

CREATING WEB-PAGES

Using HTML5 and CSS3

HTML



CSS



Lesson 1

Introduction to the Web- Technologies. HTML Structure

Contents

Introduction to the Subject.	3
Introduction to the Markup Languages.	
Markup Language of the HTML	4
Development of the HTML, its Versions.	
Current Versions: HTML and XHTML	4
Cross-browser Compatibility. Browser wars	6
W3C	7
Tags: the Main Elements of the HTML Structure	10
Concept of Tags and Attributes	10
Tag Assignment	11
Rules for Writing Tags and Attributes	16
Basic Mistakes in Tag Writing	18
Ancestors of HTML4/5 and XHTML: SGML and XML ...	20
Page Character Set Encodings and <meta> Tags	21
Character Entity References	23
Hometask.	25

Introduction to the Subject

The modern Internet provides various functions such as instant messaging, e-mail, stream video. However, the basis of the Internet is web sites. Nowadays there are more than 10 billion of sites, and their number continues to grow. The basis for the majority of sites is three technologies that allow providing information of various types.

- HTML technology helps to structure information and separate logical blocks on the website page.
- CSS technology is used to design and position, to present information on the page.
- JavaScript technology is used to implement the dynamics and interactive communication with the user.

The development of the HTML technology was caused by the necessity to create a simple tool for document layout to simplify the information surfing of the people without special technical expertise. Let's begin considering the HTML in details.

Introduction to the Markup Languages. Markup Language of the HTML

The hypertext markup language, HTML, is a set of rules and characters (tags) used for structuring (layout) of textual information, adding external image files, creating tables, lists, and so on. The text tagged with the help of HTML is interpreted by the browser, and as the result, users see not the source code with the page layout elements but the result of the page processing.

One of the features of HTML is the possibility to create links to third-party files (documents) or certain parts in files in the code, this significantly simplifies the navigation between pages and also makes it easier to use documents. Such a connection between documents in the HTML format is called hyperlinks, their usage allows providing a simple and convenient connection between files.

Development of the HTML, its Versions.

Current Versions: HTML and XHTML

Ease of using and simplicity of mastering the HTML allowed it to move beyond the limits of a scientific tool and defined it as the basis of the Internet. During formation and development of HTML, the following stages (versions) are usually distinguished:

- **HTML 0.9:** described the bases of the language and protocols, was developed in 1991.

- **RFC 1866** (or HTML 2.0): developed in November, 1995 provided the detailed description of the language, rules of writing tags and attributes, provided the description of versions in the HTML document. This version introduced the majority of the modern tags which nowadays are essential part of the HTML: headers, paragraphs, images, hyperlinks, and lists.
- **HTML 3.2** (accepted in January, 1997) — the main feature of this version was browser support (Microsoft and Netscape), in the future this feature made the developers to maintain the current standard. This version provided tables, word wrap, superscripts and subscripts.
- **HTML 4.0** (December, 1997) is not the standard, but the collection of W3C recommendations, including three types of the recommendations: strict, transitional, and frameset. The main function of the recommendations 4.0 was the decrease of the cross-browser incompatibility, legalization of the tags which were put to use by the browser manufactures without authorization.
- **HTML 4.01** (December, 1999) was first developed as a collection of recommendations, and two years later as the international standard of the standardization organization. It provided the main rules of tag and attribute usage.
- **HTML5** (2014). The main goal of the standard is the appearance of semantic tags specifically for the automatic page processing system. It provided a lot of new tags; some tags changed their original meaning.
- **HTML 5.1**. Was released in November, 2016 as a set of recommendations.

XHTML, extensible hypertext markup language, the markup language on the basis of the XML, was used along with the HTML for a long time.

The files written in the XHTML had a more strict syntax similar to XML rules and were processed by the XML parser, in addition, there was no error correction. This format was implemented to improve the opportunities of HTML and for compatibility with different network services.

The similarity between the HTML and XHTML syntax caused confusion in writing rules and appearance of various incorrectly written documents.

HTML5 ended the work on the XHTML, thus XHTML combined two standards.

Cross-browser Compatibility. Browser wars

As we can see from the HTML versions and adopting standards, initially the controlling bodies tried to coordinate the mapping rules implemented in different browsers in a different way. Moreover, there were only several players on the market, as the browsers from Microsoft and Netscape ruled the market for a long time, and this allowed these players to not stick to standards, but force the mapping rules and pages processing on users.

The situation, when a page displays differently in different browsers, makes it more difficult to write a page and leads to cross-browser incompatibility.

The popularization and the development of the Internet and increase of the user number led to the necessity to develop convenient tools to navigate through the pages, and this caused the development of alternative browsers.

