Introduction to HTML and CSS

SoftUni Team

Technical Trainers

Software University

SoftUni

Software University

https://softuni.bg

Have a Question?

Software University

sli.do

#html-css

Table of Contents

What is HTML?

What is HTML?

- HTML is a **markup language**
- HTML is the basis for **creating web pages** and other information that can be displayed in a **web browser**
- Language for expressing **semantic structure** in textual documents

```
<p>
  <b>Document</b> content goes here…
</p>
```

Opening tag

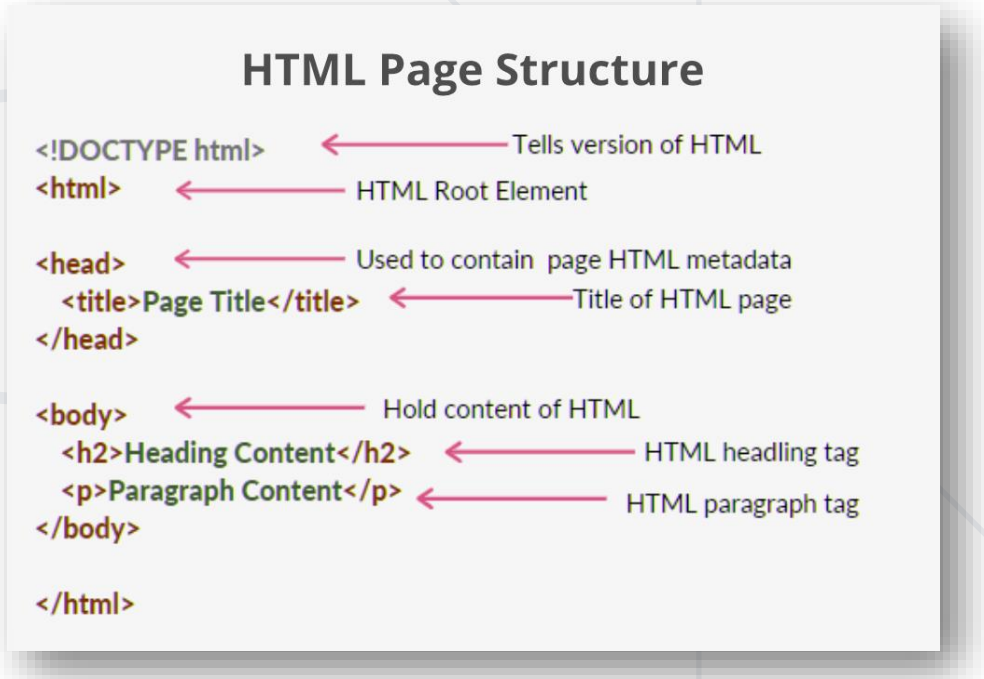
Element

Closing tag

Document content goes here…

What is HTML?

- HTML is a **language** for **describing web pages**
- HTML documents contain **HTML tags** and **plain text**
- A markup language is a set of **markup tags**
- The tags describe **document content**

