



# HTML Trainer Materials

## Subchapter 1 – The basics of HTML

WP3: Code4SP Training Materials

Prepared by: 



CITIZENS  
IN POWER



Center for Social  
Innovation



social  
hackers  
academy



Escola Profissional de Espinho



# Subchapter 1: The basics of HTML

Co-funded by the  
Erasmus+ Programme  
of the European Union



# HTML Introduction

- HTML (Hyper Text Markup Language) is the **standard markup language for creating Web pages**;
- It describes the **structure of a Web page**
- It consists of a **series of elements**
- Its elements tell the browser **how to display the content**
- Its **elements label pieces of content** such as "this is a heading", "this is a paragraph", "this is a link", etc.

Shall we give it a try?

[https://www.w3schools.com/html/tryit.asp?filename=tryhtml\\_intro](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_intro)





## What can be done with HTML?

- Publish documents with text, images, lists, tables, etc.
- Access web resources (images, videos or other HTML document) via hyperlinks.
- Create forms to collect user inputs (name, e-mail address, etc.)
- Add images, videos, sound clips, flash movies, applications and other HTML documents directly inside an HTML document.
- Create offline version of a website that works without internet.
- Store data in the user's web browser and access later on.
- Find the current location of your website's visitor.



# HTML Introduction

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>
<h1>My First Heading</h1>
<p>My first paragraph.</p>
</body>
</html>
```

→ this declaration defines that this is an HTML document.

→ the root element of an HTML page

→ contains meta information about the HTML page

→ specifies a title for the HTML page

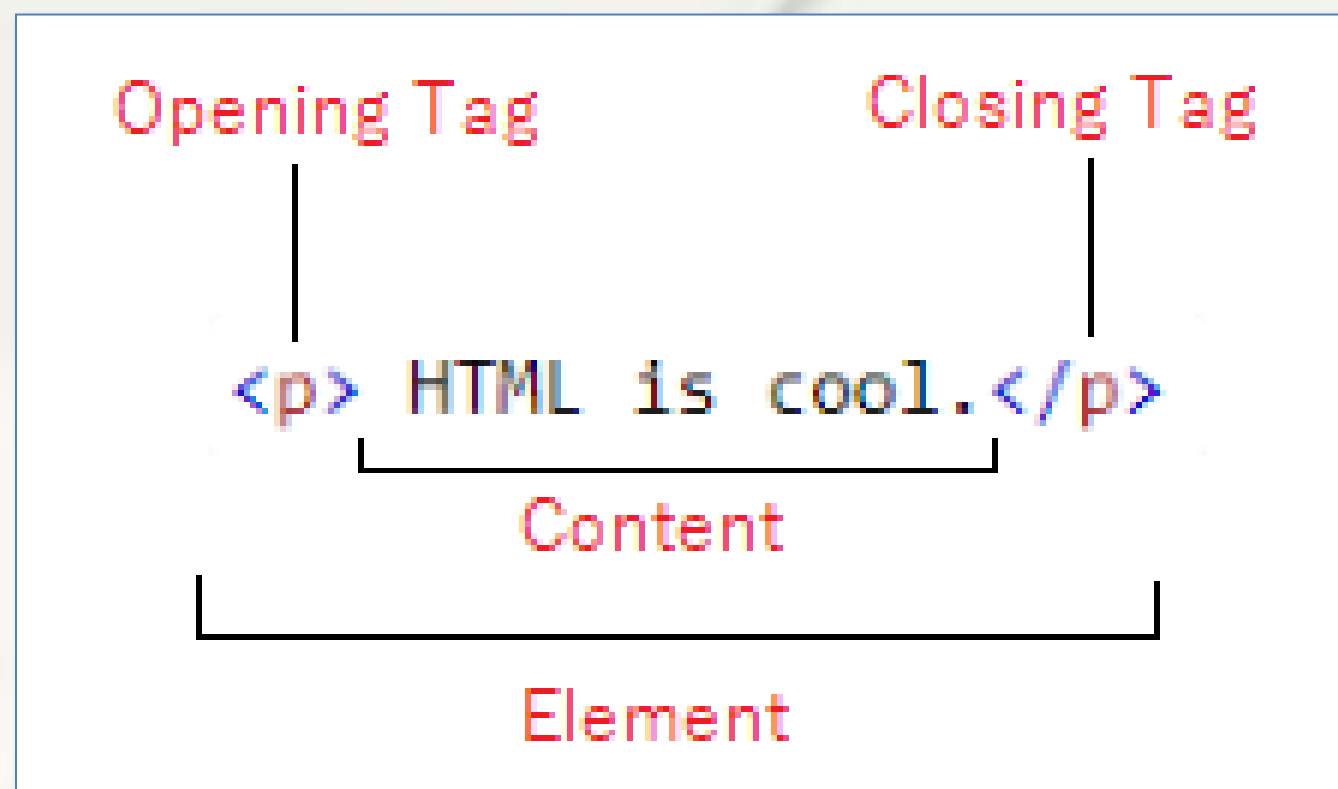
→ defines the document's body, and is a container for all the visible contents, such as headings, paragraphs, images, hyperlinks, tables, lists, etc.

→ defines a large heading

→ defines a paragraph

# Anatomy of a HTML element

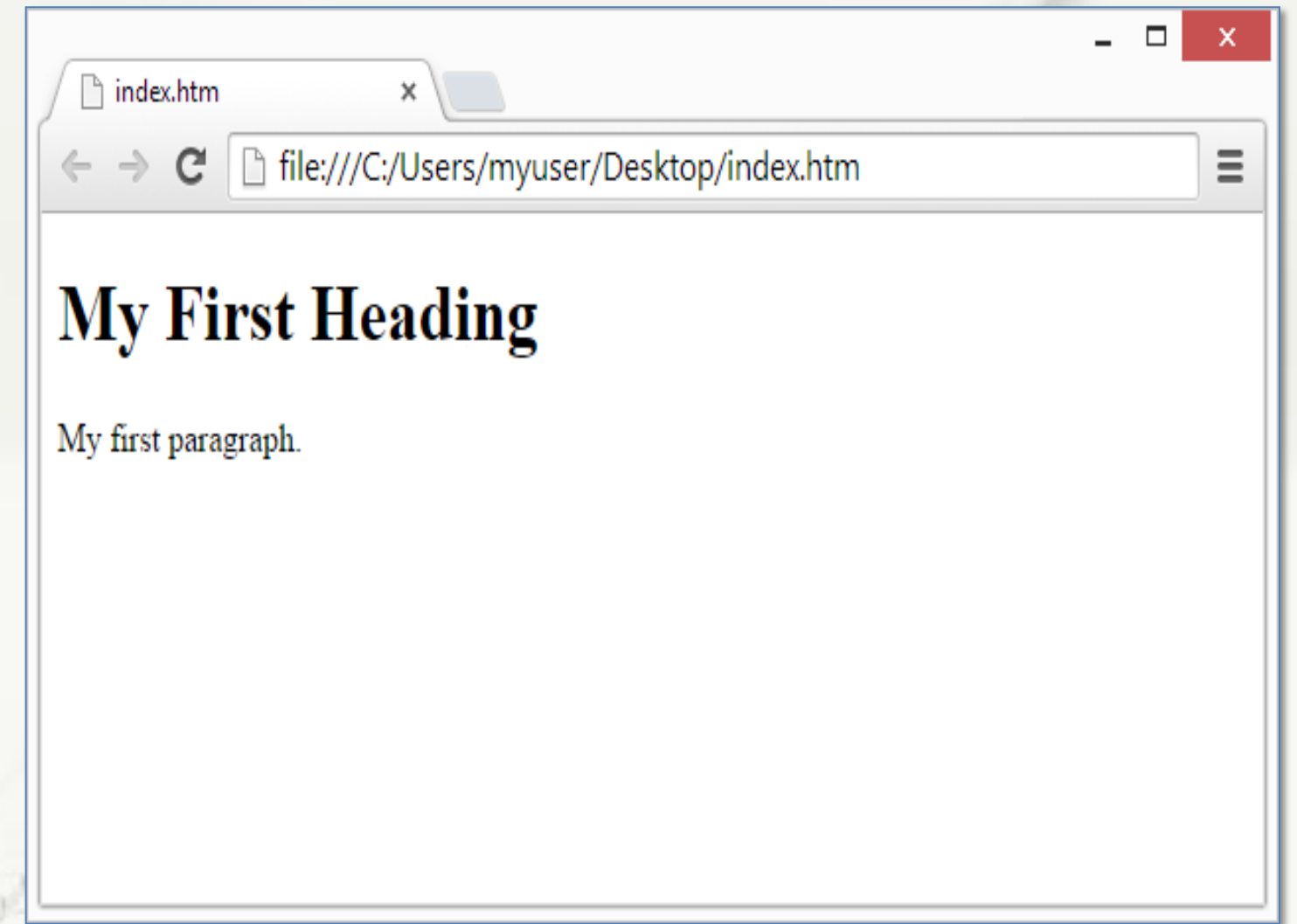
- **The opening tag:** the name of the element (in this example, *p* for paragraph), wrapped in opening and closing angle brackets. This opening tag marks where the element begins or starts to take effect. In this example, it comes first, at the start of the paragraph text.
- **The content:** This is the content of the element. In this example, it is the paragraph text.
- **The closing tag:** This is the same as the opening tag, except that it includes a forward slash before the element name. This marks where the element ends. Failing to include a closing tag is a common beginner error that can produce peculiar results.
- The **element** is the opening tag, followed by content, followed by the closing tag.





# Web Browsers

- The purpose of a web browser (Chrome, Edge, Firefox, Safari) is to read HTML documents and display them properly.
- A browser does not display the HTML tags. It uses them to determine how to display the document.



# 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>
```

