Versions of HTML

YEAR	VERSION
1991	HTML
1993	HTML+
1995	HTML 2.0
1997	HTML 3.2
1999	HTML 4.01
2000	XHTML
2012	HTML 5

HTML -1991

- 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. It does not have rich features like styling and other things that were related to how content will be rendered in a browser.
- The initial version of HTML does not provide support for tables, font support, etc., as it provides us in the latest version.
- We would also like to discuss that W3C did not exist before HTML 2.0; hence it does not show details about HTML 1.

HTML 2.0 -1995

- HTML version 2.0 was developed in 1995 with basic intention of improving HTML version 1.0
- Now a standard got started to develop so as to maintain common rules and regulations across different browsers.
- In HTML 2.0 version concept of form came into force. Forms were developed, but still, they had basic tags like text boxes, buttons, etc.
- Also, the table came as an HTML tag.
- W3C was also formed. The main intention of W3C is to maintain standard across different web browsers so that these browsers understand and render HTML tags in a similar manner.

HTML 3.2 -1997

- 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. It is worth noting that because of W3C standard maintenance, the newer version of HTML was 3.2 instead of 3.
- Now, HTML 3.2 has better support for new form elements. Another important feature what HTML 3.2 implemented was support for CSS. CSS stands for Cascading Style Sheet.
- With the upgradation of browsers to HTML 3.2, the browser also supported for frame tags, although HTML specifications still do not support frame markup tags.

HTML 4.0 -1999

- It was developed in 1999. It extended the support of cascading styling sheets. In version 3.2, CSS were embedded in HTML page itself. Therefore, if the website has various web pages to apply to the style of each page, we must place CSS on each web page. Hence there was a repetition of the same block of CSS.
- To overcome this thing, in version 4.01 concept of an external styling sheet emerged. Under this concept, an external CSS file could be developed, and this external styling file could be included in HTML itself. HTML 4.01 provided support for further new tags of HTML.

Now,
what is HTML5?

What is HTML5?

HTML5 is the new standard for HTML. – around 2012, almost after one decade of HTML 4.0
As, The previous version of HTML was – HTML 4.01, came in 1999.

- HTML5 is designed to deliver almost everything you want to do online without requiring additional plugins. It does everything from animation to apps, music to movies, and can also be used to build complicated applications that run in your browser.

- HTML5 is also cross-platform (it does not care whether you are using a tablet or a smartphone, a notebook, notebook or a Smart TV).

Advantages of HTML5

- It helps developers to develop an application that fits different resolutions, screen sizes, aspect ratios, and guidelines, and it automatically fit for each and every mobile and desktop screen(responsive).
- With HTML5, programmers can use some advanced features such as GPS, camera, and accelerometer in their web applications.
- JavaScript, HTML, and CSS are the main programming languages for web applications.
- We can easily convert HTML5 web applications to mobile applications. This is the main advantage of HTML.
- Easy to write, understand and use.
- Every browser understand html language, You can develop rich graphics and animations.

Disadvantages of HTML5

- HTML5 works only on updated browsers.
- Not flexible as Flash

** Important – To support different cross browser compatibility and to support older browsers like Internet explorer 6.0, we have “polyfills” javascript library , it refers to a JavaScript library that implements an HTML5 or CSS web standard, either an established standard on older browsers, or a proposed standard on existing browsers.

Differences Between HTML4 and HTML5

Differences Between HTML5 & HTML4

HTML 5

1. HTML5 uses new structures such as drag, drop and much more.
2. HTML 5 can contain embedded video and audio without using flash player.
3. HTML 5 is capable of handling inaccurate syntax
4. HTML 5 introduced many new API's which facilitate flexibility of web pages.
5. In HTML 5, new tags and new features like local storage are enhanced.

HTML 4

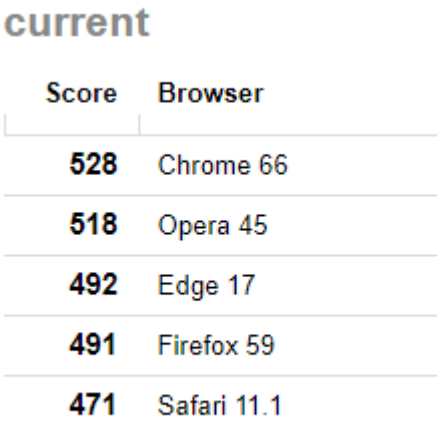
1. HTML 4 uses common structures like headers , footers.
2. HTML 4 cannot embed video or audio directly and makes use of flash player for it.
3. HTML 4 cannot handle inaccurate syntax
4. HTML 4 has traditional API's which does not include canvas and content editable API's.
5. In HTML 4, local storage is not possible and tags that can handle only one dimension are present