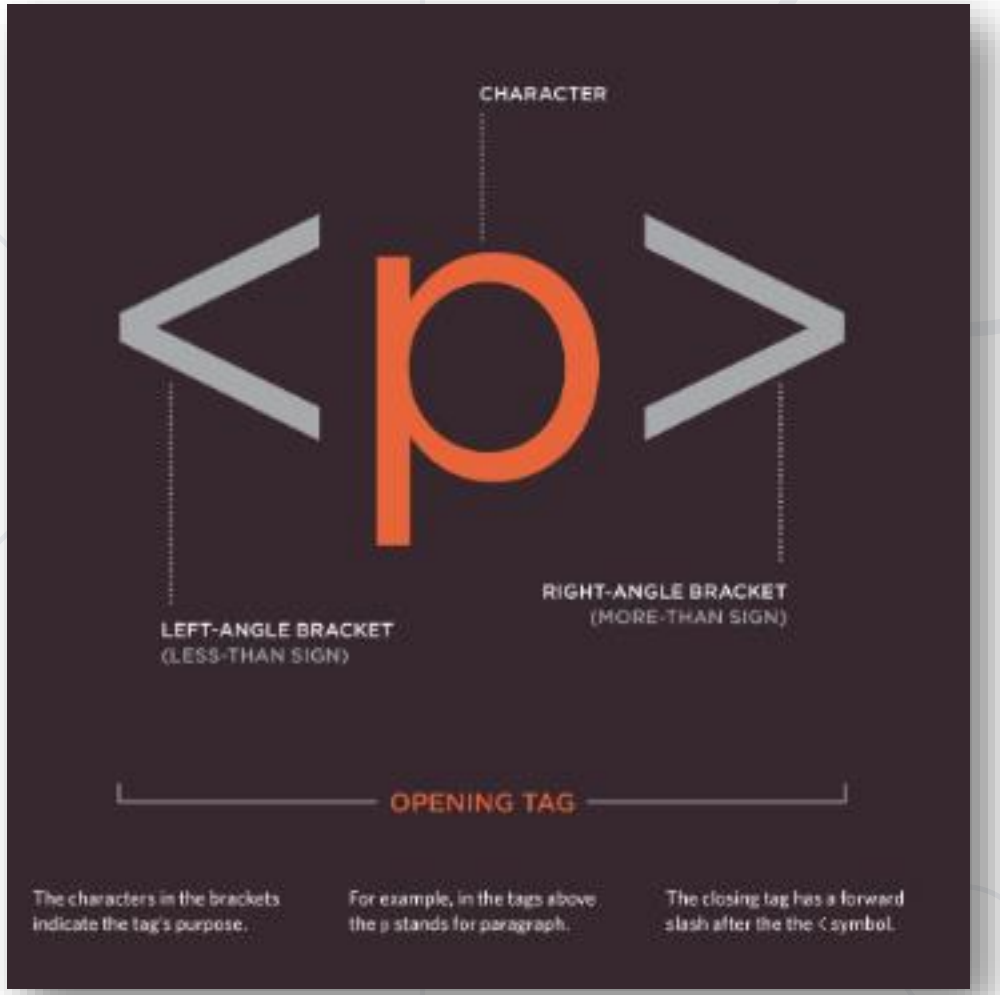


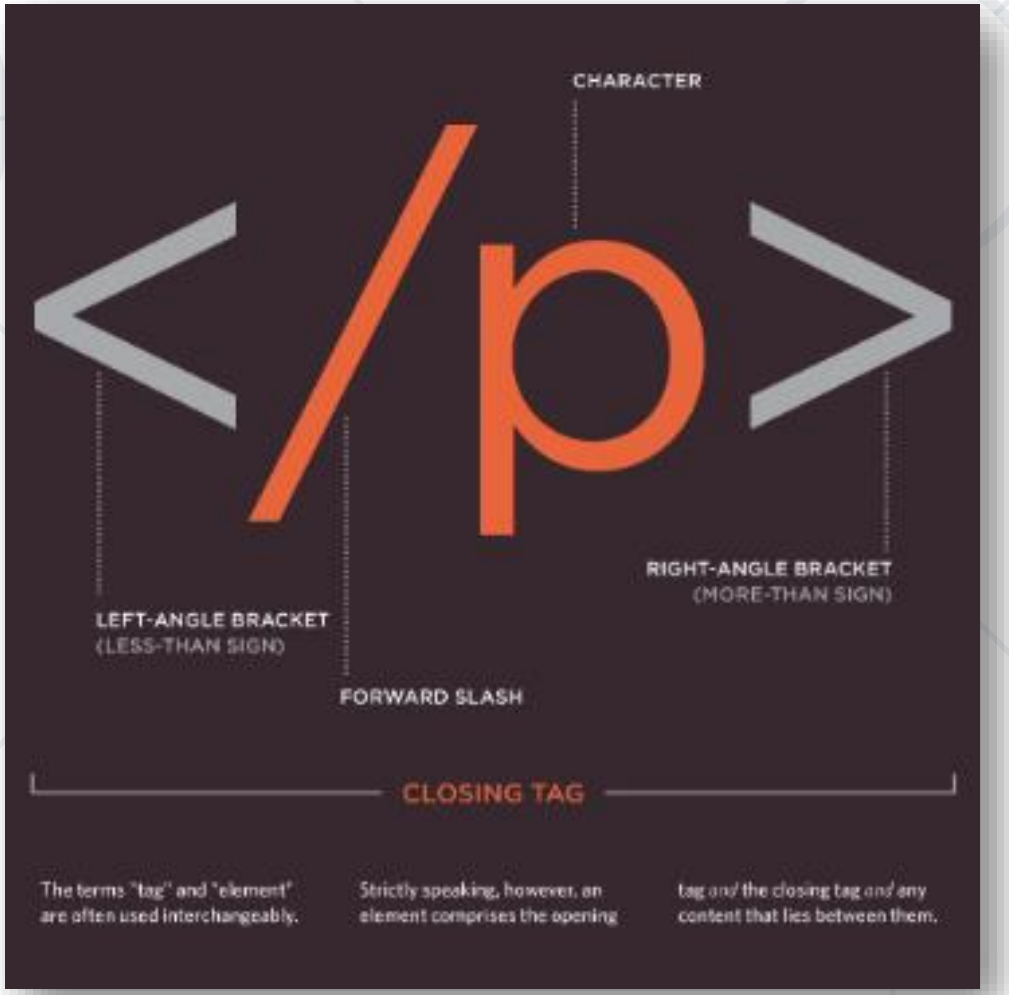
What is HTML?

- **HTML** is consumed by **web browsers**
- The purpose of a **web browser** is to read HTML documents and compose them into visible or audible web pages.
- **The browser** does not display the HTML tags, it uses the tags to interpret the content of the page.