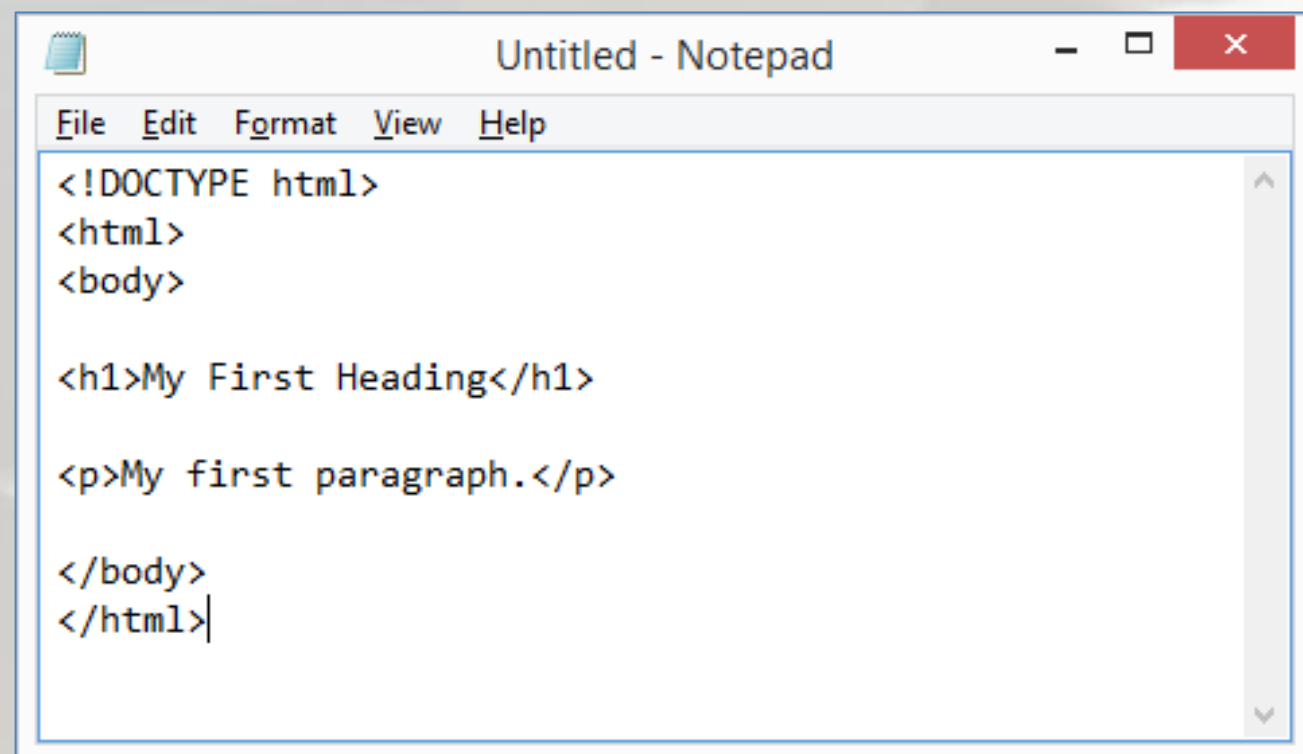
The content inside the `<body>` section (the white area) will be shown in a browser. The content inside the `<title>` element will be presented in the browser's title bar or in the page's tab.





# HTML Editors

- A simple text editor, like Notepad, is all one needs to learn and write code on HTML, even though web pages can be created and modified by using professional HTML editors. Let's create our first web page with Notepad.
- **STEP 1:** Open Notepad (PC)
- **STEP 2:** Write some HTML, as follows:
- **STEP 3:** Save the HTML Page and name the file "index.html"
- **STEP 4:** Open the saved HTML file in the browser.

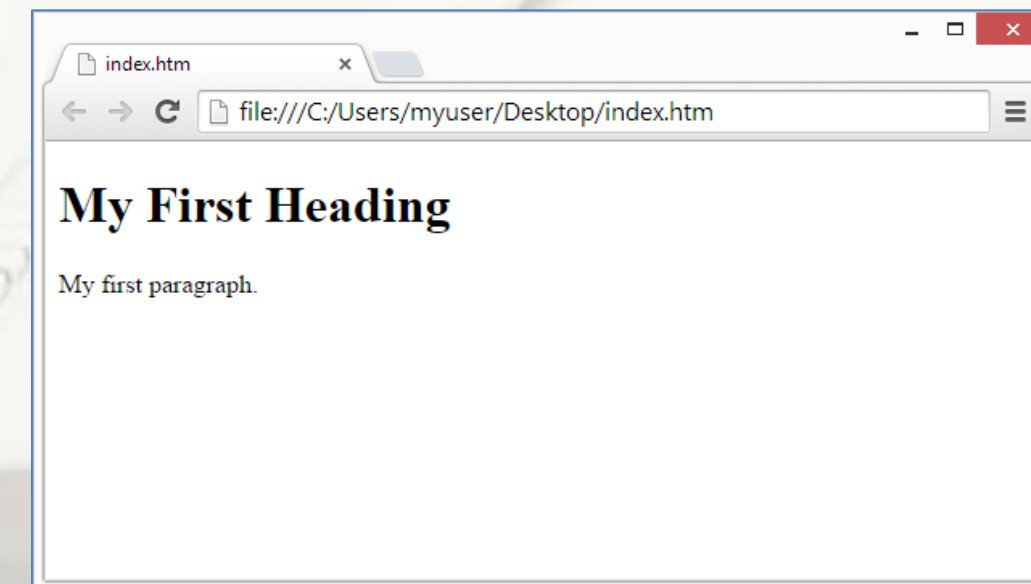


```
File Edit Format View Help
<!DOCTYPE html>
<html>
<body>

<h1>My First Heading</h1>

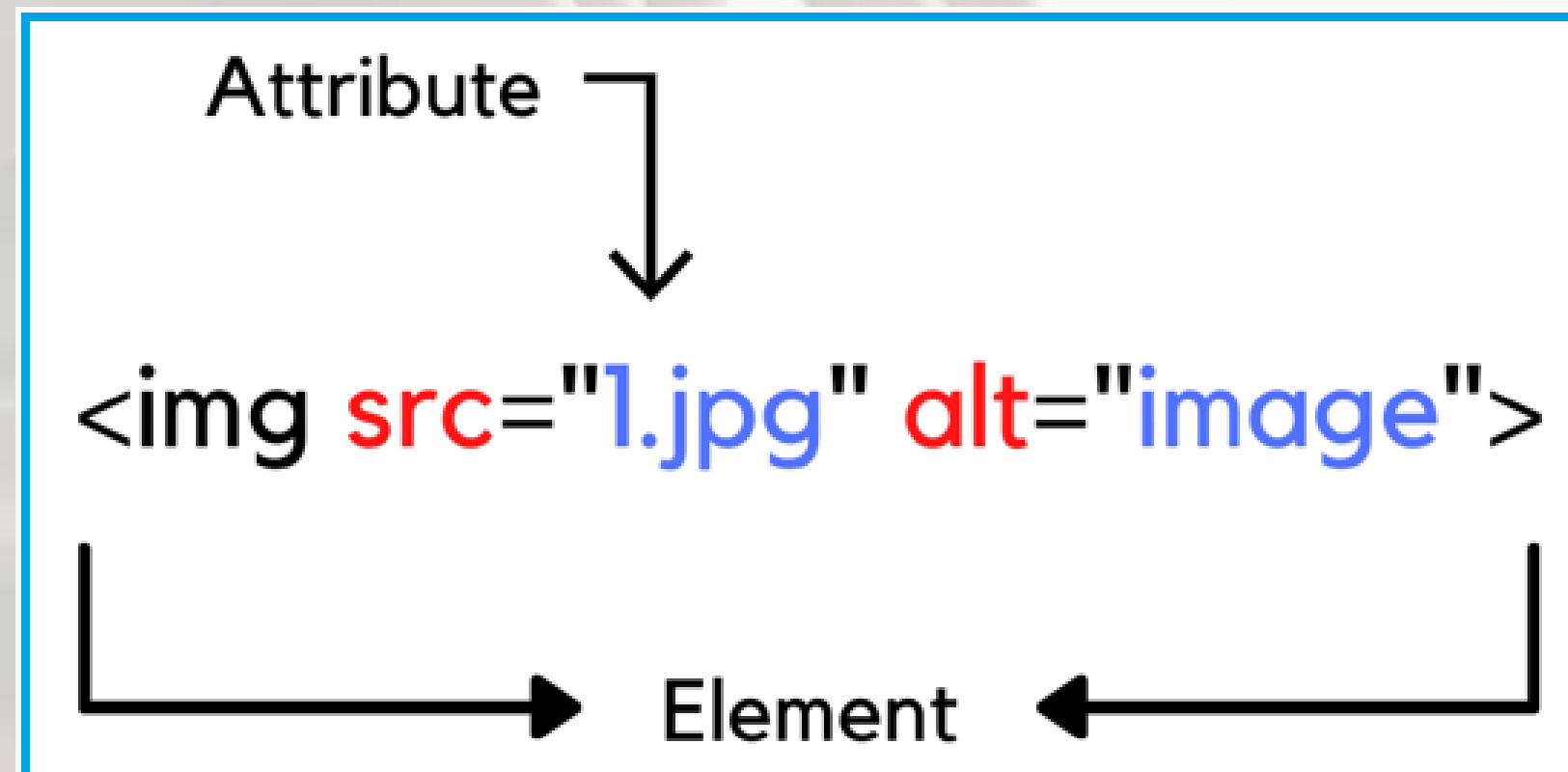
<p>My first paragraph.</p>

</body>
</html>
```



# HTML Attributes

- HTML elements can have **attributes**
- Attributes deliver **additional information** about elements
- Attributes are always specified in **the start tag**
- Attributes usually come in name/value pairs, as: **name="value"**



# HTML Attributes

## List of common attributes:

- **href** – the `<a>` tag defines a hyperlink. The **href** attribute specifies the URL of the page the link goes to, as follows:

```
<a href="https://www.w3schools.com">Visit W3Schools</a>
```

- **src** – the `<img>` tag is used to embed an image in an HTML page. The **src** attribute specifies the path to the image to be displayed, as seen below:

```

```

- **width and height** – the `<img>` tag should also comprise the width and height attributes, which stipulates the width and height of the image (in pixels):

```

```



# HTML Attributes

## List of common attributes:

**alt** – the required alt attribute for the `<img>` tag specifies an alternate text for an image, if the image cannot be displayed for some reason. This can be due to slow connection, or an error in the `src` attribute, or if the user uses a screen reader.

```

```

If we try to display an image that does not exist, the value of the alt attribute will be displayed instead, as follows:

 Girl with a jacket

# HTML Attributes

## List of common attributes:

- **style** – the style attribute is used to add styles to an element, such as color, font, size, and more.

```
<p style="color:red;">This is a red paragraph.</p>
```

<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;  &lt;h2&gt;The style Attribute&lt;/h2&gt; &lt;p&gt;The style attribute is used to add styles to an element, such as color: &lt;/p&gt;  &lt;p style="color:red;"&gt;This is a red paragraph.&lt;/p&gt;  &lt;/body&gt; &lt;/html&gt;</pre>	<h3>The style Attribute</h3> <p>The style attribute is used to add styles to an element, such as color:</p> <p><b>This is a red paragraph.</b></p>
---	--



# HTML Attributes

## List of common attributes:

- **lang** – the lang attribute should always be included inside the `<html>` tag, to declare the language of the Web page. This is meant to support search engines and browsers. The example below stipulates English as the language in use:

```
<!DOCTYPE html>  
<html lang="en">  
<body>  
...  
</body>  
</html>
```



