HTML Vs HTML 5

| **HTML** | | **HTML5** | |
| --- | --- | --- | --- |
| It didn’t support audio and video without the use of flash player support. | | It supports audio and video controls with the use of <audio> and <video> tags. | |
| It uses cookies to store temporary data. | | It uses SQL databases and application cache to store offline data. | |
| Does not allow JavaScript to run in the browser. | | Allows JavaScript to run in the background. This is possible due to JS Web worker API in HTML5. | |
| Vector graphics are possible in HTML with the help of various technologies such as VML, Silver-light, Flash, etc. | | Vector graphics are additionally an integral part of HTML5 like SVG and Canvas. | |
| It does not allow drag and drop effects. | | It allows drag and drop effects. | |
| Not possible to draw shapes like circle, rectangle, triangle etc. | | HTML5 allows to draw shapes like circle, rectangle, triangle etc. | |
| It works with all old browsers. | | It supported by all new browser like Firefox, Mozilla, Chrome, Safari, etc. | |
| <HTML>,<Body> , and <Head> tags are mandatory while writing a HTML code. | | These tags can be omitted while writing HTML code. | |
| Older version of HTML are less mobile-friendly. | | HTML5 language is more mobile-friendly. | |
| Doctype declaration is too long and complicated. | | Doctype declaration is quite simple and easy. | |
| Elements like nav, header were not present. | | New element for web structure like nav, header, footer etc. | |
| Character encoding is long and complicated. | | Character encoding is simple and easy. | |
| It is almost impossible to get true GeoLocation of user with the help of browser. | | One can track the GeoLocation of a user easily by using JS GeoLocation API. | |
| It can not handle inaccurate syntax. | | It is capable of handling inaccurate syntax. | |
| Being an older version , it is not fast , flexible , and efficient as compared to HTML5. | | It is efficient, flexible and more fast in comparison to HTML. | |
| **Tags (Elements)** | **Description** | | | |
| <article> | Represents an independent piece of content of a document, such as a blog entry or newspaper article | | | |
| <aside > | Represents a piece of content that is only slightly related to the rest of the page. | | | |
| <audio> | Defines an audio file. | | | |
| <canvas> | This is used for rendering dynamic bitmap graphics on the fly, such as graphs or games. | | | |
| <command> | Represents a command the user can invoke. | | | |
| <datalist> | Together with the a new list attribute for input can be used to make comboboxes | | | |
| <details> | Represents additional information or controls which the user can obtain on demand | | | |
|  | | |
| <figure> | Represents a piece of self-contained flow content, typically referenced as a single unit from the main flow of the document. | | | |
| <footer> | Represents a footer for a section and can contain information about the author, copyright information, et cetera. | | | |
| <header> | Represents a group of introductory or navigational aids. | | | |
| <hgroup> | Represents the header of a section. | | | |
|  |  | | | |
| <mark> | Represents a run of text in one document marked or highlighted for reference purposes, due to its relevance in another context. | | | |
| <meter> | Represents a measurement, such as disk usage. | | | |
|  |  | | | |
| <output> | Represents some type of output, such as from a calculation done through scripting. | | | |
| <progress> | Represents a completion of a task, such as downloading or when performing a series of expensive operations. | | | |
|  |  | | | |
| <section> | Represents a generic document or application section | | | |
| <time > | Represents a date and/or time. | | | |
| <video> | Defines a video file. | | | |
| <wbr> | Represents a line break opportunity. | | | |

**Input Type attributes:**

1. [**color**](https://www.geeksforgeeks.org/html-input-typecolor/)**:** This input type allows the user to select a color from a color picker.
2. [**date**](https://www.geeksforgeeks.org/html-input-typedate/)**:**Thisinput type allows the user to select a date from a drop-down calendar.
3. [**time:**](https://www.geeksforgeeks.org/html-input-typetime/)This input type allows the user to enter a time.
4. **datetime:** This input type allows the user to select date and time along with timezone.
5. **datetime-local:**This input type allows the user to select both local date and time.
6. [**week**](https://www.geeksforgeeks.org/html-input-typeweek/)**:** This input type allows the user to select week and year from the drop-down calendar.
7. [**email**](https://www.geeksforgeeks.org/html-input-typeemail/)**:**This input type allows the user to enter an e-mail address.
8. [**month**](https://www.geeksforgeeks.org/html-input-typemonth/)**:**This input type allows the user to select a month and year from a drop-down calendar.
9. **number:** This input type allows the user to enter a numerical value.
10. **range:**This input type allows the user to enter a numerical value within aspecified range**.**
11. **search:** This input type allows the user to enter a search string within the input field.
12. **tel:**This input type allows the user to enter a telephone number.
13. **url:**This input type allows the user to enter the URL.