HTML Tags

- Tags are **keywords** surrounded by **angle brackets**





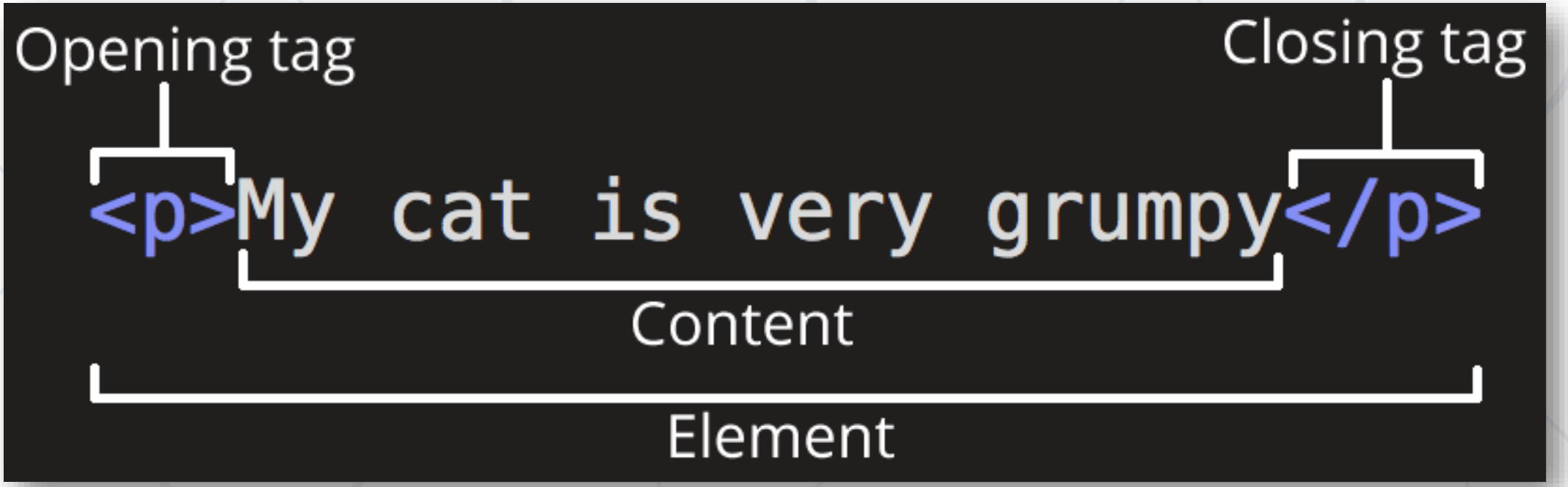
HTML Tag Pairs

- HTML tags normally come in **pairs**

```
'<' + '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, with a **forward slash** before the **tag** name
- **Start** and **end** tags are also called **opening tags** and **closing tags**

HTML Tag Pairs - Example

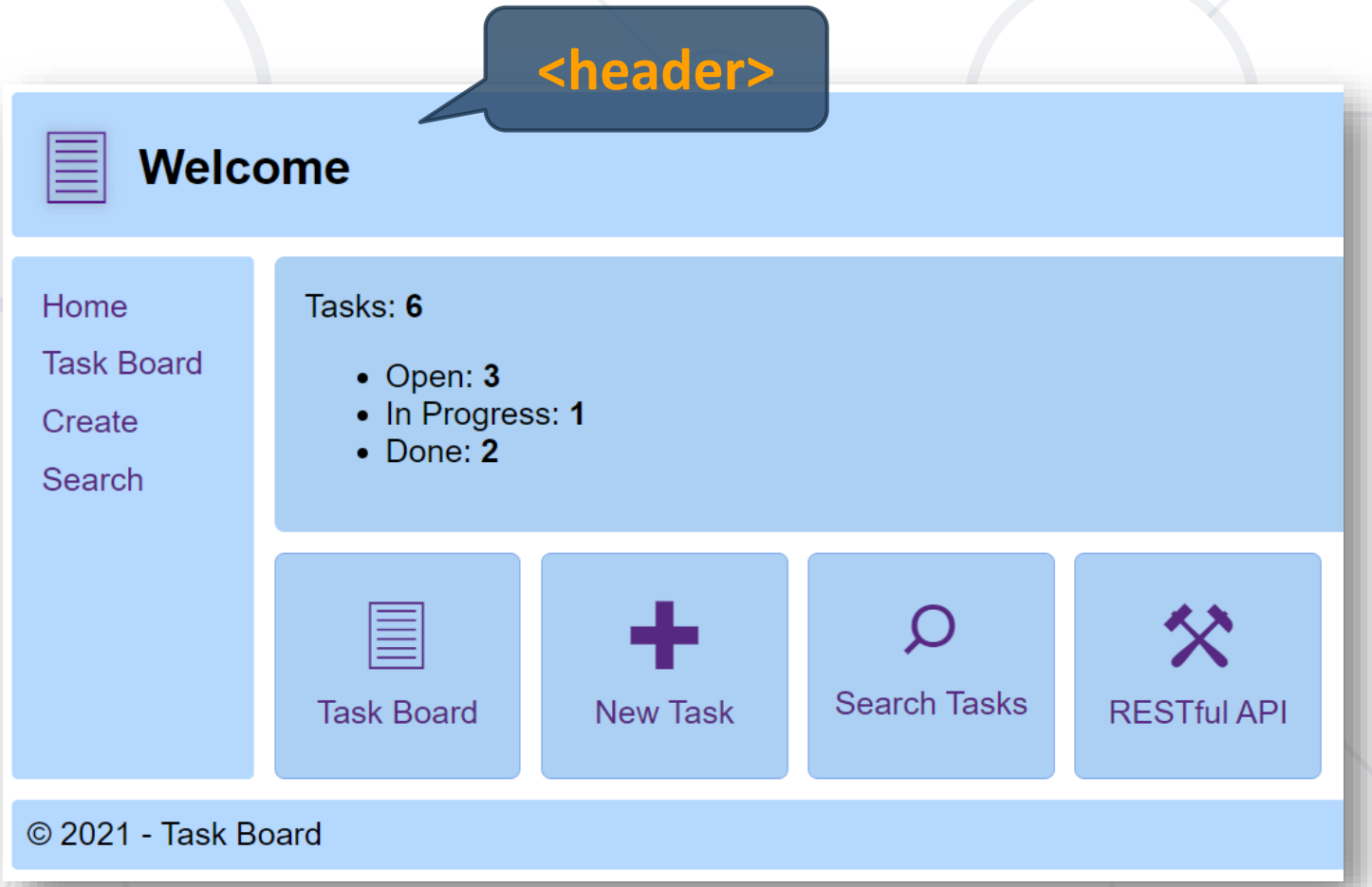


<header></header>

- Represents **introductory content**