Browser Support for HTML5

Browser Support for HTML5

- HTML5 is not yet an official standard, and no browsers have full HTML5 support.
- But all major browsers (Chrome, Firefox, Opera) continue to add new HTML5 features to their latest versions.
- With this link, you can test your browser compatibility, https://html5test.com/index.html

Below are the current versions of desktop browser scores w.r.t. HTML5 features.



current

Score	Browser
528	Chrome 66
518	Opera 45
492	Edge 17
491	Firefox 59
471	Safari 11.1

HTML5 Document

The HTML5 <!DOCTYPE>

In HTML5 there is only one **<!doctype>** declaration, and it is very simple:

```
<!DOCTYPE html>
```

Minimum HTML5 Document

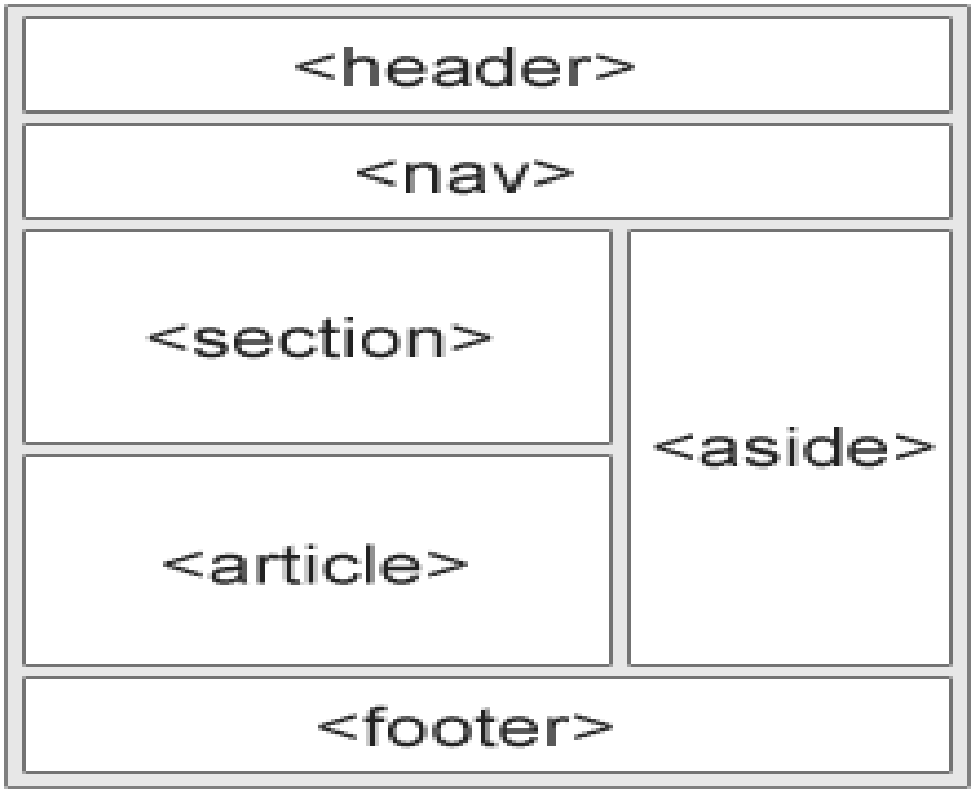
Below is a simple HTML5 document, with the minimum of required tags:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Title of the document</title>
</head>

<body>
Content of the document.....
</body>

</html>
```

HTML5 New Elements

HTML5 Semantic Elements:

- <header>
- <nav>
- <section>
- <article>
- <aside>
- <figcaption>
- <figure>
- <footer>



New Semantic/Structural Elements

Tag	Description
<article>	Defines an article, This tag is used to define the self-contained content inside the HTML document
<aside>	This tag is used to display a section of a page aside from the related section.
<details>	This tag is used to display additional information which can be hidden using the extra button option.
<header>	This tag is used to define the header content of a section or document.
<footer>	Defines additional details that the This tag is used to define the footer content of its nearest or sectioning root element.
<dialog>	Defines a dialog box or window
<nav>	This tag is used to define the links to other page or parts within the same page

New Semantic/Structural Elements

Tag	Description
<bdi>	Isolates a part of text that might be formatted in a different direction from other text outside it
<command>	Defines a command button that a user can invoke
<dialog>	Defines a dialog box or window
<figure>	Specifies self-contained content, like illustrations, diagrams, photos, code listings, etc.
<figcaption>	Defines a caption for a <figure> element
<mark>	Defines marked/highlighted text
<meter>	Defines a scalar measurement within a known range (a gauge)

New Semantic/Structural Elements

Tag	Description
<progress>	Represents the progress of a task
<ruby>	Defines a ruby annotation (for East Asian typography)
<rt>	Defines an explanation/pronunciation of characters (for East Asian typography)
<rp>	Defines what to show in browsers that do not support ruby annotations
<section>	Defines a section in a document
<time>	Defines a date/time
<wbr>	Defines a possible line-break

Non - Semantic Elements

Tag	Description
	This tag is used to declare a form that contains many pairs.
	This tag is used to create a table on a web page.
<article>	This tag is used to define the self-contained composition inside the HTML content.

Practical example

```
<!DOCTYPE html>
<html>
<head>
<title>Header Element for the page</title>
<style>
color:red;
h1, h4 {
Color:#red;
Text-align:centre;
margin-bottom:2px;
}
p {
font-size:20px;
text-align:centre;
margin-top:2px;
}
</style>
</head>
……..