We will use the above attributes & understand their usage through the example.

**Example 1:**In this example, you will get to know about **color, date and time**input type.

**Date Syntax:**

<input type="date">

**Time Syntax:**

<input type="time">

**Color Syntax:**

<input type="color">

**Note:** **date** and**time**are not supported by the Internet Explorer and Safari browsers.

<!DOCTYPE html>

<**html**>

<**head**>

<**style**>

**h1** {

**color**: **green**;

**text-align**: **center**;

}

**form** {

**text-align**: **center**;

**color**: **red;**

}

</**style**>

</**head**>

<**body**>

<**h1**>Welcome </**h1**>

<**form**>

<**label** for="color">Select Color : </**label**>

<**input** type="color" />

</**form**>

<**br** />

<**form**>

<**label** for="date">Select Date : </**label**>

<**input** type="date" />

</**form**>

<**br** />

<**form**>

<**label** for="time">Select Time :</**label**>

<**input** type="time" />

</**form**>

</**body**>

</**html**>

AUDIO

<audio>  
  <source src="horse.ogg">  
  <source src="duck.mp3">  
  Your browser does not support the audio tag.  
</audio>

Video

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

SVG

SVG Shapes

SVG has some predefined shape elements that can be used by developers:

* Rectangle <rect>
* Circle <circle>
* Ellipse <ellipse>
* Line <line>
* Polyline <polyline>
* Polygon <polygon>
* Path <path>

<svg width="300" height="130”>  
  <circle width="200" height="100" x="10" y="10"  fill="red" />  
</svg>

<svg height="100" width="100">  
  <circle r="45" cx="50" cy="50" fill="red" />  
</svg>

Example:

<!DOCTYPE html>

<html>

<body>

<h2>SVG rect Element</h2>

<svg width="300" height="130">

  <rect width="200" height="100" x="10" y="10" rx="20" ry="20" fill="blue" />

Sorry, your browser does not support inline SVG.

</svg>

</body>

</html>

CSS:

What is CSS?

* CSS stands for Cascading Style Sheets
* CSS describes how HTML elements are to be displayed on screen, paper, or in other media
* CSS saves a lot of work. It can control the layout of multiple web pages all at once
* External stylesheets are stored in CSS files

<!DOCTYPE html>

<html>

<head>

<style>

p {

color: red;

text-align: center;

}

</style>

</head>

<body>

<p>Hello World!</p>

<p>These paragraphs are styled with CSS.</p>

</body>

</html>

Three Ways to Insert CSS

There are three ways of inserting a style sheet:

* External CSS
* Internal CSS
* Inline CSS
* Example
* External styles are defined within the <link> element, inside the <head> section of an HTML page:
* <!DOCTYPE html>  
  <html>  
  <head>  
  <link rel="stylesheet" type="text/css" href="mystyle.css">  
  </head>  
  <body>  
    
  <h1>This is a heading</h1>  
  <p>This is a paragraph.</p>  
    
  </body>  
  </html>
* An external style sheet can be written in any text editor, and must be saved with a .css extension.
* The external .css file should not contain any HTML tags.
* Here is how the "mystyle.css" file looks:
* "mystyle.css"
* body {  
    background-color: lightblue;  
  }  
    
  h1 {  
    color: navy;  
    margin-left: 20px;  
  }
* Internal CSS
* An internal style sheet may be used if one single HTML page has a unique style.
* The internal style is defined inside the <style> element, inside the head section.
* Example
* Internal styles are defined within the <style> element, inside the <head> section of an HTML page:
* <!DOCTYPE html>  
  <html>  
  <head>  
  <style>  
  body {  
    background-color: linen;  
  }  
    
  h1 {  
    color: maroon;  
    margin-left: 40px;  
  }  
  </style>  
  </head>  
  <body>  
    
  <h1>This is a heading</h1>  
  <p>This is a paragraph.</p>  
    
  </body>  
  </html>
* Inline CSS
* An inline style may be used to apply a unique style for a single element.
* To use inline styles, add the style attribute to the relevant element. The style attribute can contain any CSS property.
* Example
* Inline styles are defined within the "style" attribute of the relevant element:
* <!DOCTYPE html>  
  <html>  
  <body>  
    
  <h1 style=" color: maroon;margin-left: 40px; ">This is a heading</h1>  
  <p style="color:red;">This is a paragraph.</p>  
    
  </body>  
  </html>