

## **Exercises 10 and 11 - web page in HTML and uploading to github**

### Introduction:

In this exercise, we create the web content, as a carrier of multimedia on the web. The multimedia content that we want to transfer over the web refers to text, image, video and sound. Web pages are created using HTML (Hypertext Markup Language), which describes the type of content we want to display. We code HTML documents in different text editors and save documents as .html files.

The styling of HTML content is defined by the language CSS (Cascading Style Sheets), which determines the sizes, positions, colors and other stylistic characteristics of the content.

In the instructions (below) all the HTML elements that we will use in the exercise to create a web document are defined.

The video instructions describe how to create a multimedia HTML document in different editors (all instructions are in one video). Some popular text editors are suggested, and you can use some others if you are used to working in them.

The second video shows you how to create a free account on the Github platform where you will create your website and upload all the website files created locally on your computer.

### [Video instructions for creating web content in different editors:](#)

00:00 - Notepad++

03:15 – Atom

06:25 - Sublime Text

09:15 – Dreamweaver

### [Video instructions for creating a github account and uploading/publishing your web page](#)

The basic structure of an HTML document:

```
<!DOCTYPE html>
<html> - initial "tag" (label or element)
<head>
<title>document title</title>
</head>
<body> ...everything that is displayed in the browser window... </body>
</html> - final "tag"
```

`<!DOCTYPE html>` - the declaration - tells which version of the html language will be used in the document, so that the browser can display the page correctly. Most marks (or tags) come in pairs, at the beginning and at the end of the entity we want to mark. The initial `<html>` tag (or tag) is mandatory at the beginning of the document, and at the end of the document we close the whole with the final tag `</html>`.

Within the `<html>` document there are two main units:

`<head></head>` and `<body></body>`

Head contains the functional part of the page, and in `<body>` we insert the content we want to see on the page, such as text, images, etc...

Html elements that do not come in a pair are marked with one tag, e.g. `<img />` or `<br/>`

HTML tags are more precisely defined by the attributes that we insert inside the opening tag. In general, the attribute marking looks like this:

```
<tag attribute1="value" attribute2="value"> content </tag>
```

In a text editing software such as Notepad, Notepad++ or a coding program such as Adobe Dreamweaver, Sublime Text, Atom or some other, we will create HTML documents with the following elements:

- Title
- text
- links
- navigation list
- images
- video
- sound

Once we have inserted the content into the page, we will style that content using the CSS (Cascading Stylesheets) styling language.

## TEXT MARKUP

### HTML HEADINGS

The element tag for headings in HTML is denoted by the letter "h" with the corresponding number indicating the level of the heading - subtitle, sub-subheading, etc.

```
<h1>First-level heading</h1>
<h2>Second-level heading</h2>
<h3>Third-level heading</h3>
<h4>Fourth-level heading</h4>
<h5>Fifth-level heading</h5>
<h6>Sixth-level heading</h6>
```

### HTML PARAGRAPH

A paragraph is a basic text tag into which larger blocks of text are inserted - paragraphs.

```
<p>This is a basic paragraph.</p>
```

### HTML LINK

A link is denoted by the <a> tag, which, in order to function, must contain the text we want to click on and the address to which the link will take us. The address (location) to which the link takes us is defined by the href attribute

```
<a href="address or document to which the link takes us">This is the text of the link
I will click on</a>
```

Link to the existing web address:

```
<a href="https://www.w3schools.com">here i will learn html and css</a>
```

Link to local document:

```
<a href="location/exercise1.html">This is a link to another html document</a>
```

```
<a href="locija/zadaca.pdf">This is my homework in pdf</a>
```

### OTHER TEXT MARKS

Within the paragraph, we can emphasize the text in different ways:

```
<b></b> - bold text
<i></i> - italic
<mark></mark> - highlighting with a marker
<q></q> - Quote - adds quotation marks before and after the text
<blockquote></blockquote> - indents the text by some amount
<br/> - places the content in a new line
```

Examples:

<p>This is it <b>bold</b> text, <br/>and this is <i>italic</i> text in a new line,  
and I will <mark>mark this text!</mark></p>

<p>This is a short citation

<q> Everything you can imagine is real.</q> -Picasso.</p>

<p>Ovo je blok citat: </p><blockquote> Over the wintry <br/> Forest, winds howl in  
rage<br/> With no leaves to blow. <br/> <small>- Natsume Sōseki</small> </blockquote>

<p>Ovo je tekst kojem ću <span style="color:red">promijeniti stil</span>unutar  
paragrafa.</p>

## LISTS and NAVIGATION LISTS

A list is text that is organized into items. Ordered lists <ol> – when we mark content where order is important  
Unordered lists <ul> – when we mark content where order is not important

Each list has items that are marked with the <li> (list item) element

Example:

<p>Fruits by vitamin C

content:<p><ol><li>kiwi</li><li>orange</li><li>strawberry</li></ol>

<p>Shopping list:</p><ul><li>kiwi</li><li>orange</li><li>strawberry</li></ul>

## NAVIGATION LIST

A list (usually unordered) whose items are links, and is located inside the <nav> element that marks the  
navigation area.