# HTML Attributes

## List of common attributes:

- **title** – this attribute describes some additional information about an element. Its value will be displayed as a tooltip when the mouse pointer goes over the element, as follows:

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

<h2 title="I'm a header">The title Attribute</h2>

<p title="I'm a tooltip">Mouse over this paragraph, to display the title
attribute as a tooltip.</p>

</body>
</html>
```

### The title Attribute

I'm a header

Mouse over this paragraph, to display the title attribute as a tooltip.

# HTML Headings

- HTML headings are titles or subtitles that one wants to display on a webpage.
- HTML headings are delineated with the `<h1>` to `<h6>` tags. `<h1>` identifies the most important heading. `<h6>` outlines the least important heading. `<h1>` headings should be used for main headings, followed by `<h2>` headings, then the less important `<h3>`, and so on.
- Headings are of utmost importance since search engines use them to index the structure and content of webpages. Users often browse a page by its headings. It is important to use headings to exhibit the document structure.
- It is important to state that, by definition, browsers automatically add a margin before and after a heading.

<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;  &lt;h1&gt;Code4SP 1&lt;/h1&gt; &lt;h2&gt;Code4SP 2&lt;/h2&gt; &lt;h3&gt;Code4SP 3&lt;/h3&gt; &lt;h4&gt;Code4SP 4&lt;/h4&gt; &lt;h5&gt;Code4SP 5&lt;/h5&gt; &lt;h6&gt;Code4SP 6&lt;/h6&gt;  &lt;/body&gt; &lt;/html&gt;</pre>	<p><b>Code4SP 1</b></p> <p><b>Code4SP 2</b></p> <p><b>Code4SP 3</b></p> <p><b>Code4SP 4</b></p> <p><b>Code4SP 5</b></p> <p><b>Code4SP 6</b></p>
--	---

**`<h1>Heading 1</h1>`**

**`<h2>Heading 2</h2>`**

**`<h3>Heading 3</h3>`**

**`<h4>Heading 4</h4>`**

**`<h5>Heading 5</h5>`**

**`<h6>Heading 6</h6>`**

# HTML Paragraphs

<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;  &lt;p&gt;Code4SP helps me to code.&lt;/p&gt; &lt;p&gt;I love Code4SP.&lt;/p&gt; &lt;p&gt;Coding is so great!&lt;/p&gt;  &lt;/body&gt; &lt;/html&gt;</pre>	<p>Code4SP helps me to code.</p> <p>I love Code4SP.</p> <p>Coding is so great!</p>
---	--

- A paragraph constantly starts on a new line and is typically a block of text. It is defined by the HTML `<p>` element and, like headings, browsers automatically add some margin before and after a paragraph.



# HTML Display

- One cannot be sure how HTML will be presented, as it can vary from screen to screen. With HTML, the display cannot be changed by adding extra spaces or extra lines in the HTML code.
- The browser will automatically remove any extra spaces and lines when the page is displayed, as seen in the following example:

<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;  &lt;p&gt; This paragraph contains a lot of lines in the source code, but the browser ignores it. &lt;/p&gt;  &lt;p&gt; This paragraph contains      a lot of spaces in the source  code, but the      browser ignores it. &lt;/p&gt;  &lt;p&gt; The number of lines in a paragraph depends on the size of the browser window. If you resize the browser window, the number of lines in this paragraph will change. &lt;/p&gt;  &lt;/body&gt; &lt;/html&gt;</pre>	<p>This paragraph contains a lot of lines in the source code, but the browser ignores it.</p> <p>This paragraph contains a lot of spaces in the source code, but the browser ignores it.</p> <p>The number of lines in a paragraph depends on the size of the browser window. If you resize the browser window, the number of lines in this paragraph will change.</p>
--	--



## HTML Line Breaks

- The HTML `<br>` element defines a line break. It is an empty tag, which means that it has no end tag.
- `<br>` should be used if one wants a new line without starting a new paragraph:

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

<p>Code4SP really is<br>an
amazing<br>project.</p>

</body>
</html>
```

Code4SP really is  
an amazing  
project.

# HTML Styles

- The HTML style attribute is used to add styles to an element, such as color, font, size, etc.
- To set the style of an HTML element, the style attribute must be used. It has the following syntax (it should be noted that *property* and *value* are CSS features, to be learned later).
- `<tagname style="property:value;">`

# HTML Styles

## Background Colour

- The CSS *background-color* property specifies the background colour for an HTML element.

```
<!DOCTYPE html>
<html>
<body style="background-color:green;">

<h1>Code4SP</h1>
<p>Coding for Social Promotion.</p>

</body>
</html>
```

**Code4SP**

Coding for Social Promotion.



# HTML Styles

## Text Colour

- The CSS *color* property outlines the text colour for an HTML element:

```
<!DOCTYPE html>  
<html>  
<body>  
  
<h1 style="color:blue;">Code4SP</h1>  
<p style="color:red;">Coding for Social  
Promotion.</p>  
  
</body>  
</html>
```

**Code4SP**

Coding for Social Promotion.

# HTML Styles

## Fonts

- The CSS *font-family* property defines the font to be used for an HTML element:

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

<h1 style="font-
family:verdana;">Code4SP</h1>
<p style="font-family:courier;">Coding
for Social Promotion.</p>

</body>
</html>
```

## Code4SP

Coding for Social Promotion.

# HTML Styles

## Text Size

- The CSS font-size property identifies the text size for an HTML element:

<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;  &lt;h1 style="font-size:300%;"&gt;CODE4SP&lt;/h1&gt; &lt;p style="font-size:160%;"&gt;Coding for Social Promotion.&lt;/p&gt;  &lt;/body&gt; &lt;/html&gt;</pre>	<p><b>CODE4SP</b></p> <p>Coding for Social Promotion.</p>
---	---

# HTML Styles

## Text Alignment

- The CSS *text-align* feature defines the horizontal text alignment for an HTML element:

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

<h1 style="text-align:center;">CODE4SP</h1>
<p style="text-align:center;">Coding
for Social Promotion.</p>

</body>
</html>
```

## CODE4SP

Coding for Social Promotion.



# HTML Text Formatting

- HTML comprises various elements for defining text with a special implication (bold, italic, subscript, superscript, etc.).

# HTML Text Formatting

The following are the **HTML** formatting elements:

Property	Outcome	Definition	Example
<code>&lt;b&gt;</code>	Bold text	The HTML <code>&lt;b&gt;</code> element specifies bold text, without any extra importance.	<b>Bold text</b>
<code>&lt;strong&gt;</code>	Important text	The HTML <code>&lt;strong&gt;</code> element describes text with strong importance. The content inside is usually displayed in bold.	<b>Important text</b>
<code>&lt;i&gt;</code>	Italic text	The HTML <code>&lt;i&gt;</code> element defines a part of text in an alternate voice or mood. The content inside is typically displayed in italic.	<i>Italic text</i>

# HTML Text Formatting

The following are the **HTML formatting elements**:

Property	Outcome	Definition	Example
<code>&lt;em&gt;</code>	Emphasized text	The HTML <code>&lt;em&gt;</code> element defines emphasized text. The content inside is typically displayed in italic.	<i>Emphasized text</i>
<code>&lt;mark&gt;</code>	Marked text	The HTML <code>&lt;mark&gt;</code> element defines text that should be marked or highlighted.	<b>Marked text</b>
<code>&lt;small&gt;</code>	Smaller text	The HTML <code>&lt;small&gt;</code> element defines smaller text.	Smaller text

# HTML Text Formatting

The following are the **HTML formatting elements**:

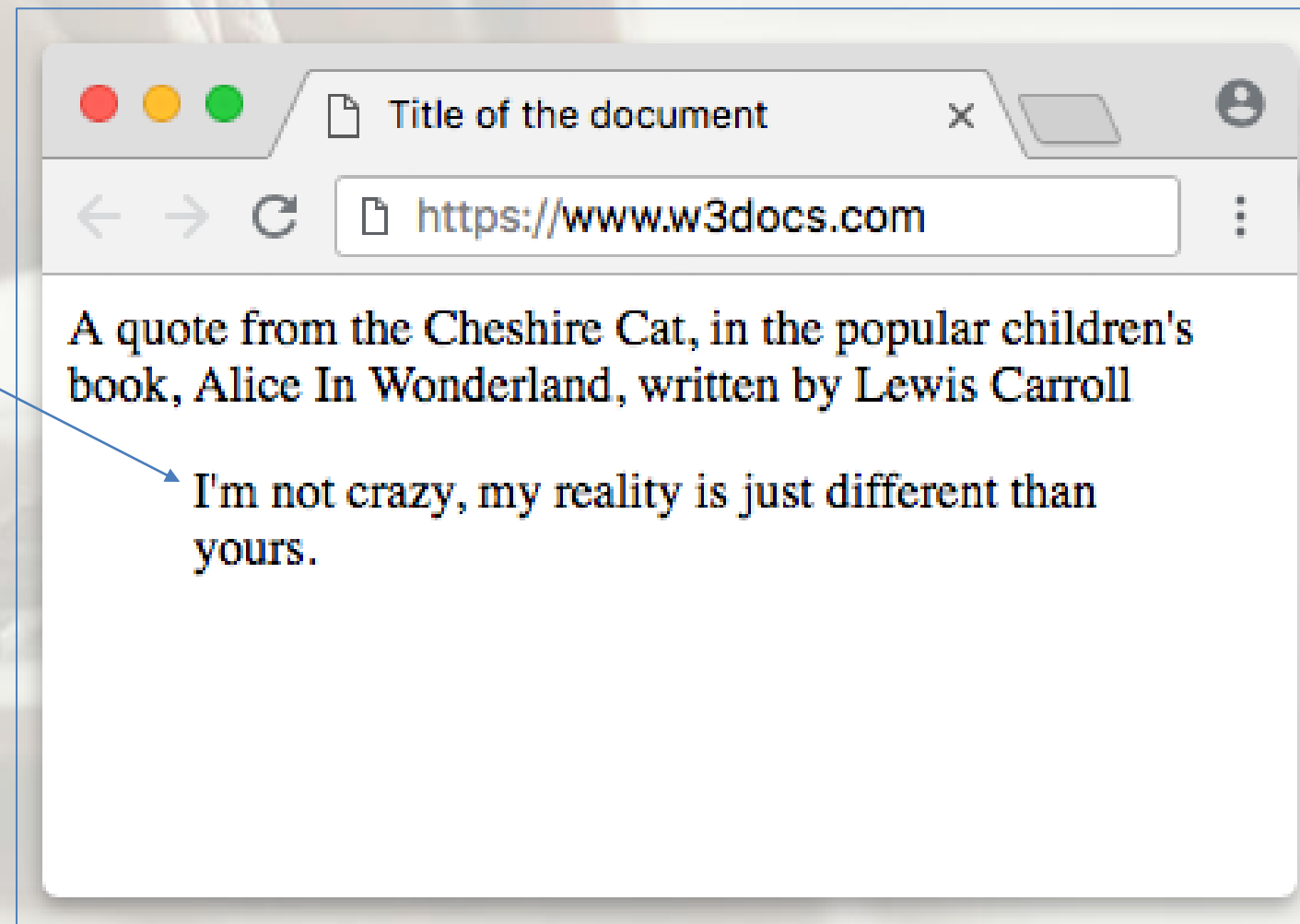
Property	Outcome	Definition	Example
<code>&lt;del&gt;</code>	Deleted text	The HTML <code>&lt;del&gt;</code> element defines text that has been deleted from a document. Browsers will usually strike a line through deleted text.	Deleted <del>text</del>
<code>&lt;ins&gt;</code>	Inserted text	The HTML <code>&lt;ins&gt;</code> element defines a text that has been inserted into a document. Browsers will usually underline inserted text:	Inserted <u>text</u> .
<code>&lt;sub&gt;</code>	Subscript text	The HTML <code>&lt;sub&gt;</code> element expresses subscript text. Subscript text appears half a character below the normal line and is sometimes rendered in a smaller font. Subscript text can be used for Chemistry, like H <sub>2</sub> O.	Su <sub>b</sub> script text
<code>&lt;sup&gt;</code>	Superscript text	The HTML <code>&lt;sup&gt;</code> element specifies superscript text. Superscript text appears half a character above the normal line and is sometimes rendered in a smaller font. Superscript text can be used for footnotes, as WWW [1]:	Su <sup>P</sup> erscript text



