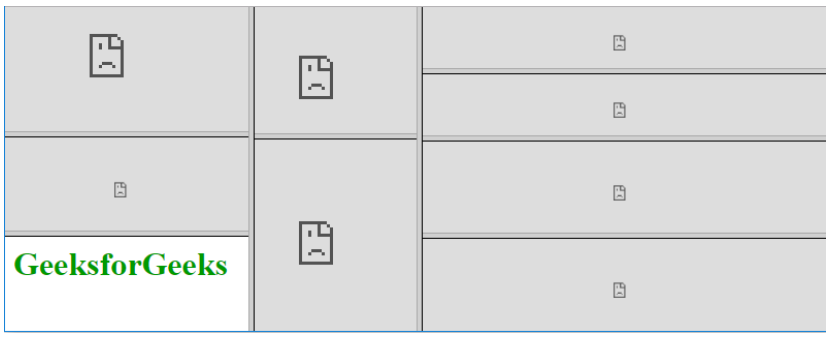
Syntax:

```
<!DOCTYPE html
PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"
"DTD/xhtml1-frameset.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
```

Example 2: In this example we will see the code for writing an XHTML document with an example for frameset DTD.

```
<head>
   <title>Frameset DTD XHTML</title>
</head>
<frameset cols="30%, 20%, *">
   <frameset rows="40%, 30%, *">
      <frame src="gfg.html" />
      <frame src="gfg1.html" />
      <frame src="geeks.html" />
   </frameset>
   <frameset rows="40%, 60%">
      <frame src="g4g.html" />
      <frame src="g4g1.html" />
   </frameset>
   <frameset rows="20%, 20%, 30%, *">
      <frame src="geeksforgeeks.html" />
      <frame src="geeksforgeeks1.html" />
      <frame src="geeksforgeeks2.html" />
      <frame src="geeksforgeeks3.html" />
   </frameset>
</frameset>
</html>
```

Output:



Why use XHTML?

- XHTML documents are validated with standard XML tools.
- It is easily to maintain, convert, edit document in the long run.
- It is used to define the quality standard of web pages.
- XHTML is an official standard of the W3C, your website becomes more compatible and accurate with many browsers.

Benefits of XHTML:

- All XHTML tags must have closing tags and are nested correctly. This generates cleaner code.
- XHTML documents are lean which means they use less bandwidth. This reduces cost particularly if your web site has 1000s of pages.
- XHTML documents are well formatted well–formed and can easily be transported to wireless devices, Braille readers and other specialized web environments.
- All new developments will be in XML (of which XHTML is an application).
- XHTML works in association with CSS to create web pages that can easily be updated.

Difference Between HTML and XHTML:

HTML	**XHTML**
HTML or HyperText Markup Language is the main markup language for creating web pages	XHTML (Extensible HyperText Markup Language) is a family of XML markup languages that mirror or extend versions of the widely used Hypertext Markup Language (HTML)
Flexible framework requiring lenient HTML specific parser	Restrictive subset of XML which needs to be parsed with standard XML parsers
Proposed by Tim Berners-Lee in 1987	World Wide Web Consortium Recommendation in 2000.
Application of Standard Generalized Markup Language (SGML).	Application of XML

HTML	XHTML
Extended from SGML.	Extended from XML, HTML

CSS3

CSS3 (Cascading Style Sheets Level 3) is a language used to style HTML web pages. It's the most recent version of Cascading Style Sheets (CSS).

What does CSS3 do?

- **Design**: CSS3 allows developers to create more complex designs for web pages.
- **Animations**: CSS3 allows developers to add more complex animations, such as transitions, transforms, and special effects.
- **Opacity**: CSS3 allows developers to make elements partially or fully transparent.
- **Rounded corners**: CSS3 allows developers to make elements look more professional by rounding off their corners.
- **Text and box shadows**: CSS3 allows developers to add text and box shadows to elements.
- **Web fonts**: CSS3 allows developers to use web fonts, such as those available through Google Fonts and Typecast.

Why use CSS3?

- CSS3 makes web pages more attractive and user-friendly.
- CSS3 makes web pages more dynamic and interactive.
- CSS3 makes web pages more accessible.
- CSS3 makes web pages load faster.
- CSS3 makes web pages more usable.

What browsers support CSS3?

- CSS3 is supported by most modern web browsers, including Google Chrome, Firefox, Safari, and Opera.