# Week 2 Day 1 (sreda)

1. **HTML едитори**  
   NetBeans  
   NotePade++

Brakets

1. **основи на HTML**

## What is HTML?

HTML is the standard markup language for creating Web pages.

* HTML stands for Hyper Text Markup Language
* HTML describes the structure of Web pages using markup
* HTML elements are the building blocks of HTML pages
* HTML elements are represented by tags
* HTML tags label pieces of content such as "heading", "paragraph", "table", and so on
* Browsers do not display the HTML tags, but use them to render the content of the page

### Example Explained

* The <!DOCTYPE html> declaration defines this document to be HTML5
* The <html> element is the root element of an HTML page
* The <head> element contains meta information about the document
* The <title> element specifies a title for the document
* The <body> element contains the visible page content
* The <h1> element defines a large heading
* The <p> element defines a paragraph

## HTML Tags

HTML tags are element names surrounded by angle brackets:

<tagname>content goes here...</tagname>

* HTML tags normally come **in pairs** like <p> and </p>
* The first tag in a pair is the **start tag,** the second tag is the **end tag**
* The end tag is written like the start tag, but with a **forward slash** inserted before the tag name

**Tip:** The start tag is also called the **opening tag**, and the end tag the **closing tag**.

1. **интро во HTML**

You are now ready to learn the essence of HTML: elements.

Elements give structure to a HTML document and tells the browser how you want your website to be presented. Generally elements consists of a start tag, some content, and an end tag.

## "Tags"?

Tags are labels you use to mark up the begining and end of an element.

All tags have the same format: they begin with a less-than sign "<" and end with a greater-than sign ">".

Generally speaking, there are two kinds of tags - opening tags: <html> and closing tags: </html>. The only difference between an opening tag and a closing tag is the forward slash "/". You label content by putting it between an opening tag and a closing tag.

HTML is all about elements. To learn HTML is to learn and use different tags.

## Can you show me some examples?

Okay, the element em emphasis text. All text between the opening tag <em> and the closing tag </em> is emphasised in the browser. ("em" is short for "emphasis".)

**Example 1:**

<em>Emphasised text.</em>

Will look like this in the browser:

Emphasised text.

The elements h1, h2, h3, h4, h5 and h6 is used to make headings (h stands for "heading"), where h1 is the first level and normally the largest text, h2 is the second level and normally slightly smaller text, and h6 is the sixth and last in the hierarchy of headings and normally the smallest text.

**Example 2:**

<h1>This is a heading</h1>

<h2>This is a subheading</h2>

Will look like this in the browser:

# This is a heading

## This is a subheading

## So, I always need an opening tag and a closing tag?

As they say, there's an exception to every rule and in HTML the exception is that there are a few elements which both open and close in the same tag. These so-called empty elements are not connected to a specific passage in the text but rather are isolated labels, for example, a line break which looks like this: <br />.

## Should tags be typed in uppercase or lowercase?

Most browsers might not care if you type your tags in upper, lower or mixed cases. <HTML>, <html> or <HtMl> will normally give the same result. However, the **correct** way is to type tags in lowercase. So get into the **habit of writing your tags in lowercase**.

## Where do I put all these tags?

You type your tags in an HTML document. A website contains one or more HTML documents. When you surf the Web, you merely open different HTML documents.

If you continue to the next lesson in 10 minutes you will have made your first website.

## Web Browsers

The purpose of a web browser (Chrome, IE, Firefox, Safari) is to read HTML documents and display them.

The browser does not display the HTML tags, but uses them to determine how to display the document:

1. **Примери**

## 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>

**Note:** Only the content inside the <body> section (the white area above) is displayed in a browser.

## 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.

The <!DOCTYPE> declaration for HTML is:

<!DOCTYPE html>

## HTML Versions

Since the early days of the web, there have been many versions of HTML:

|  |  |
| --- | --- |
| **Version** | **Year** |
| HTML | 1991 |
| HTML 2.0 | 1995 |
| HTML 3.2 | 1997 |
| HTML 4.01 | 1999 |
| XHTML | 2000 |
| HTML5 | 2014 |

1. **HTML EDITORI**

# HTML Editors

## Write HTML Using Notepad or TextEdit

Web pages can be created and modified by using professional HTML editors.

However, for learning HTML we recommend a simple text editor like Notepad (PC) or TextEdit (Mac).

We believe using a simple text editor is a good way to learn HTML.

Follow the four steps below to create your first web page with Notepad or TextEdit.

## Step 1: Open Notepad (PC)

**Windows 8 or later:**

Open the **Start Screen** (the window symbol at the bottom left on your screen). Type **Notepad**.

**Windows 7 or earlier:**

Open **Start** > **Programs >** **Accessories >** **Notepad**

## Step 1: Open TextEdit (Mac)

Open **Finder > Applications > TextEdit**

Also change some preferences to get the application to save files correctly. In **Preferences > Format >** choose **"Plain Text"**

Then under "Open and Save", check the box that says "Ignore rich text commands in HTML files".