One of the objectives of the alternative browsers was to gain the popularity, which was only possible by supporting the standards.

The fact that users began to prefer the alternative browsers, which often operated much faster and adequately than the browsers of the main players of the market, increased the competition and this led to the improvement of the browser quality, implementation of new functions to gain liking of users.

Technology race among the browsers caused by the competition is called the Browser wars. These wars allowed creating a lot of popular browsers which significantly moved forward from a simple page mapping.

Nowadays, there are many possibilities to view a page on the desktop as well as on mobile platforms, and this makes the developers to implement the cross-browser compatibility by both compliance with standard and compatibility implementation with the help of JavaScript or upload of the CSS styles made for a certain browser.

W3C

The popularity of the HTML and its usage as the basis for the Internet led to the necessity to control the HTML standards. The Timothy John Berners-Lee, the developer of the hypertext, established in 1994 and became the head of the organization which developed the HTML and its standards. This company was called World Wide Web Consortium, or W3C in abbreviated form.

The main objective of the consortium is to provide the openness of the Internet standards and its internationalization. According to this idea, all the documentation on the control of

the network operation has the status of recommendations and is open. Any developer has the right to implement the standards without any limitations and W3C fees.

The approach of openness and recommendations, and not obligations is the basic idea of the Internet development. However, the openness of the standards has also the disadvantage as it does not oblige the developers to maintain this standard and this makes it difficult to implement web sites for browsers of different developers.

Taking into account the idea of openness, W3C refused the certification idea, in other words, it does not check for the compliance to the standards and does not issue certificates. On the other hand, any developer can check if the logic of HTML documents is written correctly and if tags are used correctly by validating them on the W3C website, it is totally free of charge.

Validation is checking that HTML documents comply with the HTML standard declared in them. As the result of validation, the developer either passes the validation or gets a list of the recommended patches.

The **validator** is a program which checks the correspondence of the uploaded file to the HTML standards. The validator checks the following elements:

- Correct form of tags.
- Relevance of tags and the possibility to use them in this HTML version.
- Correctness of closing of tags and embedding of tags.
- The correct usage of attributes and many other things.

Then the result is output in the form of a list of errors and warnings.

The example of the validator of the Internet standard developer: [The W3C Markup Validation Service](#)

The developer should try to create pages in such a way so that they are validated.

When the W3C works on the HTML standards, all the recommendations pass the following stages: working draft, last call, candidate recommendation and proposed recommendation. During each stage, the content of recommendations may be completely changed. After these stages, the official recommendation of W3C appears.

During the operation of recommendations, they are often corrected and added, which is posted on the website. For example, the *hgroup* tag introduced in the HTML5 standard was soon recommended not to use in documents.

All the recommendations are available on the W3C website with a lot of examples.

Tags: the Main Elements of the HTML Structure

Concept of Tags and Attributes

To control the display of the document, HTML uses the control structures (sometimes they are called descriptors), tags. To separate the tags among the other document text, one uses special characters called angle brackets. The example of the tag:

```
<p>Hello</p>
```

Paired tags: the tags consisting of the opening tag (for example, `<p>`) and closing tag (for the above-mentioned example it is `</p>`). Pair tags wrap the code and allow changing its properties or display. There can be other tags inside the paired tags.

Paired tags in a document are the following:

- `<html> </html>;`
- `<head> </head>;`
- `<title> </title>;`
- `<body> </body>.`

Unpaired tags (aka self-closing, singular, or stand-alone tags) are the tags consisting of one element. unpaired tags are used to create one element on the page or to change the document properties, file attachment, etc.

The above-mentioned example has the following unpaired tags:

- `<!DOCTYPE html>;`
- `<meta>.`

Often the functions of unpaired and pair tags are changed by means of attributes which are indicated in the angle brackets after the tag name. For example: the above-mentioned example has the following attributes:

- lang="en";
- charset="utf-8".

The number of the attributes in the tag is not limited.

By default all the tags can be divided by the display type into two large groups:

- **Block-level elements:** elements which take all the line when displayed even if their content is smaller than the width of the browser window. By default these block-level elements cannot be arranged in one line, in other words, they take up the full width available. The headers (h1, h2, h3, h4...) and paragraphs are block-level elements.
- **Inline elements:** elements are displayed in one line one after another, Let`s discuss the purpose of the tags.

Tag Assignment

Let`s consider the tags used in an empty HTML document:

- **<!DOCTYPEhtml>:** self-closing tag at the beginning of the document. It tells the browser the standard which should be used to interpret the file. This tag refers to the usage of the modern HTML 5 standard. Reference to a standard limits the tags to be used, page layout rules. Now the standards of the versions 4.01, 4.0, 3.2 are considered to be out-of-date and are not recommended to use.