```
<header>
    <h1>Welcome</h1>
</header>
```

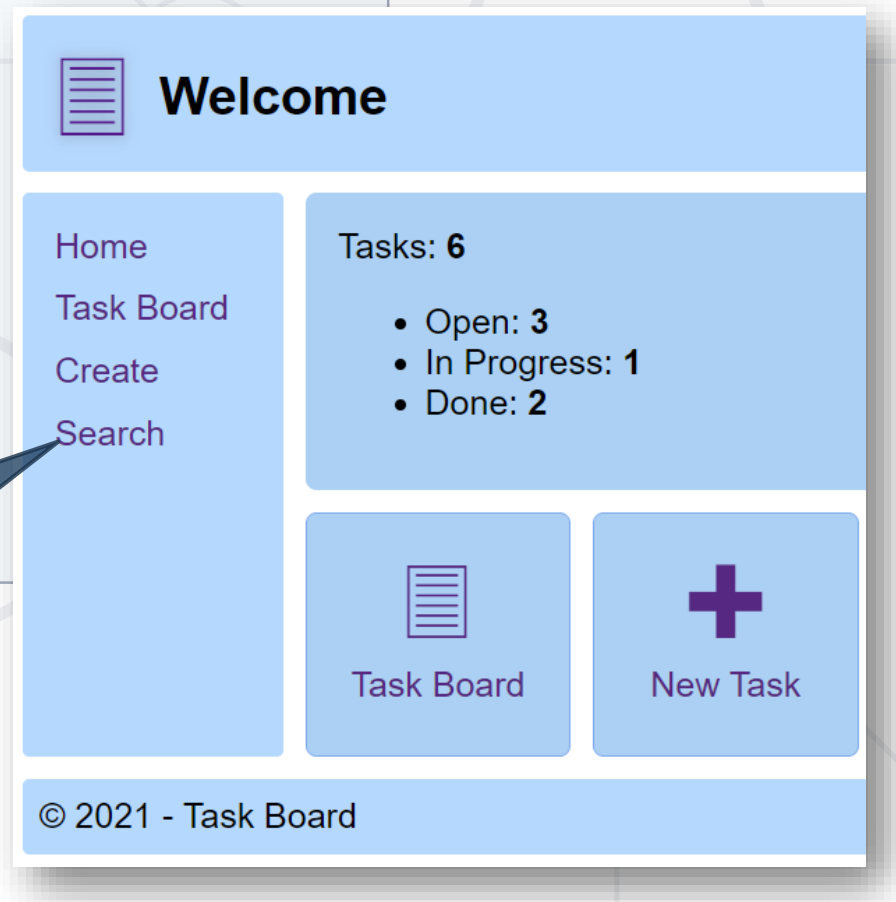
- It may contain:
  - headings
  - logo
  - search form



<nav></nav>

- Defines a set of **navigation links**

```
<nav id="leftmenu">
  <ul>
    <li><a href="/home">Home</a></li>
    <li><a href="/tasks">Task Board</a></li>
    <li><a href="/create">Create</a></li>
    <li><a href="/search">Search</a></li>
 </ul>
</nav>
```



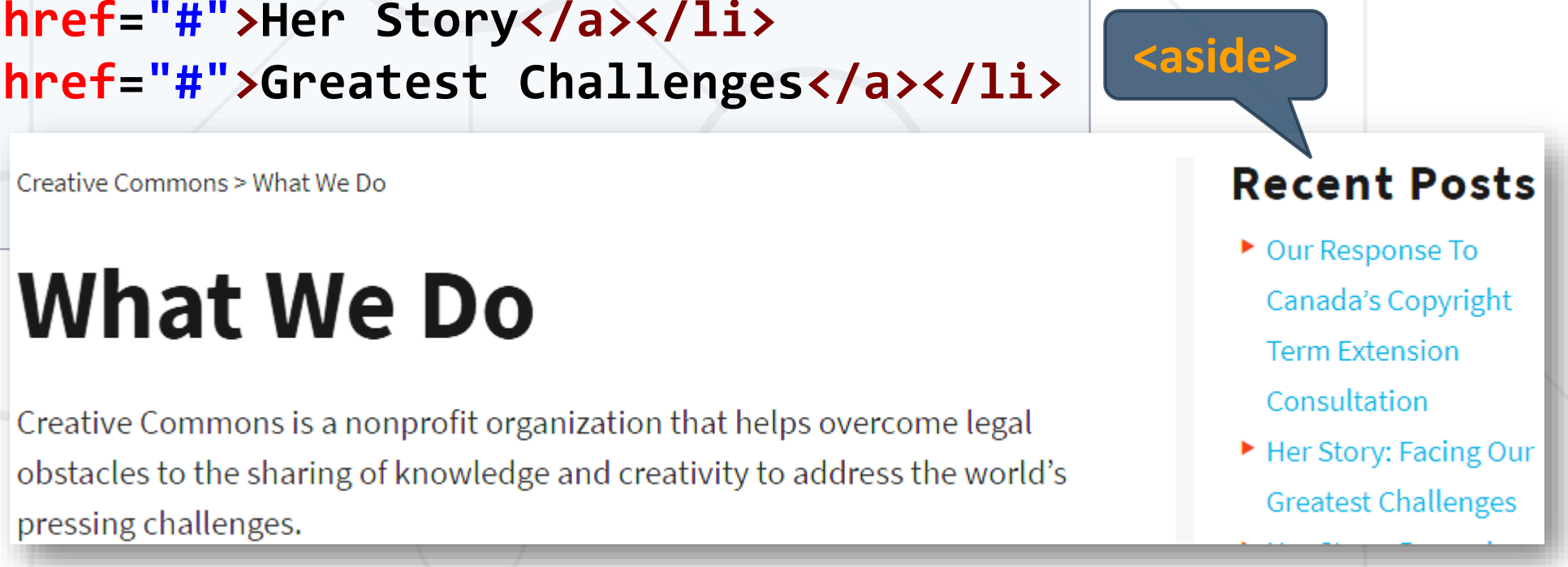
<main></main>

- `<main>` holds the main content of a document
  - Helps crawlers
  - There must not be more than one `<main>` element in a document
  - Wrap the most important information in the body

<aside></aside>

- Defines a sidebar (**left / right navigation**)