# HTML Text Formatting

## HTML `<blockquote>` for Quotations

The HTML `<blockquote>` element identifies a section that is quoted from another source. Browsers typically indent `<blockquote>` elements, as can be seen below:



# HTML Text Formatting

## HTML `<q>` for Short Quotations

The HTML `<q>` tag specifies a short quotation. Browsers generally insert quotation marks around the quotation, as follows:

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

<p>Browsers usually insert quotation
marks around the q element.</p>

<p>WWF's goal is to: <q>Build a future
where people live in harmony with
nature.</q></p>

</body>
</html>
```

Browsers usually insert quotation marks around the q element.

WWF's goal is to: "Build a future where people live in harmony with nature."

# HTML Text Formatting

## HTML <abbr> for Abbreviations

- The HTML <abbr> tag specifies an abbreviation or an acronym, like "HTML", "CSS", "Mr.", "Dr.", "ASAP", etc. Marking abbreviations can give valuable information to browsers, translation systems and search engines, as seen previously.
- In case one does not know the meaning of any given abbreviation, he/she could use the global title attribute to show the description for the abbreviation/acronym when mousing over the element, as seen below:

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

<p>The <abbr title="World Health
Organization">WHO</abbr> was founded in
1948.</p>

<p>Marking up abbreviations can give
useful information to browsers,
translation systems and search-engines.
</p>

</body>
</html>
```

The WHO was founded in 1948.

Marking up World Health Organization useful information to browsers, translation systems and search-engines.



# HTML Text Formatting

## HTML `<address>` for Contact Information

- The HTML `<address>` tag defines the contact information for the author/owner of a document or an article. It can be an email address, URL, physical address, phone number, etc. The text comprised in the `<address>` element is typically presented in italic, and browsers will always add a line break before and after it, as follows:

<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;  &lt;p&gt;Please contact us:.&lt;/p&gt;  &lt;address&gt; https://code4sp.eu/&lt;br&gt; https://www.facebook.com/Code4SP&lt;br&gt; &lt;/address&gt;  &lt;/body&gt; &lt;/html&gt;</pre>	<p>Please contact us:.</p> <p><i>https://code4sp.eu/</i> <i>https://www.facebook.com/Code4SP</i></p>
--	--




# HTML Text Formatting

## HTML `<cite>` for Work Title

- The HTML `<cite>` tag defines the title of a book, a poem, a song, a movie, a painting, and all creative works. It should be stated that the author's name is not the title of a work.
- As in the aforementioned tags, the text in the `<cite>` element normally renders in italic:

```
<!DOCTYPE html>  
<html>  
<body>  
  
  
<p><cite>The Scream</cite> by Edvard  
Munch, 1893.</p>  
  
</body>  
</html>
```



*The Scream* by Edvard Munch, 1893.

# HTML Text Formatting

## HTML `<bdo>` for Bi-Directional Override

- HTML `<bdo>` tag stands for "bidirectional override" which is used to override the current/default text direction. This tag sets the direction of content within it to execute on browser from left to right or right to left (rtl – right to left; ltr – left to right).

<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;  &lt;p&gt;If your browser supports bi- directional override (bdo), the next line will be written from right to left (rtl):&lt;/p&gt;  &lt;bdo dir="rtl"&gt;Code4SP&lt;/bdo&gt;  &lt;/body&gt; &lt;/html&gt;</pre>	<p>If your browser supports bi-directional override (bdo), the next line will be written from right to left (rtl):</p> <p>PS4edoC</p>
--	---

# HTML Comments

## Add Comments

- This element is used to add a comment to an HTML document. An HTML comment begins with `<!--` and closes with `-->`. HTML comments are visible to anyone that views the page source code but are not rendered when the HTML document is rendered by a browser.
- It should be noted that there is an exclamation point in the start tag, but not in the end tag. This feature is especially useful for placing notifications and reminders in the HTML code:

<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;  &lt;!-- This is a comment --&gt; &lt;p&gt;Code4SP project.&lt;/p&gt; &lt;!-- Comments are not displayed in the browser --&gt;  &lt;/body&gt; &lt;/html&gt;</pre>	<p>Code4SP project.</p>
--	-------------------------

# HTML Comments

## Hide Content

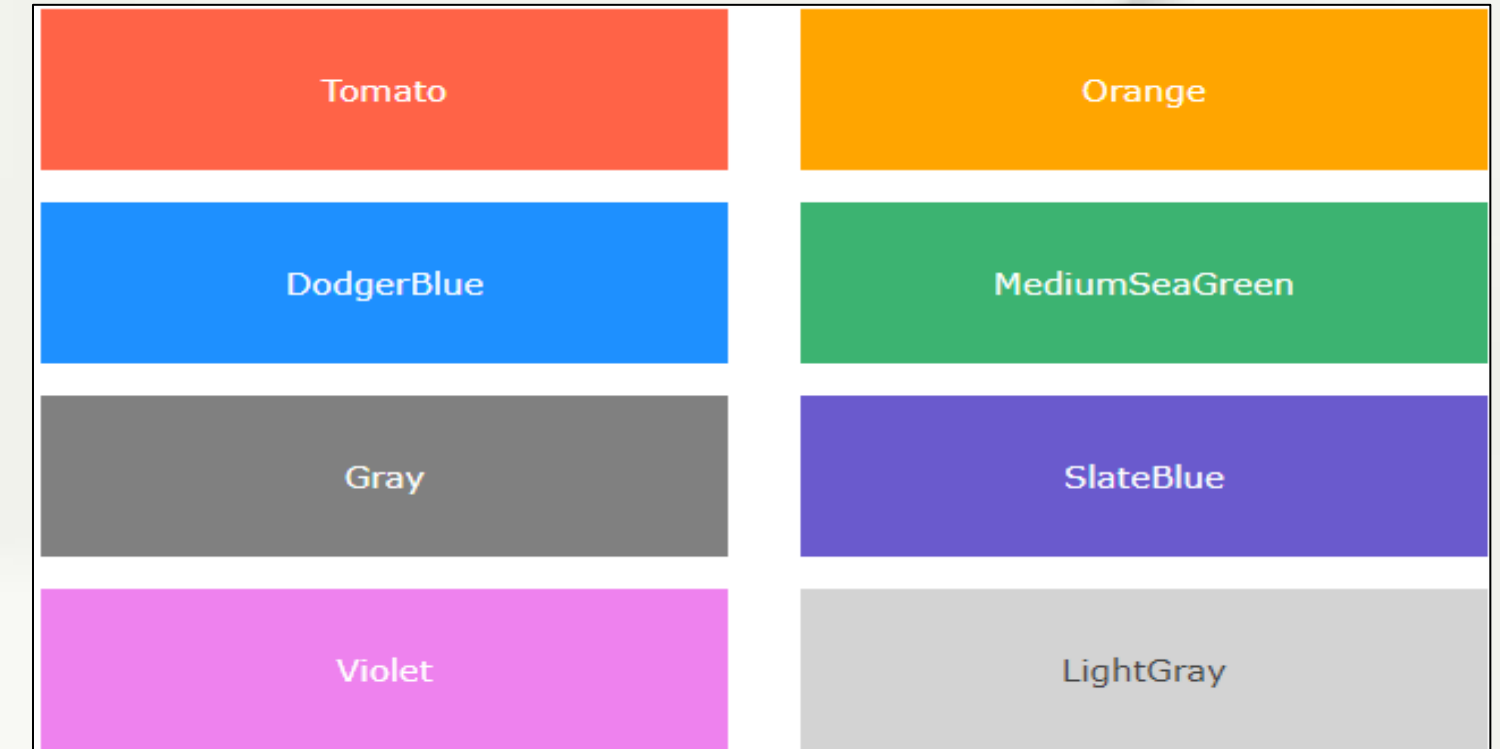
- Comments can also be used to hide content, and that can be helpful if one hides it for the moment. You can also hide more than one line, everything between the `<!--` and the `-->` will be hidden from the display.
- Comments are also great for debugging HTML, because one can comment out HTML lines of code, one at a time, to search for errors.

<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;  &lt;p&gt;Code4SP project.&lt;/p&gt;  &lt;!-- &lt;p&gt;This content is hidden. &lt;/p&gt; -- &gt;  &lt;p&gt;But this will appear.&lt;/p&gt;  &lt;/body&gt; &lt;/html&gt;</pre>	<p>Code4SP project.</p> <p>But this will appear.</p>
---	--



# HTML Colours

- HTML colours are stipulated with predefined colour names, or with RGB, HEX, HSL, RGBA, or HSLA values.



```

<!DOCTYPE html>
<html>
<body>

<h1 style="background-color: DodgerBlue;">Code4SP</h1>

<p style="background-color: Tomato;">
Code4SP's main objectives and
priorities are in full interweaving
with the European Commission's goals,
contributing towards providing tailored
education and training to digitally
excluded groups, including migrants and
young people from disadvantaged
backgrounds, while in parallel, taking
into consideration the labor market
needs.
</p>

</body>
</html>

```

## Code4SP

Code4SP's main objectives and priorities are in full interweaving with the European Commission's goals, contributing towards providing tailored education and training to digitally excluded groups, including migrants and young people from disadvantaged backgrounds, while in parallel, taking into consideration the labor market needs.