The example:

```
<!DOCTYPE html>
```

- **<html>**: tag which opens and closes an HTML page. Paired tag. It indicates the file panel processed by the browser as an HTML document. Closing `</html>` should be the last element on an HTML page.

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

- **<head>**: paired tag which is called the page header. The header of the page is the service field which is used to configure the page, control the display of the page and the connection of external files. Usually, tags in the header are not displayed for the user, but they directly influence the page view.

```
<!DOCTYPE html>  
<html>  
  <head>  
  </head>  
</html>
```

- **<meta>**: multifunctional tag extended with the help of attributes. In this context, it determines the page encoding. Initially, the encoding of the UTF-8 was developed for the OS, but soon it began to be used for the web. The advantages of the UTF-8 are the simultaneous operation with the several languages, the possibility to avoid code tables and to add hieroglyph characters to the code.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
  </head>
</html>
```

- **<title>** is a tag where the title of the page is written. Usually, it is displayed on the browser tab. The other HTML tags inside the title tag are not processed and will be displayed in the browser just like they are written in the HTML file. It is a paired tag:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Hello!</title>
  </head>
</html>
```

- **<body>** contains the whole content of the page. All the information we can see on the page is in the body tag. It is a paired tag.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Hello!</title>
  </head>
  <body>
  </body>
</html>
```

- **<h1>..**</h1>**, **<h2>..**</h2>**, **<h3>..**</h3>**, **<h4>..**</h4>**, **<h5>..**</h5>**, **<h6>..**</h6>****: the headers are used to create headers for different texts. Since text can have several headers of different levels and relevancy, HTML has six header levels. By default, *the headers are displayed in bold, the size of the font depends on the header level.***********

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Hello!</title>
  </head>
  <body>
    <h1>Header 1</h1>
    <h2>Header 2</h2>
    <h3>Header 3</h3>
    <h4>Header 4</h4>
    <h5>Header 5</h5>
    <h6>Header 6</h6>
  </body>
</html>
```

- **<p>..**</p>**** is a paragraph which is used to define complete blocks of text. It is a block-level element.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Hello!</title>
  </head>
  <body>
```

```

<h1>Header 1</h1>
<p>Text paragraph </p>
</body>
</html>

```

- **<hr>** horizontal line which is used to visually separate elements on the page. It is a block-level element.
- ****: the image (img), unpaired element. Image is an inline element, in other words, if there are several elements in series, by default they are displayed in one line (if the sizes of the image and browser window allow doing this). To display it correctly, one needs to have two attributes:
 - **src=""**: indicates the location of the image file. Thus the HTML document stores only a link to the image, and the file itself is stored either in local store or in remote store. Many browsers ignore the img tag if the src attribute is empty or absent.
 - **alt=""**: alternative text which, before the HTML5, was displayed for the user in case the image could not be uploaded. HTML5 helped this attribute to become obliged and to be used for screen readers. If there is no alt attribute, the browser will display the image; however, this document will not be validated.

```



```

- **..******
- is a hypertext reference (anchor) which is used to create a hyper transition to the other document or transition to a tag in the current document. When you hover over a hyper link, the cursor becomes a hand

with a pointed finger (pointer), and a hyper link text is underlined and highlighted in blue.

Hyperlink is a paired tag, inline element. In addition to the text itself inside the `<a>` tags, you can also write other tags. When you click the link, there may be the following options:

- if a link refers to another page, this page will be uploaded and displayed in the browser, in other words, one will go to another page.
- if the link refers to the file different from the HTML, there may be options: if the browser can display this document, it will be displayed (just like in case of the pdf format), if no, file download dialog window will be opened.
- if the link refers to the tag in the current document, one will go to this part of the current document.