```
<aside>
  <h2>Recent posts</h2>
  <ul>
    <li><a href="#">Our Response</a></li>
    <li><a href="#">Her Story</a></li>
    <li><a href="#">Greatest Challenges</a></li>
  </ul>
</aside>
```



<footer></footer>

- A document / section footer

```
<footer>
  <p>Posted by: Hege Refsnes</p>
      <p>&copy;copyright</p>
      <p>2021 Task Board</p>
</footer>
```

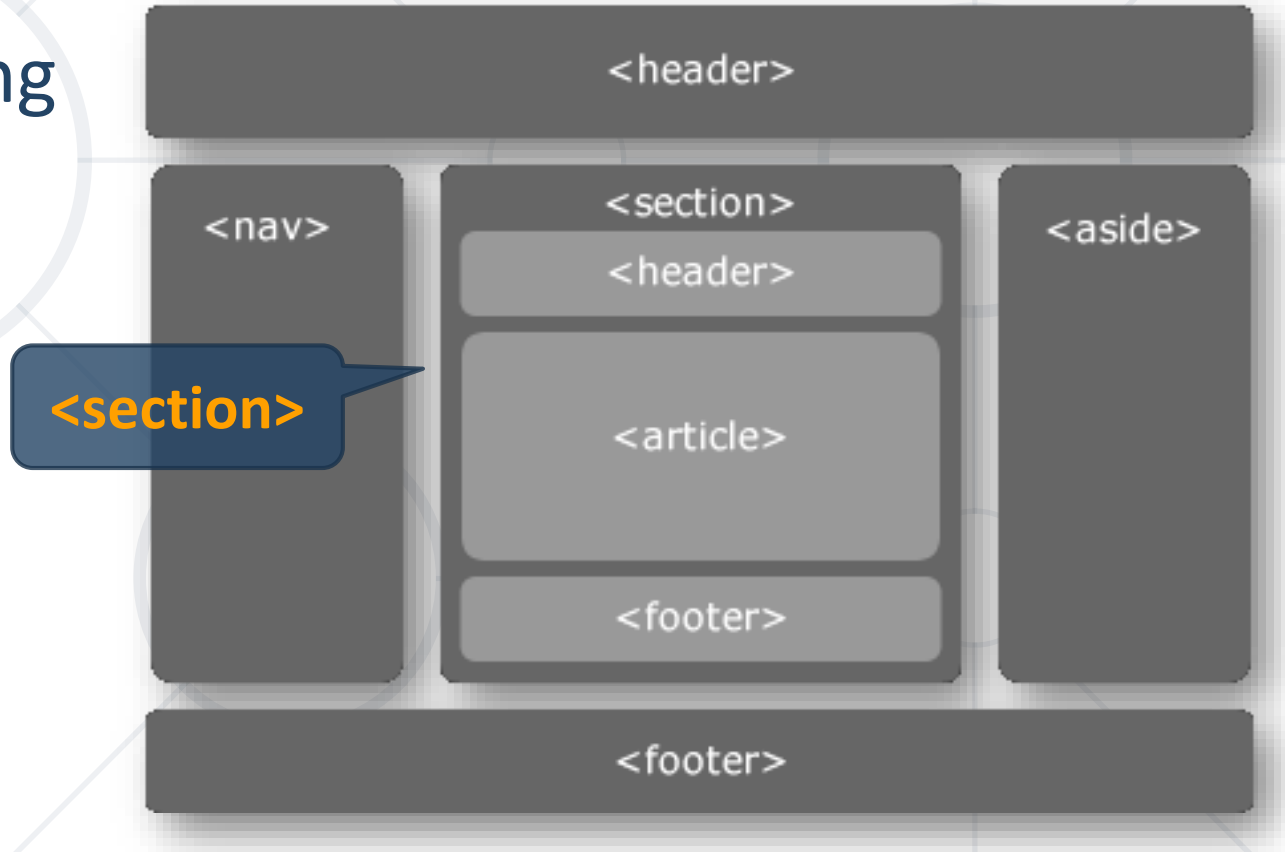
- A footer typically contains:
  - Navigation links
  - Copyright data

<section></section>

- Represents a standalone section
  - Typically followed by a heading