- Colours can be set for the **page background**:



# HTML Colours

- The same principle can be applied to the **text colour** →

<pre> &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;  &lt;h3 style="color:Tomato;"&gt;Code4SP&lt;/h3&gt;  &lt;p style="color:DodgerBlue;"&gt;The target will be reached through the upscaling of an already existing good practice at a local level in Germany, which had as a result, top paid programming jobs for asylum seekers.&lt;/p&gt;  &lt;p style="color:MediumSeaGreen;"&gt;Enhance employers' motivation and predisposition for potential employment of individuals that belong to disadvantaged populations, thus breaking any negative stereotypes on this issue.&lt;/p&gt;  &lt;/body&gt; &lt;/html&gt; </pre>	<p><b>Code4SP</b></p> <p>The target will be reached through the upscaling of an already existing good practice at a local level in Germany, which had as a result, top paid programming jobs for asylum seekers.</p> <p>Enhance employers' motivation and predisposition for potential employment of individuals that belong to disadvantaged populations, thus breaking any negative stereotypes on this issue.</p>
--	--

<pre> &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;  &lt;h1 style="border: 2px solid Tomato;"&gt;Code4SP&lt;/h1&gt;  &lt;h1 style="border: 2px solid DodgerBlue;"&gt;Code4SP&lt;/h1&gt;  &lt;h1 style="border: 2px solid Violet;"&gt;Code4SP&lt;/h1&gt;  &lt;/body&gt; &lt;/html&gt; </pre>	<div style="border: 2px solid red; padding: 5px; text-align: center;">Code4SP</div> <div style="border: 2px solid blue; padding: 5px; text-align: center;">Code4SP</div> <div style="border: 2px solid purple; padding: 5px; text-align: center;">Code4SP</div>
---	---

← Also, for the colour of **borders**:

# HTML Colours

- **Colour Values**

As stated above, HTML colours are stipulated with predefined colour names, or with RGB, HEX, HSL, RGBA, or HSLA values. The following three <div> elements have their background color set with RGB, HEX, and HSL values:

```
rgb(255, 99, 71)
```

```
#ff6347
```

```
hsl(9, 100%, 64%)
```

# HTML Colours

- **Colour Values**

Transparency is also a feature that could be added when defining a colour, by adding an Alpha channel to it. The following example as 50% transparency:

```
rgba(255, 99, 71, 0.5)
```

```
hsla(9, 100%, 64%, 0.5)
```



# HTML Colours

## Colour Values

The code for setting up both features will ease learners' understanding. As seen, it comprises CSS features to be explored later:

<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;  &lt;p&gt;Same as color name "Tomato":&lt;/p&gt;  &lt;h1 style="background-color:rgb(255, 99, 71);"&gt;rgb(255, 99, 71)&lt;/h1&gt; &lt;h1 style="background- color:#ff6347;"&gt;#ff6347&lt;/h1&gt; &lt;h1 style="background-color:hsl(9, 100%, 64%);"&gt;hsl(9, 100%, 64%)&lt;/h1&gt;  &lt;p&gt;Same as color name "Tomato", but 50% transparent:&lt;/p&gt; &lt;h1 style="background-color:rgba(255, 99, 71, 0.5);"&gt;rgba(255, 99, 71, 0.5) &lt;/h1&gt; &lt;h1 style="background-color:hsla(9, 100%, 64%, 0.5);"&gt;hsla(9, 100%, 64%, 0.5)&lt;/h1&gt;  &lt;p&gt;In addition to the predefined color names, colors can be specified using RGB, HEX, HSL, or even transparent colors using RGBA or HSLA color values. &lt;/p&gt;  &lt;/body&gt; &lt;/html&gt;</pre>	<p>Same as color name "Tomato":</p> <p><b>rgb(255, 99, 71)</b></p> <p><b>#ff6347</b></p> <p><b>hsl(9, 100%, 64%)</b></p> <p>Same as color name "Tomato", but 50% transparent:</p> <p><b>rgba(255, 99, 71, 0.5)</b></p> <p><b>hsla(9, 100%, 64%, 0.5)</b></p> <p>In addition to the predefined color names, colors can be specified using RGB, HEX, HSL, or even transparent colors using RGBA or HSLA color values.</p>
---	---

# HTML Links

- Links can be found in nearly all webpages.
- They allow internet users to navigate from page to page. It does not have to be text, as it can be an image or any other HTML element.
- HTML links are hyperlinks, which can be clicked, jumping to another document.
- When the mouse is moved over a link, the mouse arrow will turn into a little hand.
- The HTML `<a>` tag defines a hyperlink. It has the following syntax:

```
<a href="url">link text</a>
```

# HTML Links

- The most significant attribute of the `<a>` element is the `href` attribute, which indicates the link's destination.
- The link text is the part that will be visible to the user. By clicking on the link text, the reader will be redirected to the required URL address.

- By default, links will be seen in all browsers as follows:

An unvisited link is [underlined and blue](#)

A visited link is [underlined and purple](#)

An active link is [underlined and red](#)

- **Note:** Link colours can be changed by using CSS features.



# HTML Links

## The target attribute

- By default, the linked page will be shown in the current browser window. To modify this, learners must indicate another target for the link.,
- The **target** attribute indicates where to open the linked document. It can have one of the following values:
  - \_self** - Default. Opens the document in the same window/tab as it was clicked
  - \_blank** - Opens the document in a new window or tab
  - \_parent** - Opens the document in the parent frame
  - \_top** - Opens the document in the full body of the window



# HTML Links

## Using an image as a Link

- To use an image as a link, just put the `<img>` tag inside the `<a>` tag, as it can be checked over the following tutorial:

[https://www.w3schools.com/html/tryit.asp?filename=tryhtml\\_links\\_image](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_links_image)

# HTML Links

## Linking an email address

- To create a link that opens the user's email software (letting them send a new email), **mailto:** should be added inside the **href** attribute, following the next example:

```
<a href="mailto:someone@example.com">Send email</a>
```

## Create a Bookmark in HTML

- HTML links can be applied to make bookmarks, so that readers can jump to particular parts of a web page, and it can be truly useful if the web page is very long. This process is composed of two very simple steps:
- To create a bookmark - first the bookmark should be created, and then a link should be added to it. To create, the **id** attribute should be used (e.g.: `<h2 id="C1">Chapter 1</h2>`), then, a link to the bookmark, from within the same page, should be added (e.g.: `<a href="#C4">Jump to Chapter 4</a>`)
- When the link is clicked, the page will scroll to the location with the bookmark.



# HTML Images

## Inserting images into webpages

- Images improve visual look of the web pages by making them more appealing and colourful. The `<img>` tag is used to add images in the HTML pages. It is an empty element and contains attributes only.
- Each image must have at least two attributes: the `src` and `alt` attributes. The `src` attribute informs the browser where to find the image, being its value the URL of the image file. The `alt` attribute provides an alternative text for the image if it is inaccessible or cannot be displayed for some reason (slow connection, image is not available at the specified URL, or if the user uses a screen reader or non-graphical browser). Its value should be a meaningful substitute for the image, preferably a suggestive text.



# HTML Images

## Inserting images into webpages

- Images improve visual look of the web pages by making them more appealing and colourful. The `<img>` tag is used to add images in the HTML pages. It is an empty element and contains attributes only.
- Each image must have at least two attributes: the `src` and `alt` attributes. The `src` attribute informs the browser where to find the image, being its value the URL of the image file. The `alt` attribute provides an alternative text for the image if it is inaccessible or cannot be displayed for some reason (slow connection, image is not available at the specified URL, or if the user uses a screen reader or non-graphical browser). Its value should be a meaningful substitute for the image, preferably a suggestive text.

# HTML Images

## Inserting images into webpages

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Placing Images in HTML Documents</title>
</head>
<body>
  
</body>
</html>
```





# HTML Images

## Setting the Width and Height of an Image

- The **width** and **height** attributes are used to indicate the width and height of an image.
- The values of these attributes are interpreted in pixels by default. It's a good practice to specify both the width and height attributes, so that browser can assign that much of space for the image before it is transferred.

# HTML Images

## Setting the Width and Height of an Image

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Setting Image Dimensions in HTML</title>
</head>
<body>
  
  
  
</body>
</html>
```







# HTML Images

## Setting the Width and Height of an Image

The **style** attribute can also be used to indicate width and height. It prevents style sheets from changing the image size by accident, because inline style has the highest priority.



# HTML Images

## Using the HTML5 Picture Element

- Now and then, scaling an image up or down to fit different devices (or screen sizes) does not work as expected. In addition, reducing the image dimension using the **width** and **height** attribute does not decrease the initial file size. In order to solve these problems, HTML5 has introduced the **<picture>** tag that allows to define multiple versions of an image to target different types of devices.
- The **<picture>** element contains zero or more **<source>** elements, each referring to different image source, and one **<img>** element at the end. Likewise, each **<source>** element has the media attribute which specifies a media condition that is used by the browser to determine when a particular source should be used.

# HTML Images

## Image Maps

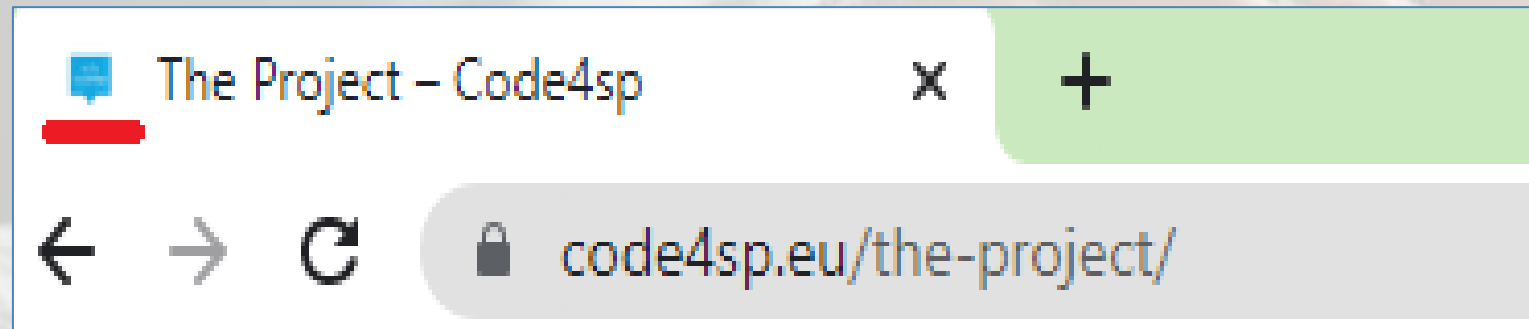
- An image map allows one to define hotspots on an image that acts out just like a hyperlink. The key idea behind creating an image map is to give an simple way of linking various parts of an image without dividing it into separate image files. For example, a map of a country may have each city hyperlinked to more information about that city.
- The following example is pretty accurate about these features:  
<https://www.tutorialrepublic.com/codelab.php?topic=html&file=image-maps>
- The **name** attribute of the **<map>** tag is used to reference the map from the **<img>** tag using its **usemap** attribute. The **<area>** tag is used inside the **<map>** element to define the clickable areas on an image. Any number of clickable areas can be defined within an image.