**Then open a new document to place the code.**

## Step 2: Write Some HTML

Write or copy some HTML into Notepad.

<!DOCTYPE html>  
<html>  
<body>  
  
<h1>My First Heading</h1>  
  
<p>My first paragraph.</p>  
  
</body>  
</html>



## Step 3: Save the HTML Page

Save the file on your computer. Select **File > Save as** in the Notepad menu.

Name the file **"index.htm"** and set the encoding to **UTF-8** (which is the preferred encoding for HTML files).



You can use either .htm or .html as file extension. There is no difference, it is up to you.

## Step 4: View the HTML Page in Your Browser

Open the saved HTML file in your favorite browser (double click on the file, or right-click - and choose "Open with").

The result will look much like this:



# HTML Head

[❮ Previous](https://www.w3schools.com/html/html_filepaths.asp)[Next ❯](https://www.w3schools.com/html/html_layout.asp)

## The HTML <head> Element

The **<head>** element is a container for metadata (data about data) and is placed between the <html> tag and the <body> tag.

HTML metadata is data about the HTML document. Metadata is not displayed.

Metadata typically define the document title, character set, styles, links, scripts, and other meta information.

The following tags describe metadata: <title>, <style>, <meta>, <link>, <script>, and <base>.

## The HTML <title> Element

The **<title>** element defines the title of the document, and is required in all HTML/XHTML documents.

The <title> element:

* defines a title in the browser tab
* provides a title for the page when it is added to favorites
* displays a title for the page in search engine results

A simple HTML document:

### Example

<!DOCTYPE html>  
<html>  
  
<head>  
  <title>Page Title</title>  
</head>  
  
<body>  
The content of the document......  
</body>  
  
</html>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_head_title)

## The HTML <style> Element

The **<style>** element is used to define style information for a single HTML page:

### Example

<style>  
  body {background-color: powderblue;}  
  h1 {color: red;}  
  p {color: blue;}  
</style>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_head_style)

## The HTML <link> Element

The **<link>** element is used to link to external style sheets:

### Example

<link rel="stylesheet" href="mystyle.css">

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_head_link)

**Tip:** To learn all about CSS, visit our [CSS Tutorial](https://www.w3schools.com/css/default.asp).

## The HTML <meta> Element

The **<meta>** element is used to specify which character set is used, page description, keywords, author, and other metadata.

Metadata is used by browsers (how to display content), by search engines (keywords), and other web services.

Define the character set used:

<meta charset="UTF-8">

Define a description of your web page:

<meta name="description" content="Free Web tutorials">

Define keywords for search engines:

<meta name="keywords" content="HTML, CSS, XML, JavaScript">

Define the author of a page:

<meta name="author" content="John Doe">

Refresh document every 30 seconds:

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

Example of <meta> tags:

### Example

<meta charset="UTF-8">  
<meta name="description" content="Free Web tutorials">  
<meta name="keywords" content="HTML,CSS,XML,JavaScript">  
<meta name="author" content="John Doe">

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_head_meta)

## Setting The Viewport

HTML5 introduced a method to let web designers take control over the viewport, through the <meta> tag.

The viewport is the user's visible area of a web page. It varies with the device, and will be smaller on a mobile phone than on a computer screen.

You should include the following <meta> viewport element in all your web pages:

<meta name="viewport" content="width=device-width, initial-scale=1.0">

A <meta> viewport element gives the browser instructions on how to control the page's dimensions and scaling.

The width=device-width part sets the width of the page to follow the screen-width of the device (which will vary depending on the device).

The initial-scale=1.0 part sets the initial zoom level when the page is first loaded by the browser.

Here is an example of a web page without the viewport meta tag, and the same web page with the viewport meta tag:

**Tip:** If you are browsing this page with a phone or a tablet, you can click on the two links below to see the difference.

[**Without the viewport meta tag**](https://www.w3schools.com/html/example_withoutviewport.htm) 

[**With the viewport meta tag**](https://www.w3schools.com/html/example_withviewport.htm) 

## The HTML <script> Element

The <script> element is used to define client-side JavaScripts.

This JavaScript writes "Hello JavaScript!" into an HTML element with id="demo":

### Example

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

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_head_script)

**Tip:** To learn all about JavaScript, visit our [JavaScript Tutorial](https://www.w3schools.com/js/default.asp).

## The HTML <base> Element

The <base> element specifies the base URL and base target for all relative URLs in a page:

### Example

<base href="https://www.w3schools.com/images/" target="\_blank">

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_head_base)

## Omitting <html>, <head> and <body>?

According to the HTML5 standard; the <html>, the <body>, and the <head> tag can be omitted.