```
<section>
  <h2>Heading</h2>
</section>
```

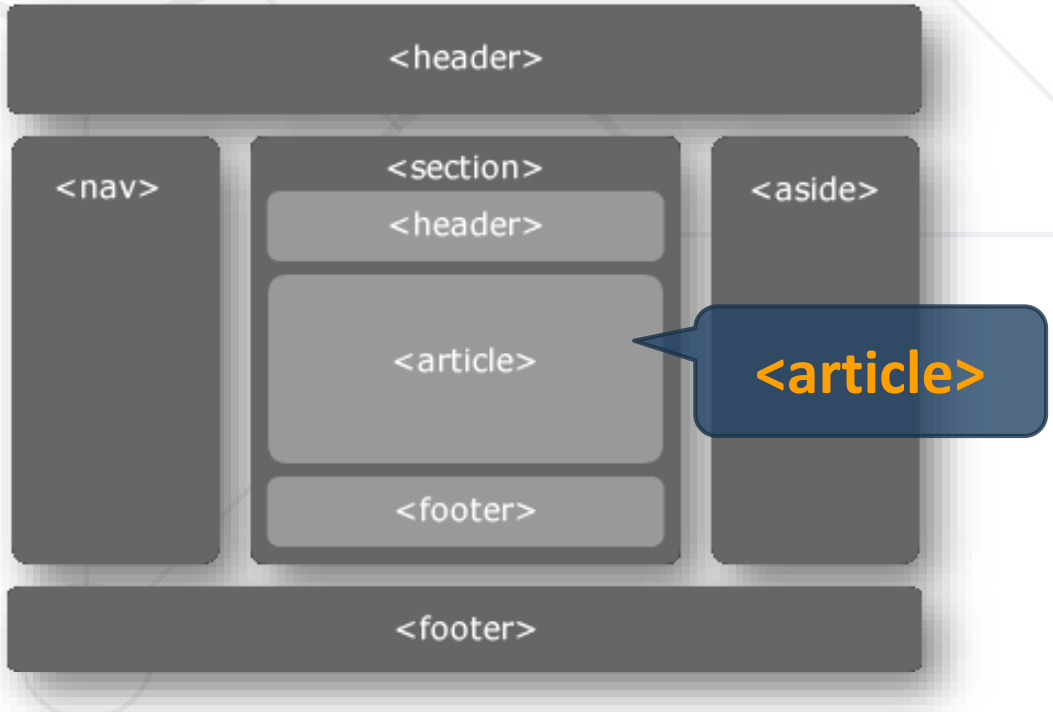
- Sections may have header, several articles, and footer



<article></article>

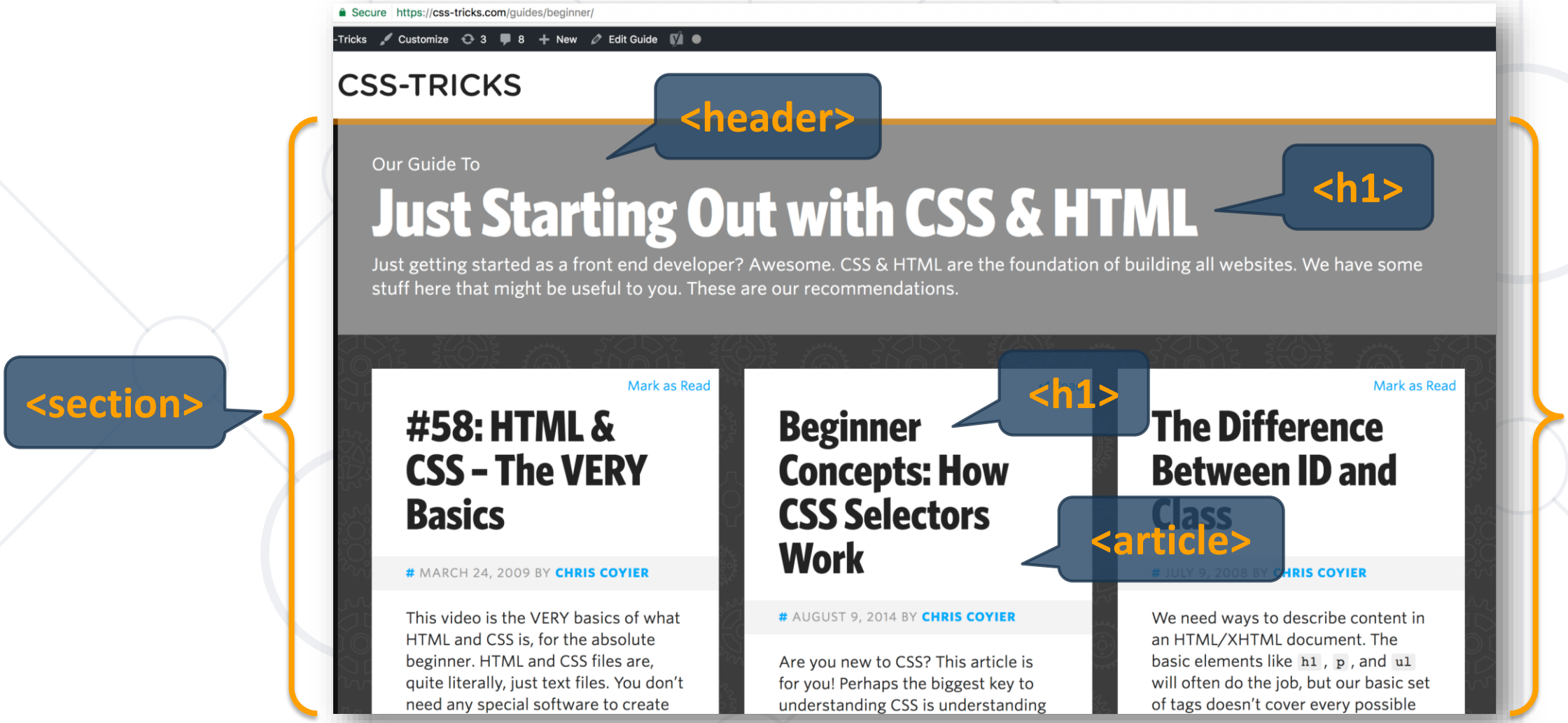
- Represents a **self-contained composition** in a document, page, application, or site
- Intended to be **independently distributable** or **reusable**
- Examples:
  - Forum post
  - Magazine
  - Newspaper article
  - Blog entry