# HTML Images

## HTML Favicon

A favicon is a small image displayed to the left of the page title in the browser tab:



- To add a favicon to a website, a favicon image should be saved to the root directory of webserver. Other way is by creating a folder in the root directory called images, then saving the favicon image in this folder.
- A common name for a favicon image is "favicon.ico".



# HTML Images

## HTML Favicon

- Next, a `<link>` element should be added to the "index.html" file, after the `<title>` element, as follows:

```
<!DOCTYPE html>
<html>
<head>
  <title>Code4SP</title>
  <link rel="icon" type="image/x-icon" href="/images/favicon.ico">
</head>
<body>
<h1>Our project</h1>
<p>Co-funded by the E+ fund of the EC.</p>
</body>
</html>
```

# HTML Tables

## Creating Tables in HTML

- HTML tables allow to arrange data into rows and columns. They are generally used to display tabular data like product listings, customer's details, financial reports, etc.
- A table can be created using the `<table>` element. Inside the `<table>` element, the `<tr>` elements can be utilized to create rows, and to create columns inside a row the `<td>` elements can be used. A cell can be defined as a header for a group of table cells using the `<th>` element.
- Tables do not have any borders by default. The CSS `border` property can be used to add borders to the tables. Furthermore, table cells are sized just large enough to fit the contents by default. To add more space around the content in the table cells, the CSS `padding` property can be used.

# HTML Tables

## Creating Tables in HTML

- By default, borders around the table and their cells are separated from each other. But they can be collapsed into one by using the **border-collapse** property on the **<table>** element. Additionally, text inside the **<th>** elements is displayed in bold font, aligned horizontally center in the cell by default.
- To change the default alignment, the CSS text-align property can be used. To do so, it is suggested that learners reach the CSS topic first. Most of the **<table>** element's attribute such as **border**, **cellpadding**, **cellspacing**, **width**, **align**, etc. for styling table appearances in earlier versions has been dropped in HTML5, so they should be avoided. **CSS should be privileged to style HTML tables.**



# HTML Tables

## Creating Tables in HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Creating Tables in HTML</title>
</head>
<body>
  <h2>Spotify Top Songs of 2021 (USA)</h2>
  <table>
    <tr>
      <th>No.</th>
      <th>Song - Band </th>
      <th>Lenght</th>
    </tr>
    <tr>
      <td>1</td>
      <td>drivers licence - Olivia Rodrigo</td>
      <td>4:02</td>
    </tr>
    <tr>
      <td>2</td>
      <td>MONTERO (Call me by your name) - Lil Nas X</td>
      <td>2:17</td>
    </tr>
    <tr>
      <td>3</td>
      <td>STAY - The Kid LAROI ft. Justin Bieber</td>
      <td>2:21</td>
    </tr>
  </table>
</body>
</html>
```

### Spotify Top Songs of 2021 (USA)

No.	Song - Band	Lenght
1	drivers licence - Olivia Rodrigo	4:02
2	MONTERO (Call me by your name) - Lil Nas X	2:17
3	STAY - The Kid LAROI ft. Justin Bieber	2:21





# HTML Tables

## Spanning Multiple Rows and Columns

- Spanning allows to extend table rows and columns across multiple other rows and columns. Usually, a table cell cannot pass over into the space below or above another table cell.
- However, the **rowspan** or **colspan** attributes can be used to span multiple rows or columns in a table.

# HTML Tables

## Spanning Multiple Rows and Columns

- Likewise, the **rowspan** attribute can be used to create a cell that spans more than one row, as follows:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Span Multiple Rows in an HTML Table</title>
  <style>
    table {
      width: 300px;
      border-collapse: collapse;
    }
    table, th, td {
      border: 1px solid black;
    }
    th, td {
      padding: 10px;
    }
  </style>
</head>
<body>
  <h2>Spanning Rows</h2>
  <table>
    <tr>
      <th>Name:</th>
      <td>John Carter</td>
    </tr>
    <tr>
      <th rowspan="2">Phone:</th>
      <td>55577854</td>
    </tr>
    <tr>
      <td>55577855</td>
    </tr>
  </table>
</body>
</html>
```

**Spanning Rows**

<b>Name:</b>	John Carter
<b>Phone:</b>	55577854
	55577855

# HTML Tables

## Table Captions

- A caption (or title) for tables can be created by using the `<caption>` element. This element should be placed directly after the (opening) `<table>` tag. By default, caption appears at the top of the table, but this can be changed by using the CSS `caption-side` property.

```
<!DOCTYPE html>
<html>
  <body>
    <table border=1>
      <caption> WIKITECHY WEBSITE </caption>
      <tr>
        <th>Firstname</th>
        <th>Lastname</th>
      </tr>
      <tr>
        <td>Wiki</td>
        <td>techy</td>
      </tr>
    </table>
  </body>
</html>
```

WIKITECHY  
WEBSITE

Firstname	Lastname
Wiki	techy

# HTML Tables

## Defining a Table Header, Body and Footer

- HTML provides a series of tags `<thead>`, `<tbody>`, and `<tfoot>` that aid learners to create more coordinated tables, by defining header, body and footer regions, in that order.



# HTML Tables

## Defining a Table Header, Body and Footer

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>HTML Table with a Header, Footer and Body</title>
  <style>
    table {
      width: 300px;
      border-collapse: collapse;
    }
    table, th, td {
      border: 1px solid black;
    }
    th, td {
      padding: 10px;
      text-align: left;
    }
  </style>
</head>
<body>
  <table>
    <thead>
      <tr>
        <th>Items</th>
        <th>Expenditure</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td>Stationary</td>
        <td>2,000</td>
      </tr>
      <tr>
        <td>Furniture</td>
        <td>10,000</td>
      </tr>
    </tbody>
    <tfoot>
      <tr>
        <th>Total</th>
        <td>12,000</td>
      </tr>
    </tfoot>
  </table>
</body>
</html>
```

Items	Expenditure
Stationary	2,000
Furniture	10,000
<b>Total</b>	12,000

# HTML Lists

HTML lists are applied to present information in a well-formed and semantic way. Inside of them, one can add text, images, links, line breaks, etc. There are three different types of lists in HTML and each one has a specific purpose and meaning:

- **A) Unordered lists** — Used to create a list of related items, in no particular order.
- **B) Ordered lists** — Used to create a list of related items, in a certain order.
- **C) Description lists** — Used to create a list of terms and their descriptions.

# HTML Lists

## A) Unordered lists

- An unordered list created using the `<ul>` element, and each list item begins with the `<li>` element. It is marked with bullets, as follows:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Reasons why you should travel by train</title>
</head>
<body>
  <h2>Reasons why you should travel by train</h2>
  <ul>
    <li>It is less expensive</li>
    <li>It is eco-friendly</li>
    <li>You can enjoy the view</li>
  </ul>
</body>
</html>
```

### Reasons why you should travel by train

- It is less expensive
- It is eco-friendly
- You can enjoy the view



# HTML Lists

## B) Ordered lists

- An ordered list is created by using the `<ol>` element, and each list item starts with the `<li>` element. Ordered lists are used when the order of the list's items is critical. It is marked with numbers, as follows:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>HTML Ordered List</title>
</head>
<body>
  <h2>How to cook your own veggie burger</h2>
  <ol>
    <li>Dump your ground meat into a bowl. (We go for ground meat with around 20% fat.) Season it with salt, pepper, and whatever else you want; you can add spices, perhaps, or Worcestershire sauce, or shallots, or chiles.</li>
    <li>Shape your burgers into patties, using your thumb to make an indentation in the center; this will keep the burgers from puffing up. Keep in mind that the burgers will shrink up a bit once you cook them, so make your patties a bit bigger than you want them later.</li>
    <li>Oil your grill or a cast-iron pan, and grill or sear those patties. (How many times to flip them is up for debate -- but when I'm grilling, I flip once so I can get get those nice grill marks.) Cook them until your desired doneness (around 125-130°F for medium rare, around 1 minute per side for each inch of thickness). But before you take them off the grill...</li>
    <li>...add your cheese and toast your buns. Let the cheese melt while the burgers are still on the grill; to speed things up, you can close the cover.</li>
    <li>Once your burgers iare finished cooking, and your cheese is melty and your buns are nicely charred, throw some condiments and toppings on those burgers. Anything goes. (Really, anything goes.) Bite into it and let those juices run down your chin, and rejoice that it's summer. And then make another round, because now you know how.</li>
  </ol>
</body>
</html>
```

### How to cook your own veggie burger

1. Dump your ground meat into a bowl. (We go for ground meat with around 20% fat.) Season it with salt, pepper, and whatever else you want; you can add spices, perhaps, or Worcestershire sauce, or shallots, or chiles.
2. Shape your burgers into patties, using your thumb to make an indentation in the center; this will keep the burgers from puffing up. Keep in mind that the burgers will shrink up a bit once you cook them, so make your patties a bit bigger than you want them later.
3. Oil your grill or a cast-iron pan, and grill or sear those patties. (How many times to flip them is up for debate -- but when I'm grilling, I flip once so I can get get those nice grill marks.) Cook them until your desired doneness (around 125-130°F for medium rare, around 1 minute per side for each inch of thickness). But before you take them off the grill...
4. ...add your cheese and toast your buns. Let the cheese melt while the burgers are still on the grill; to speed things up, you can close the cover.
5. Once your burgers iare finished cooking, and your cheese is melty and your buns are nicely charred, throw some condiments and toppings on those burgers. Anything goes. (Really, anything goes.) Bite into it and let those juices run down your chin, and rejoice that it's summer. And then make another round, because now you know how.



# HTML Lists

## C) Description Lists

- A description list is a list of items with a description or definition of each item.
- The description list is created using `<dl>` element. The `<dl>` element is used in conjunction with the `<dt>` element which specify a term, and the `<dd>` element which specify the term's definition.
- Browsers usually render the definition lists by placing the terms and definitions in separate lines, where the term's definitions are slightly indented.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>HTML Description or Definition List</title>
</head>
<body>
  <h2>What are bread and coffee?</h2>
  <dl>
    <dt>Bread</dt>
    <dd>A baked food made of flour.</dd>
    <dt>Coffee</dt>
    <dd>A drink made from roasted coffee beans.</dd>
  </dl>
</body>
</html>
```

### What are bread and coffee?

Bread  
A baked food made of flour.