```
<a href="http://ya.ru">Google web site</a>
```

- **..******: in the HTML5 standard, this tag is used for specifying bold word or word combination.
- **<i>..**</i>****: in HTML5 standard, this tag is used for the italicizing of word or word combination.
- **^{..**}**** is used to create a superscript. When it is used for a word or a character, it lifts them above the level of the main text and reduces them in sizes.
- **_{..**}**** creates a subscript. When it is used for a word or a character, it lowers them below the level of the main text and reduces them in sizes.
- **
** is a line break. It breaks the line and wraps it onto another line. It is not an alternative for paragraphs.

Rules for Writing Tags and Attributes

According to the current HTML 5 standard, tags can be written in uppercase as well as in lower case.

- `<BODY>` — correct;
- `<body>` — correct;
- `<BoDy>` — correct;
- `< body>` — incorrect, space between the opening angle bracket and the tag name are not admissible.

Though HTML 5 standard provides massive opportunities for tag writing, we recommend maintaining the writing style used by the biggest world companies: tags are written in the lower case, for example: `<body>`, `<html>`.

Also, all paired tags should be obligatorily closed.

The correct writing of the attributes which do not have a space:

- **`lang="en"`** — correct, double quotes are used for attribute value.
- **`lang='en'`** — correct, single quotes are used for attribute value.
- **`lang=en`** — correct, we may not use the quotes for values without space.

If an attribute value has quotes, the value is parenthesized.

Good manner during the layout is the usage of double quotes.

Basic Mistakes in Tag Writing

You can make a lot of mistakes when laying out the HTML document, let's analyze the typical ones:

- Mistakes in the HTML tags which can be divided into mistakes in the syntax itself and in absence of the angle brackets.
- Mistakes of the tag opening and closing, in other words, there is the opening tag, for example, `<p>`, but there is no `</p>` closing tag.
- Mistakes of the tags attachment logic, for example, `<p></p>` — in this case `</p>` is closed before the `` tag.
- The absence of the required tags: for example, tags `html`, `body`, `head` are required ones, and here is also the first mistake: the absence of the closing tags.
- The absence of the required attributes, for example, the required attribute for the hyper link is the *href*, for the image `src` and *alt*.
- The usage of out-of-date and inadmissible tags.
- Let's discuss how to deal with such mistakes:
- Learn the syntax of the HTML, documentation, W3C recommendations.
- Use a language sensitive editor, editor tips.
- Use a validator.

As we have already mentioned, the validator allows checking the validation of the destination document and the standard, `DOCTYPE` tag is used to indicate the standard.

In particular, the `<!DOCTYPEhtml>` tag indicates the usage of the modern HTML5 standard, however, in operation you may face other out-of-date formats:

- `<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">` — strict syntax of HTML 4.01 standard.
- `<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">` — transitional syntax of the HTML 4.01 standard.

If the validator does not find mistakes in operation, it is said that the HTML document is well-formed, in other words, it is syntactically valid. Pay attention that the expression well-formed is not similar to the *correct HTML which has errors*. In other words, a *well-formed* document simply meets a certain standard, but it is still important to use the validator to check it. We should aim to get a well-formed document during the layout.

Ancestors of HTML4/5 and XHTML: SGML and XML

The current rules of the HTML layout appeared as a long-term development and merging of two languages, SGML and XML, accompanied by the addition of new possibilities in accordance with the web specificity.

SGML is a standard generalized markup language; it was developed as a way to describe the documents which further should be processed with the help of machines. Initially, it was used in governmental and aerospace projects, and then, due to its simplicity and convenience, it began to be used to store information for further display and processing.

The SGML used a declaration that defined rules for writing a document and this allowed making flexible rules and changing them according to various requirements for the document.

XML, extendable markup language, was developed as a necessity to create a simple language to describe information; this language should be easily read by both human and machine. Language does not fixate the markup, it only determines the rules of its usage, that's why each project uses its own tags.

Nowadays XML is one of the standards to exchange information between applications.

Page Character Set Encodings and <meta> Tags

To display the HTML tag correctly, the browser should correctly read and display the document encoding. Modern browsers can automatically determine the character set encoding; however, in some cases we should do it manually. The change of the browser character set encoding is made on the client side. To display the page in the browser correctly, we need to indicate the page character set encoding in the HTML header. To do this, we use the multifunctional meta tag:

```
<meta charset="utf-8">
```

As we can see, this unpaired tag uses attributes.

Encoding is a compliance table between the finite set of alphabetic characters and the set sets of sequences. To simplify we can imagine the character set encoding as a way to represent letters and alphabet characters with the help of numerical codes. There are various character set encodings which are supported by both OS and browsers. For example, the encoding of the Windows-1251 for the Cyrillic alphabets, Shift JIS for Japanese, and so on.

UTF-8 encoding is initially developed for the operating systems due to the advantages, such as a big number of characters (up to 100,000), the possibility to use various languages in one encoding was spread to the webs and became one of the most popular encodings in the Internet.

Besides this attribute, the meta tag can also have:

- **Expires**

```
<meta http-equiv="expires" content="Sun, 01 Jan 2013  
07:01:00 GMT">
```

The expires attribute indicates the date after which the information will be considered out-of-date. This attribute can be used for pages of sales and discounts.

- **Refresh**

The refresh attribute allows setting the time period for the page rebooting.

Unit of measurements = seconds.