```
<article>
  <h2>Tips</h2>
  <p>Tip #1 …</p>
</article>
```



Sections and Articles – Example

Software University



<figure></figure>

- Represents self-contained content
- Frequently with a caption "figcaption"
- Typically referenced as a single unit

```
<figure>
  <img src="pic_trulli.jpg"
    alt="Trulli">
  <figcaption>
    Fig.1 Trulli, Puglia, Italy.
  </figcaption>
</figure>
```

<figure>

<details> + <summary>

- **<details>** – additional details that the user can view or hide
- **<summary>** – defines a visible heading for the **<details>**


```
<details>
  <summary>Some details</summary>
  <p>More info about the details.</p>
</details>
```

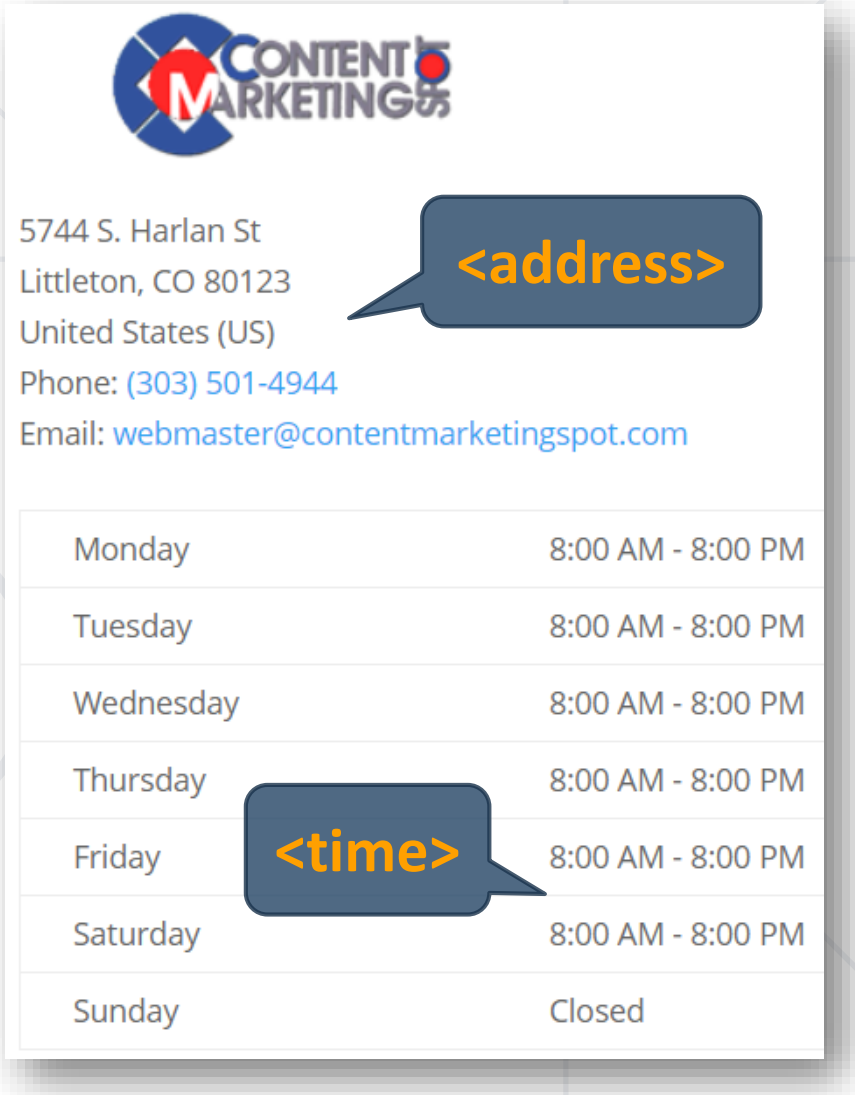
<time> + <address>

- `<time>` – a human-readable time
  - Search engines can produce smarter results

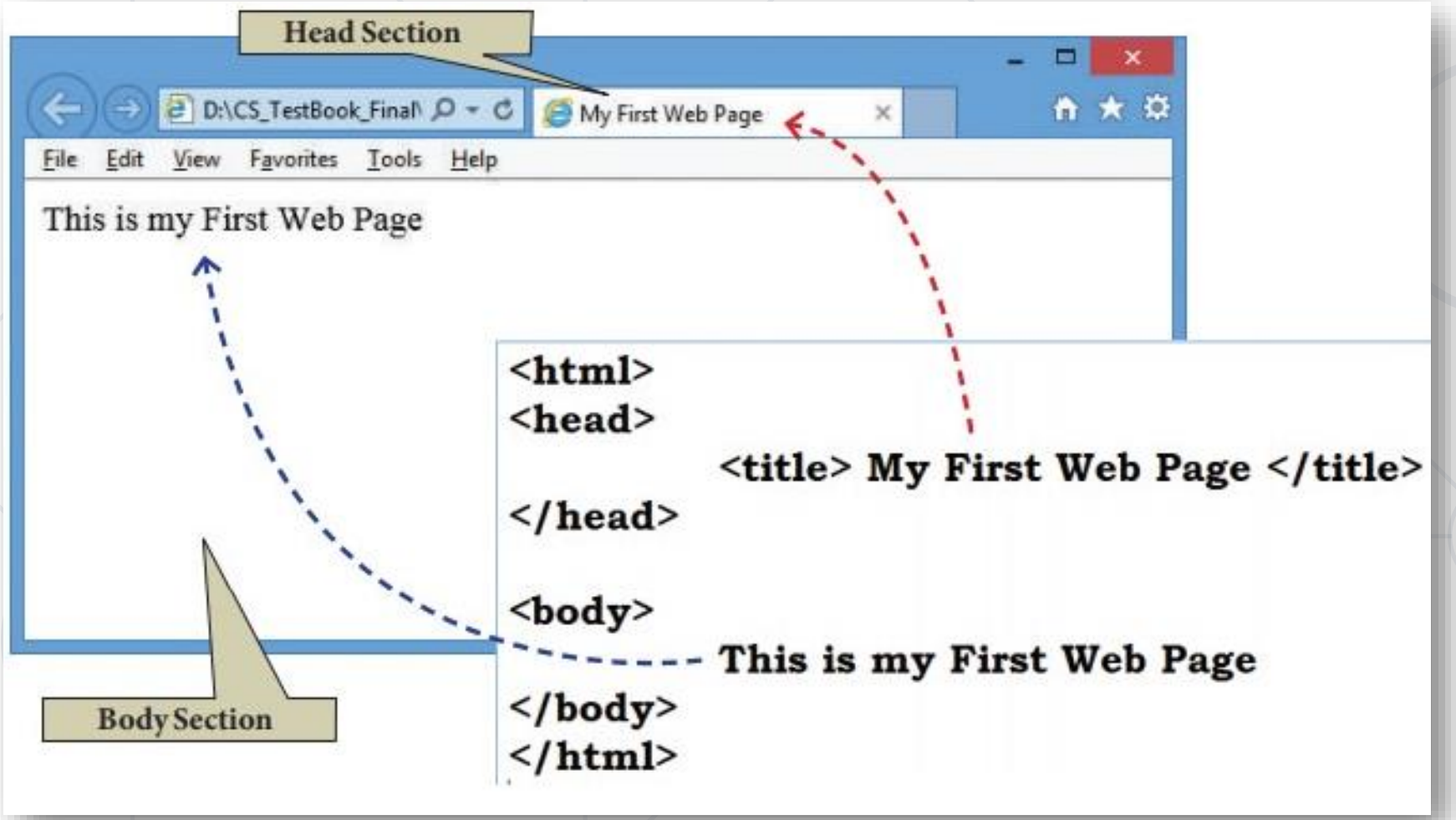
```
<p>We open at <time>10:00</time>
  every morning.</p>
```

- `<address>` – contact information for site author / owner