Coffee  
A drink made from roasted coffee beans.

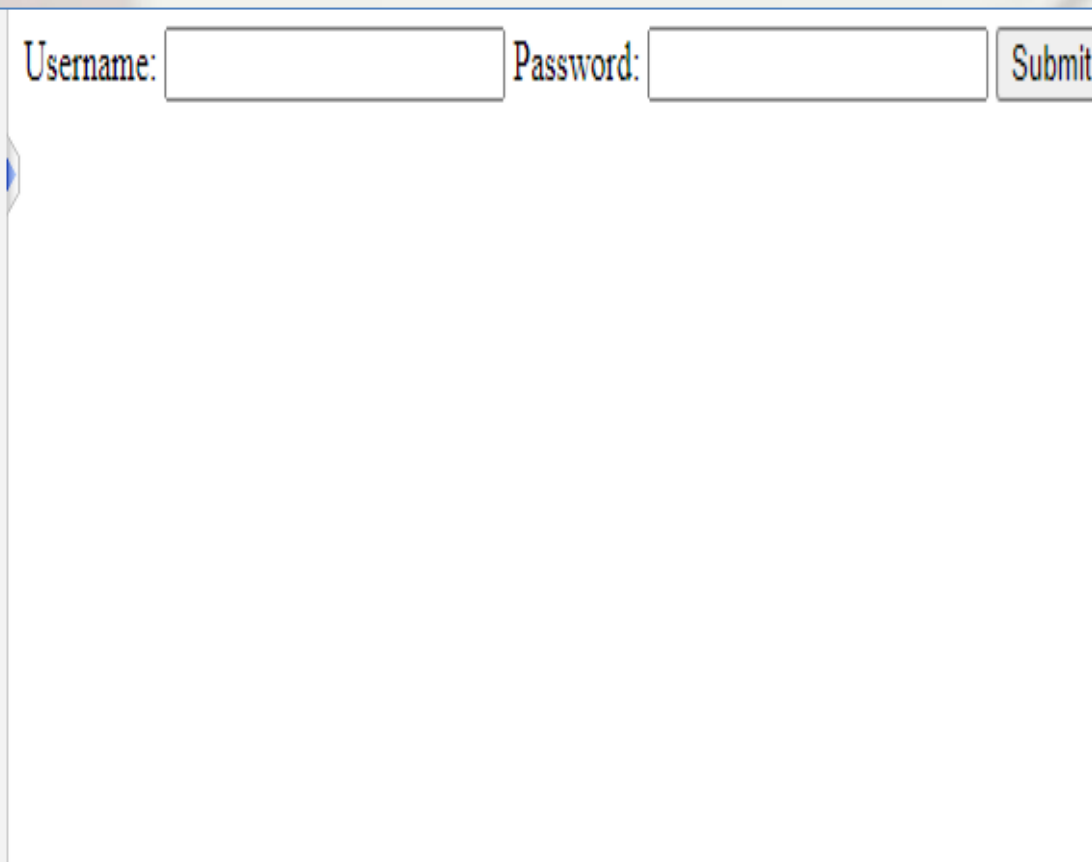
# HTML Forms

- HTML Forms are expected to collect different types of user inputs, such as contact details (name, email, phone number, bank account, etc.).
- Forms include special elements known as controls like input box, checkboxes, radio-buttons, submit buttons, etc.
- To this end, users fill in a form with this data, either via text or box selection, submitting it later to a webserver for processing this data.

# HTML Forms

- The `<form>` tag is used to generate an HTML form. A simple example of a login form would be as follows:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Simple HTML Form</title>
</head>
<body>
  <form action="/examples/actions/confirmation.php" method="post">
    <label>Username: <input type="text" name="username"></label>
    <label>Password: <input type="password" name="userpass"></label>
    <input type="submit" value="Submit">
  </form>
</body>
</html>
```





# HTML Forms

There are different types of controls to be applied in a HTML form, being **input elements** the most frequently used ones. They identify various types of user input fields, depending on the type attribute. Input elements can be **text fields, password fields, checkboxes, submit buttons, reset buttons, file select boxes**, as well as several new input types introduced in HTML5 (they can be checked out [here](#)).



# HTML Forms

- **Text fields** are areas that let users add text. These are created by using an `<input>` element, whose `type` attribute has a value of `text`. It should be noted that the `<label>` tag is used to identify the labels for `<input>` elements. If the webmaster wants his/her users to enter several lines, `<textarea>` should be added instead.

# HTML Forms

- **Text fields** are areas that let users add text. These are created by using an `<input>` element, whose `type` attribute has a value of `text`. It should be noted that the `<label>` tag is used to identify the labels for `<input>` elements. If the webmaster wants his/her users to enter several lines, `<textarea>` should be added instead.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>HTML Text Input Field</title>
</head>
<body>
  <form>
    <label for="user-name">Login:</label>
    <input type="text" name="username" id="user-name">
  </form>
</body>
</html>
```

Login:

# HTML Forms

- **Password fields** are like text fields, being the only difference characters in a password field are hidden, so they are displayed as asterisks or dots. Likewise, this procedure prevents someone else from reading the password on the screen. This is as well a single-line text input control generated using an `<input>` element whose `type` attribute has a value of `password`.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>HTML Password Input Field</title>
</head>
<body>
  <form>
    <label for="user-pwd">Password:</label>
    <input type="password" name="user-password" id="user-pwd">
  </form>
</body>
</html>
```

Password:

# HTML Forms

- **Radio buttons** are applied to allow the user select exactly one option from a pre-specified set of options. It is generated using an `<input>` element whose `type` attribute has a value of `radio`.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>HTML Radio Buttons</title>
</head>
<body>
  <form>
    <input type="radio" name="Civil Status" value="single" id="single">
    <label for="single">Single</label>
    <input type="radio" name="Civil Status" value="married" id="married">
    <label for="married">Married</label>
    <input type="radio" name="Civil Status" value="other" id="other">
    <label for="other">Other</label>
  </form>
</body>
</html>
```

Single  Married  Other



# HTML Forms

- **Checkboxes** make available to the user one or more choices from a pre-defined set of options. It is generated using an `<input>` element whose `type` attribute has a value of `checkbox`. If one prefers to generate a checkbox (or radio button) selected by default, one simply must add the attribute `checked` to the input element (`<input type="checkbox" checked>`).

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>HTML Checkboxes</title>
</head>
<body>
  <form>
    <input type="checkbox" name="sports" value="football" id="football">
    <label for="football">Football</label>
    <input type="checkbox" name="sports" value="handball" id="handball">
    <label for="handball">Handball</label>
    <input type="checkbox" name="sports" value="basketball" id="basketball">
    <label for="basketball">Basketball</label>
  </form>
</body>
</html>
```

Football  Handball  Basketball

# HTML Forms

- **Checkboxes** make available to the user one or more choices from a pre-defined set of options. It is generated using an `<input>` element whose `type` attribute has a value of `checkbox`. If one prefers to generate a checkbox (or radio button) selected by default, one simply must add the attribute `checked` to the input element (`<input type="checkbox" checked>`).

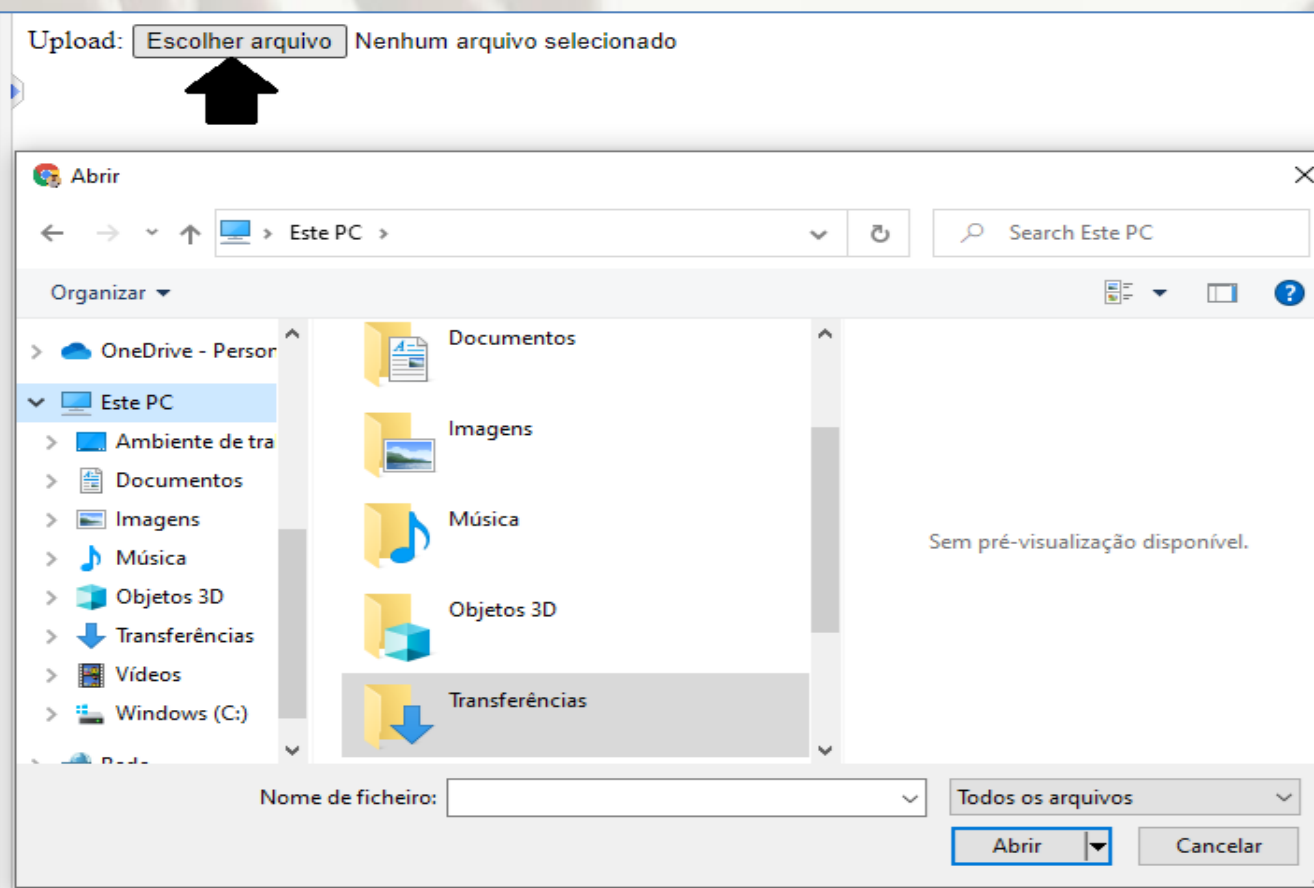
```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>HTML Checkboxes</title>
</head>
<body>
  <form>
    <input type="checkbox" name="sports" value="football" id="football">
    <label for="football">Football</label>
    <input type="checkbox" name="sports" value="handball" id="handball">
    <label for="handball">Handball</label>
    <input type="checkbox" name="sports" value="basketball" id="basketball">
    <label for="basketball">Basketball</label>
  </form>
</body>
</html>
```

Football  Handball  Basketball

# HTML Forms

- **File select boxes** allow a user to browse for a local file and send it as an attachment with the form data. E.g., Google Chrome provides a file select input field with a 'Browse' button that permits the user to select a file from his/her hard drive.
- File select boxes are also created using an `<input>` element, whose **type** attribute value is set to **file**.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>HTML File Select box</title>
</head>
<body>
  <form>
    <label for="file-select">Upload:</label>
    <input type="file" name="upload" id="file-select">
  </form>
</body>
</html>
```



The screenshot shows a web browser window with a file upload form. The form has a label "Upload:" and a button "Escolher arquivo". Below the form, a file explorer window is open, showing the "Este PC" view with folders like "Documentos", "Imagens", "Música", and "Transferências". The file explorer window has a search bar and a "Abrir" button. An arrow points from the "Escolher arquivo" button to the file explorer window.