```
<meta http-equiv="refresh" content="30">
```

- **Author**

It allows setting the author of the HTML page.

```
<meta name="author" content=". . ." />
```

Now it is used rarely as Google uses its own social network called Google+ for the identification.

- **Copyright**

It allows setting the rights for the HTML document.

```
<meta name="copyright" content="MMMcorporationwebsite/>
```

As we have already mentioned, Google+ social network is often used to identify the author and to assign the copyright of the materials now.

- **Description**

The meta tag is used by a search engine to provide the results of the search in form of snippets. Since the snippet is limited

to 2–3 lines, it is not recommended to exceed 170 characters including spaces.

Description provides the description of the current web site page. All the pages have different unique description.

The rules of usage:

```
< meta name="description" content=
"Meta-tag content "description"">
```

■ Keywords

Meta tag which indicates the key word combinations describing the current page of the site. It is used by the search engine during the indexing.

```
<meta name="keywords" content="Keywords list">
```

Currently SEO analytics think that the usage of the keywords tag is not up-to-date as it is not used by search engines when ranking web sites. Many plug-ins for the SEO-optimization of the pages have already refused using the keywords field for the pages (for example, All-in-one-seo-pack).

Character Entity References

We have already listed the main mistakes of the layout, however, we have not mentioned another mistake which can appear when using the “<”, “/”, “ “ “ characters, which can be read by browser as control structures, these are special characters (brockets, clashes, quotes).

When you use these characters in a document text, a browser can think that they are a part of the tags and close or generate a tag, this will lead to mistakes in the layout. To avoid it, special

characters are put in the HTML in the form of the code set. There are three options of adding characters:

- **&mnemonic;** — the most simple to remember (for example, ©)
- **&#CODE10;** — symbol insertion with the help of decimal code (for example, &)
- **&#CODE16;** — the insertions of a special character with the help of hexadecimal code (for example, ®).

As we can see, all the sequences begin with the & character (ampersand) and end with the ; character (semicolon).

Let's consider several examples (see the full version of the table on the W3C website).

Character	Entity reference	10 code	16 code	Description
<	<	<	<	
>	>	>	>	
&	&	&	&	Ampersand
"	"	"	"	
	 	 	 	The nonbreaking space character. Words separated by it are not wrapped separately.
	­	­	­	The soft hyphen character. Parts of the word linked by it are wrapped if necessary. The — character is displayed on the wrap.
©	©	©	©	
®	®	®	®	
™	™	™	ࡊ	

Pay attention that in some cases layout designers copy a character to clipboard and add it to the HTML without reserved characters. This method is not cross-browser and can lead to the violation of the page layout.

Homework

1. Create an empty HTML page. Open it in an editor.
2. Set doctype of the page for the html5 page.
3. Add the following tags: html, head, body. Make sure that these tags are attached and closed correctly.
4. Add information on the page character set encoding to the head tag with the help of the meta tag.
5. Add the title of the page. Save changes. Open the page in a browser. Make sure that the title tag operates correctly. Open the web developer tools. See the structure of the document.
6. Add the header of the first level to the body tag. Put your name and surname there.
7. Add a horizontal line after the header.
8. Then add 2-3 paragraphs with information on the subjects you have learned in the STEP Computer Academy.
9. After the paragraphs with the text, add the images with the STEP Computer Academy logo.
10. Add the text link on the website of the Academy.
11. Change the added image so that you can go to the Academy website when clicking on the image.
12. Add the br tag to any paragraph.
13. Save the result. View it in the browser. Correct errors if necessary.
14. Validate the page. Look through the validation report.



Lesson 1

Introduction to the Web-Technologies. HTML Structure

© Aleksander Lushchenko.

© STEP IT Academy.

www.itstep.org

All rights to protected pictures, audio, and video belong to their authors or legal owners.

Fragments of works are used exclusively in illustration purposes to the extent justified by the purpose as part of an educational process and for educational purposes in accordance with Article 1273 Sec. 4 of the Civil Code of the Russian Federation and Articles 21 and 23 of the Law of Ukraine "On Copyright and Related Rights". The extent and method of cited works are in conformity with the standards, do not conflict with a normal exploitation of the work, and do not prejudice the legitimate interests of the authors and rightholders. Cited fragments of works can be replaced with alternative, non-protected analogs, and as such correspond the criteria of fair use.

All rights reserved. Any reproduction, in whole or in part, is prohibited. Agreement of the use of works and their fragments is carried out with the authors and other right owners. Materials from this document can be used only with resource link.

Liability for unauthorized copying and commercial use of materials is defined according to the current legislation of Ukraine.