<nav>

<p>Navigation list:</p><ul>

<li><a href="prvopredavanje.pdf">first item</a></li>

<li><a href="Drugopredavanje.pdf">second item</a></li>

<li><a href="trecepredavanje.pdf">third item</a></li>

</ul>

</nav>

## TAGGING IMAGES

Images are marked with the <img /> element. In order for the browser to know which image to display,  
<img /> must contain an attribute that points to the source of the image. This attribute is called src  
(source):



The images we insert can be static, usually in jpg or png formats, or moving, such as GIFs.

## TAGGING VIDEO FILES

```
<video width="" height="" controls>
<source src="mojvideo.mp4" type="video/mp4"/>
<source src="mojvideo.ogg" type="video/ogg"/>
Your browser does not support this video file format
</video>
```

## VIDEO EMBEDDED THROUGH THE YOUTUBE PLATFORM

When we want to embed a video that has already been uploaded to YouTube, we must first find the address of that video on YouTube. Click on Share below the video. Select the <> EMBED option and then copy the part of the code that looks something like this:

```
<iframe width="560" height="315" src="https://www.youtube.com/embed/h0kXpKW0GC4"
frameborder="0" allow="accelerometer; autoplay; encrypted-media; gyroscope; picture-
in-picture" allowfullscreen></iframe>
```

Paste this code into our html code where we want to display the video.

## TAGGING AUDIO FILES

```
<audio controls>
<source src="mojzvuk.mp3" type="audio/mp3"/>
<source src="mojvideo.ogg" type="audio/ogg"/>
Your browser does not support this audio file format
</audio>
```

## STYLING OF THE CONTENT

It is done through the language CSS (Cascading Style Sheets) which is connected to the HTML document that we want to style. CSS can be directly embedded in the HTML code or defined in a separate document.

CSS as an HTML attribute - we can insert the style attribute into any html tag, which will contain properties for formatting the content of the html tag. Syntax:

```
<label style="property1:value; property2:value;...">
```

```
<p style="color:green;">This is the text that I will <span  
style="color:white;background-color:black;">change the style</span> within the  
paragraph.</p>
```

CSS inside the `<style></style>` tags - we insert it in the `<head>` of the html document:

Syntax:

```
element {property1: value; property2: value;}  
secondelement { property1 : value ; property2: value;}
```

When we want to apply a style to all elements (e.g. all paragraphs in a document) then in the CSS file we will use the tag before the style definition `p`, `img` ...

## CLASS

A special type of style that we apply to individual tags:

```
.classname {property1: value; property2: value;}
```

Example:

`.red {color: red;}` - in the CSS file we define a class (there is a dot in front of the class name)

`<p class="red">this is red text</p>` - in HTML we call the class within the class attribute (without the dot)

```
<head> ...  
<style>  
p {color: green;}  
span { color:white; background-color:black;}  
.red {color: red; }  
</style>  
</head>
```

CSS as a separate document – we create a new document `style.css` and define the styles in it:

```
p {color: green;}  
span { color:white;background-color:black;}  
.red {color:red;}
```

In the html `<head>` section, we add a link to the `style.css` document to link elements to their style:

```
<link rel="stylesheet" href="stil.css">
```

## STYLES WE WILL USE IN THE EXERCISE

Width and height of an element: width: 200px; height: 100px; - it must be stated with the number and unit of measurement (without spaces)

Text coloring:

color: color; - we can define the color in several ways - more about it at this link.

Examples for red:

```
color: red;
```

```
color: #FF0000;
```

```
color: rgb(255, 0, 0);
```

```
color: rgba(255, 0, 0, 0.5); - the last value is opacity - 0.5 = 50% visibility
```

Background:

```
background-color: color; opacity: 0.3 (30% visibility = 70% transparency)
```

```
background-image: url("my picture.jpg");
```

Outlines and borders:

```
border: 2px solid black; - defines a border around the element with a thickness of 2px, solid line, black color.
```

```
border-radius: 5px; - rounded edge of the frame
```

Inner and outer margins:

inner space between the edge of the element and the content

```
padding: 10px;
```

outer space between elements

```
margin: 10px;
```

Font selection:

It is common practice to specify multiple fonts as alternatives; if the user of the page cannot see the first font listed, the next one is used to display the text, etc....

```
font-family: Arial, Helvetica, sans-serif;
```

```
font-size: 14px; (other units of measurement can be used: em, pt, mm...)
```

```
line-height: 16px; - line spacing
```

Links: For each link state, a special color, background, text size, etc. can be defined...

```
a:link - normal link state
```

```
a:visited - link style when already visited
```

a: hover - link style when hovered over

a: active - link style when clicked

## TASK

Create an index.html page in which you will use the following content:

- First and second level titles (Here you will put your name and surname and a description of the page)
- a navigation list with at least three links leading to:
  - **one web page-html document\***
- one pdf document
- a paragraph of text "About you" in which you will use different types of text marking (bold, italic, quote...)
- Two lists (ordered and unordered) with headings, e.g. "music I listen to" or "books I read"

**\*other html document** that will contain multimedia elements:

- a few of your images (they should be optimized to smaller dimensions, which you will use on the page)
- a video (you can use a video from the exercise)
- a video from YouTube
- sound (mp3 file that does not exceed 30 seconds)

Style these documents according to the instructions from this exercise or add other style elements as desired. You can learn more about HTML and CSS at [www.w3schools.com](http://www.w3schools.com)

Organize all the files you use into logical folders and upload them all together to your Github account. The file and folder structure on Github must be identical to the one on your local computer.

Submit your webpage address via email.