# HTML Forms

- **Textarea** can be defined as a multiple-line text input control that allows to enter more than one line of text. These controls are created using an `<textarea>` element, as follows:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>HTML Textarea</title>
</head>
<body>
  <form>
    <label for="address">Your opinion:</label>
    <textarea rows="3" cols="30" name="Your opinion" id="Your opinion">
  </textarea>
  </form>
</body>
</html>
```

Your opinion:



# HTML Forms

- **Select boxes** are dropdown lists of options in which a user can select one or more options from a pull-down menu. They are created by using the `<select>` element and `<option>` element. The `<option>` elements within the `<select>` element define each list item.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>HTML Select Box</title>
</head>
<body>
  <form>
    <label for="country">Country:</label>
    <select name="country" id="country">
      <option value="Portugal">Portugal</option>
      <option value="Cyprus">Cyprus</option>
      <option value="Greece">Greece</option>
    </select>
  </form>
</body>
</html>
```

Country: Portugal ▼

- Portugal
- Cyprus
- Greece

# HTML Forms

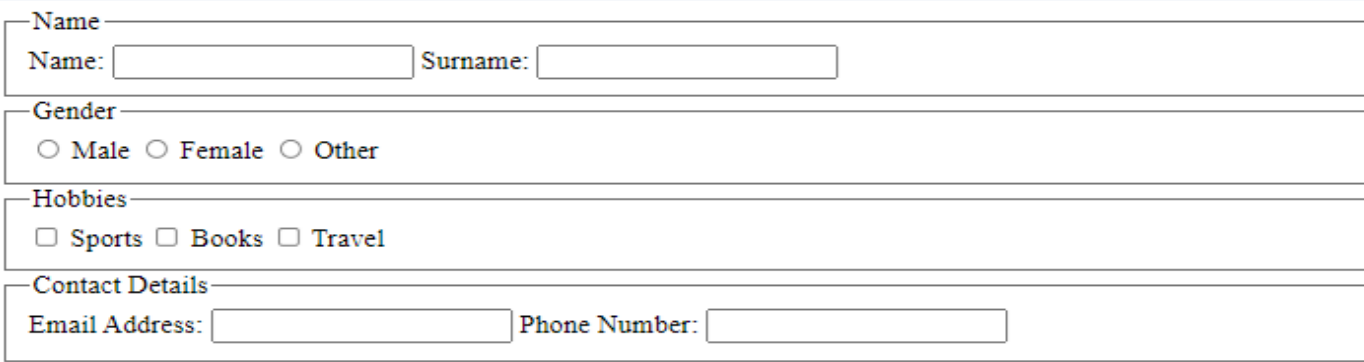
- **Submit and reset buttons** are very common in most of the websites. Submit buttons are used to send (form) data to a web server, while reset buttons are created to reset the form to default values. When the user clicks the submit button, the form data is sent to the file specified in the form's **action** attribute to process the submitted data.
- Buttons can also be created using the **<button>** element. They have the same purpose as buttons created with the input element. However, they offer more rendering possibilities, as they allow the embedding of other [HTML elements](#).

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>HTML Submit and Reset Buttons</title>
</head>
<body>
  <form action="/examples/html/action.php" method="post">
    <label for="first-name">First Name:</label>
    <input type="text" name="first-name" id="first-name">
    <input type="submit" value="Submit">
    <input type="reset" value="Reset">
  </form>
</body>
</html>
```

First Name:

# HTML Forms

- **Grouping form controls** are a great tool for users to locate a control, making the form more accessible. The `<legend>` element is key for creating logically related controls, as seen in the picture below:

<pre> &lt;!DOCTYPE html&gt; &lt;html lang="en"&gt; &lt;head&gt;   &lt;title&gt;Grouping Form Controls in HTML&lt;/title&gt; &lt;/head&gt; &lt;body&gt;   &lt;form&gt;     &lt;fieldset&gt;       &lt;legend&gt;Name&lt;/legend&gt;       &lt;label&gt;Name: &lt;input type="text" name="firstname"&gt;&lt;/label&gt;       &lt;label&gt;Surname: &lt;input type="text" name="lastname"&gt;&lt;/label&gt;     &lt;/fieldset&gt;     &lt;fieldset&gt;       &lt;legend&gt;Gender&lt;/legend&gt;       &lt;label&gt;&lt;input type="radio" name="gender" value="male"&gt; Male&lt;/label&gt;       &lt;label&gt;&lt;input type="radio" name="gender" value="female"&gt; Female&lt;/label&gt;       &lt;label&gt;&lt;input type="radio" name="gender" value="other"&gt; Other&lt;/label&gt;     &lt;/fieldset&gt;     &lt;fieldset&gt;       &lt;legend&gt;Hobbies&lt;/legend&gt;       &lt;label&gt;&lt;input type="checkbox" name="hobbies" value="sports"&gt; Sports&lt;/label&gt;       &lt;label&gt;&lt;input type="checkbox" name="hobbies" value="culture"&gt; Books&lt;/label&gt;       &lt;label&gt;&lt;input type="checkbox" name="hobbies" value="leisure"&gt; Travel&lt;/label&gt;     &lt;/fieldset&gt;     &lt;fieldset&gt;       &lt;legend&gt;Contact Details&lt;/legend&gt;       &lt;label&gt;Email Address: &lt;input type="email" name="email"&gt;&lt;/label&gt;       &lt;label&gt;Phone Number: &lt;input type="text" name="phone"&gt;&lt;/label&gt;     &lt;/fieldset&gt;   &lt;/form&gt; &lt;/body&gt; &lt;/html&gt; </pre>	
---	--

# HTML Forms

Below there is a list of frequently used form element's attributes:

Attribute	Description
<b>name</b>	Specifies the name of the form.
<b>action</b>	Specifies the URL of the program or script on the web server that will be used for processing the information submitted via form.
<b>method</b>	Specifies the HTTP method used for sending the data to the web server by the browser. The value can be either get (the default) and post.
<b>target</b>	Specifies where to display the response that is received after submitting the form. Possible values are <code>_blank</code> , <code>_self</code> , <code>_parent</code> and <code>_top</code> .
<b>enctype</b>	Specifies how the form data should be encoded when submitting the form to the server. Applicable only when the value of the method attribute is post.



# HTML Forms

Below there is a list of frequently used form element's attributes:

Attribute	Description
<b>name</b>	Specifies the name of the form.
<b>action</b>	Specifies the URL of the program or script on the web server that will be used for processing the information submitted via form.
<b>method</b>	Specifies the HTTP method used for sending the data to the web server by the browser. The value can be either get (the default) and post.
<b>target</b>	Specifies where to display the response that is received after submitting the form. Possible values are <code>_blank</code> , <code>_self</code> , <code>_parent</code> and <code>_top</code> .
<b>enctype</b>	Specifies how the form data should be encoded when submitting the form to the server. Applicable only when the value of the method attribute is post.

# HTML iFrame

- An iframe (*or inline frame*) is used to exhibit external objects within a web page, including other web pages. An iframe pretty much performs like a mini web browser within a web browser. Likewise, the content inside an iframe occurs separate from the adjacent elements.
- The basic syntax for adding this feature to a web page is as follows:  

```
<iframe src="URL"></iframe>
```
- The URL specified in the **src** attribute indicates the location of an external object or a web page.

# HTML iFrame

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <title>HTML iFrame</title>  
</head>  
<body>  
  <iframe src="https://code4sp.eu/the-project/"></iframe>  
</body>  
</html>
```



# HTML iFrame

The **height** and **width** of the iframe can be defined by applying the code disposed on the figure below. The width and height attribute values are stipulated in pixels by default, but users can also set these values in percentage, such as 50%, 100%, etc..

The default width of an iframe is 300 pixels, while the default height is 150 pixels.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Specify iFrame Dimensions in HTML</title>
</head>
<body>
  <h2>Specify Width and Height Using Attributes</h2>
  <iframe src="/examples/html/hello.html" width="400" height="200"></iframe>
  <hr>
  <h2>Specify Width and Height Using CSS</h2>
  <iframe src="/examples/html/hello.html" style="width: 400px; height: 200px;"></iframe>
</body>
</html>
```

## Specify Width and Height Using Attributes

### Hello World

This HTML document is embedded inside the current document using an iframe.

## Specify Width and Height Using CSS

### Hello World

This HTML document is embedded inside the current document using an iframe.



# HTML iFrame

As it could be verified, iframe has a border around it set by default. To modify or remove it, the best way is to use the CSS **border** feature (to be taught in the CSS topic).

An iframe can also be used as a target for the hyperlinks. It can be named using the name attribute. This means that when a link with a **target** attribute (with that name set as value) is clicked, the linked source will open in the same inline frame, as follows:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Opening Links in an iFrame</title>
  <style>
    iframe {
      width: 100%;
      height: 500px;
    }
  </style>
</head>
<body>
  <iframe src="https://code4sp.eu/the-project/" name="myFrame"></iframe>
  <p><a href="https://code4sp.eu/the-project/" target="myFrame">Open
https://code4sp.eu/the-project/</a></p>
</body>
</html>
```





# LET'S PRACTICE!

Open the following link in order to practice some of the concepts acquired so far:

[https://www.w3schools.com/html/exercise.asp?filename=exercise\\_html\\_attributes1](https://www.w3schools.com/html/exercise.asp?filename=exercise_html_attributes1)





**THANK YOU!**

**NEXT CHAPTER: HTML ADVANCED CONCEPTS**

Co-funded by the  
Erasmus+ Programme  
of the European Union