</html>
```

```
<body style="background-color:LightCyan;">
<article>
<header>
<h1>Semantic example</h1>
<nav>
<a href="https://www.Test.com/">Home</a> |
<a href="https://www.Test.com/data-science/">Data Science</a> |
<a href="https://www.Test.com/software-development/">Software Development</a> |
<a href="https://www.Test.com/design/">Design</a> |
<a href="https://www.Test.com/finance/">Finance</a>
</nav>
<details>
<h3>Become a technicall learner with Testing</h3>
<section>
<h1><mark>AWS Tutorials</mark></h1>
<p>Amazon Web Services (AWS) is one of the world's most popular and used cloud services. 175 supported services are available in AWS. </p>
</section>
<section>
<h1><mark>Data Mining</mark></h1>
<p>Through technological development such as big data and data science companies around the world have benefited from data mining by recognize opportunities and making their organizations efficient through waste reduction, to achieve their business goals.
</p>
</section>
<p><em>This is the best online training institute in ASIA</em></p>
</details></header>
</article>
<footer>copyright @2021</footer>
</body>
</html>
```

HTML <Script> Tag

- The HTML `<script>` tag is used to define a client-side script (JavaScript).
- The `<script>` element either contains script statements, or it points to an external script file through the `src` attribute.
- Common uses for JavaScript are image manipulation, form validation, and dynamic changes of content.
- To select an HTML element, JavaScript most often uses the `document.getElementById()` method.

HTML <Script> Tag

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

<h2>Use JavaScript to Change Text</h2>
<p>This example writes "Hello JavaScript!" into an HTML element
with id="demo":</p>

<p id="demo"></p>

<script>
document.getElementById("demo").innerHTML = "Hello JavaScript!";
</script>

</body>
</html>
```

HTML <Head> Tag

The HTML <head> Element is a container for following elements:

- <title>-element defines the title of the document
  - ✓ defines a title in the browser toolbar
  - ✓ provides a title for the page when it is added to favorites
  - ✓ displays a title for the page in search engine-results
- <style>- element is used to define style information for a single HTML page
- <link>- element defines the relationship between the current document and an external resource, this tag is most often used to link to external style sheets
- <script>- element is used to define client-side JavaScripts.
- <base>- element specifies the base URL and/or target for all relative URLs in a page and there can only be single <base> element in a document.

HTML <Head> Tag

- <meta>-element is typically used to specify the character set, page description, keywords, author of the document, and viewport settings.

- The metadata will not be displayed on the page, but are used by browsers (how to display content or reload page), by search engines (keywords), and other web services.

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <meta name="description" content="Free Web tutorials">
  <meta name="keywords" content="HTML, CSS, JavaScript">
  <meta name="author" content="John Doe">
</head>
<body>
<p>All meta information goes inside the head section.</p>
</body></html>
```

Responsive Design

- Responsive Web Design is about using HTML and CSS to automatically resize, hide, shrink, or enlarge, a website, to make it look good on all devices (desktops, tablets, and phones)

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
</head>
<body>

<h2>Setting the Viewport</h2>
<p>This example does not really do anything, other than showing you how to add the viewport meta element.</p>

</body>
</html>
```

The New <canvas> Element

The <canvas> element is used to draw graphics, on the fly, via scripting (usually JavaScript).

New Media Elements

Tag	Description
<audio>	Defines sound content
<video>	Defines a video or movie
<source>	Defines multiple media resources for <video> and <audio>
<embed>	Defines a container for an external application or interactive content (a plug-in)
<track>	Defines text tracks for <video> and <audio>

New Form Elements

Tag	Description
<datalist>	Specifies a list of pre-defined option for input controls
<keygen>	Defines a key-pair generator field (for forms)
<output>	Defines the result of a calculation

Removed Elements

The following HTML 4.01 elements are removed from HTML5:

- ✓ <acronym>
- ✓ <applet>
- ✓ <basefont>
- ✓ <big>
- ✓
- ✓ <dir>
- ✓ <font>
- ✓ <frame>
- ✓ <frameset>
- ✓ <noframes>
- ✓ <strike>
- ✓ <tt>

Practical example with validation example

Thanks

https://github.com/targetedtec/WebDesign/tree/main/HTML5