The following code will validate as HTML5:

### Example

<!DOCTYPE html>  
<title>Page Title</title>  
  
<h1>This is a heading</h1>  
<p>This is a paragraph.</p>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_head_none)

**Note:**

W3Schools does not recommend omitting the <html> and <body> tags. Omitting these tags can crash DOM or XML software and produce errors in older browsers (IE9).

However, omitting the <head> tag has been a common practice for quite some time now.

## HTML head Elements

|  |  |
| --- | --- |
| **Tag** | **Description** |
| [<head>](https://www.w3schools.com/tags/tag_head.asp) | Defines information about the document |
| [<title>](https://www.w3schools.com/tags/tag_title.asp) | Defines the title of a document |
| [<base>](https://www.w3schools.com/tags/tag_base.asp) | Defines a default address or a default target for all links on a page |
| [<link>](https://www.w3schools.com/tags/tag_link.asp) | Defines the relationship between a document and an external resource |
| [<meta>](https://www.w3schools.com/tags/tag_meta.asp) | Defines metadata about an HTML document |
| [<script>](https://www.w3schools.com/tags/tag_script.asp) | Defines a client-side script |
| [<style>](https://www.w3schools.com/tags/tag_style.asp) | Defines style information for a document |

# Web standards and validation

In this lesson, you will get a little more theoretical knowledge on HTML.

## What more is there to know about HTML?

HTML can be coded in many different ways. And browsers can read HTML in just as many ways. You could say that **HTML has many dialects**. That is why some websites look different in different browsers.

**There have been attempts to make a common standard of HTML through the**[**World Wide Web Consortium (W3C)**](http://www.w3.org/) founded by Tim Berners-Lee (yep! the great guy who invented HTML). But it has been a long and tough road.

In the old days - when browsers where something you had to pay for - Netscape was the dominate browser. Back then, the most supported HTML standard where called 2.0 and later 3.2. But with a market share of over 90% Netscape did not have to - and did not - care much about common standards. On the contrary, Netscape invented their own strange elements, which did not function in other browsers.

For many years Microsoft almost completely ignored the Internet. After a while they took up the competition with Netscape and introduced a browser. The first versions of Microsoft's browser, Internet Explorer, were not any better than Netscape at supporting the HTML standards. But Microsoft chose to give away their browser for free (always a popular thing to do) and Internet Explorer soon became the most popular browser.

From version 4 and 5 Microsoft aimed to support more and more of the HTML standards from W3C. Netscape did not manage to develop a new version of their browser and continued to distribute the outdated version 4.

The rest is history. Today the HTML standards are called 4.01 and XHTML. Now it is Internet Explorer that has a market share of over 90%. Internet Explorer still has its own strange elements but it also supports the W3C HTML standards. And so do all of the other browsers, such as Mozilla, Opera and Netscape.

**So, when you code HTML following the W3C standards, you make websites that can be seen in all browsers** - both now and in the future. And luckily, **what you have learned in this tutorial is a new and stricter and cleaner version of HTML called XHTML**.

## How do I tell which version is used?

With all the different types of HTML you need to tell the browser which "dialect" your HTML is in, in your case XHTML. To do that, you use a Document Type Declaration. The Document Type Declaration is always written in the top of the document:

**Example 1:**

**<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"**

**"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">**

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

<head>

<title>Title</title>

</head>

<body>

<p>text text</p>

</body>

</html>

Besides the Document Type Declaration (the first line in the example above), which tells the browser that you want to write XHTML, you also need to insert some extra information in the html tag with the two attributes xmlns and lang.

xmlns is short for "XML-Name-Space" and should always have the value **http://www.w3.org/1999/xhtml**. That is all you need to know. But if you have a big hunger for complicated knowledge you can read more about namespaces on [W3C's website](http://www.w3.org/TR/1999/REC-xml-names-19990114/).

In the lang attribute you state which language the document is written in. For this the [ISO 639 standard](http://www.w3.org/WAI/ER/IG/ert/iso639.htm#2letter) is used, which lists codes for all the languages in the world. In the example above the language are set to English ("en").

With a DTD the browser knows exactly how it should read and show your HTML. Hence, use the example above as template for all your future HTML documents.

The DTD is also important when you want to validate your pages.

## Validate? Why and how should I do that?

Insert a DTD in your pages and you can always check your HTML for errors by using [W3C's free validator](http://validator.w3.org/).

To test this out, make a page and upload it to the Internet. Now, go to [validator.w3.org](http://validator.w3.org/) and type the address (the URL) of your page and validate it. If your HTML is correct you will get a congratulations message. Otherwise you will get an error report telling you exactly what and where you have done something wrong. Make some errors on purpose to see what happens.

The validator is not just helpful to locate an error. Some browsers try to compensate for lack of skills among web developers by trying to fix errors in the HTML and showing the page as they guess it should look. With such browsers, you might never see an error in your own browser. However, other browsers might guess differently or not show the page at all. The validator can help you find errors you did not even know existed.

Always validate your pages to be sure they will always be shown correctly.