Theory: Tags and attributes

7 minutes 0 / 5 problems solved

HTML documents consist of words that describe to the browser how to properly display a web page. Such instructions indicating which block should be shown are called **tags**.

When the browser receives an HTML document, it analyzes the tags and uses them to form elements that we see and possibly interact with. In other words, tags are like bricks that build up a web page. The current specification of HTML includes ~100 types of tags. Here is a complete list of all existing tags. Please, do not try to remember them now!

The syntax of a tag is very simple: the name of an element is written between symbols `<` and `>`. Tag names are case-insensitive, but it is considered good practice to write them in lowercase.

All tags in HTML language are divided into two main types: **paired** and **unpaired**; let's consider both in more detail.

Note: you can display all examples here on this site.

§1. Paired tags

The name speaks for itself. **Paired** HTML tags consist of ***two*** instructions: a **start tag**, which marks the beginning of the block, and an **end tag**, which looks exactly the same but with an additional slash `/` after `<`.

As an example, we will consider the `<p>` tag. It represents a paragraph of text:

```
<p>Some kind of text</p>
```

Here, `<p>` is a start tag, `Some kind of text` is the content, and `</p>` is an end tag.

You can see those tags as containers in which one can put (enclose) something. Paired HTML tags usually contain either other tags or some information, for instance, a text.

§2. Unpaired tags

The first thing to say about **unpaired tags** is that they have ***no content***. They are intended for the formation of graphic HTML elements or symbols on a page. Hence, unpaired HTML tags consist only of start tags.

Here is one possible example of an unpaired tag:

```
<hr>
```

Seeing this tag, the browser will draw a horizontal line.

Another example of an unpaired tag is `<br>` that defines a single line break.

§3. Nested tags

Tags can be nested to each other.

```
<p>You have learned HTML <b>tags</b> <br>Congratulations!</p>
```

Here, `<b>` is used to bold a word.

This HTML is rendered as:

Current topic:

Tags and attributes

Topic depends on:

✕ HTTP URL

✕ HTML basics

Topic is required for:

HTML page structure

Links

Code style

Table of contents:

↑ Tags and attributes

§1. Paired tags

§2. Unpaired tags

§3. Nested tags

§4. Tag attributes

Feedback & Comments

You have learned HTML **tags**
Congratulations!

Important that a nested tag must be closed before the external one.

The outer tag is called a **parent element** and the inner tag is called **a child element**.

§4. Tag attributes

To extend the capabilities of individual tags and manage them easier, attributes come in handy. **Attributes** are clarifications that help the browser correctly display a given tag.

The syntax of HTML attributes is also simple: each consists of **names** and **values**. The following example shows the syntax of attributes.

```
<a href="https://hyperskill.org">The link</a>
```

Here, the `<a>` tag means a link, `href` is the name of an attribute and `"https://hyperskill.org"` is the value. The attribute is assigned a value using the equals sign (`=`). HTML allows you to specify attribute values without quotes if they consist of one word. However, wrapping in quotes is good practice. The value of an attribute can be enclosed in double quotes `""`, or in single `''`.

Another important feature of HTML attributes syntax is that the attribute has to be written in angle brackets of the start tag:

```
<img src="image.png">
```

In this example, an image is added to a web page using an unpaired `<img>` tag. A link to the file is specified using the `src` attribute. The value of the attribute is equal to the reference to the desired image.

Again, there are many attributes out there: it might be worth your while checking them out.

Report a typo

405 users liked this theory. **6** didn't like it. **What about you?**

Start practicing

Comments (0) Hints (0) Useful links (1) Show discussion