```
<address>
  <a href="mailto:tony@gmail.com">
    tony@gmail.com</a>
</address>
```

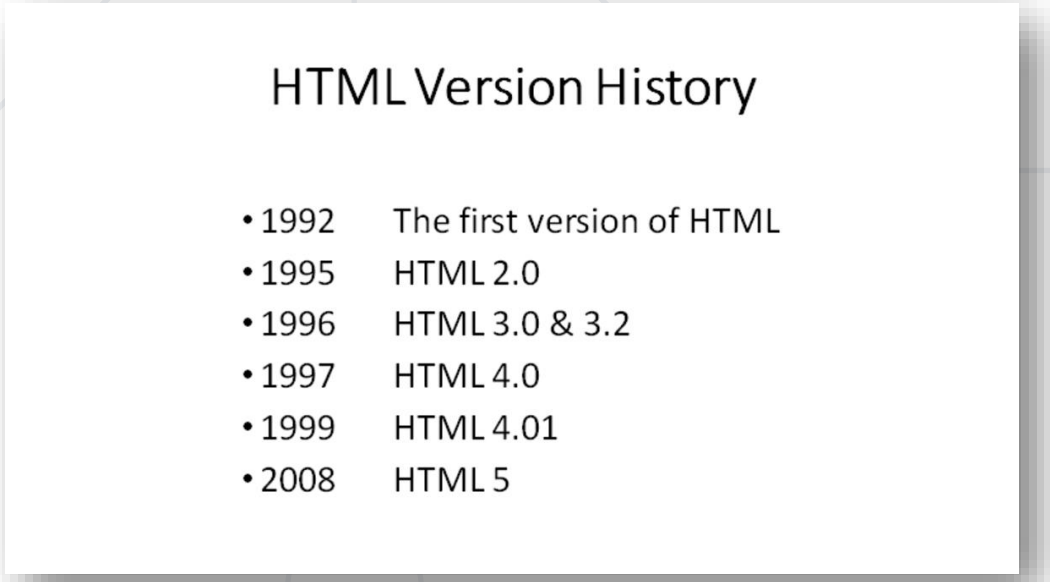


Your First HTML Document



HTML Versions

- Time passes, people think of new ways to entertain themselves
  - So **new** versions of HTML have been created
- How do we differentiate between the **different versions**?



THE <!DOCTYPE> DECLARATION

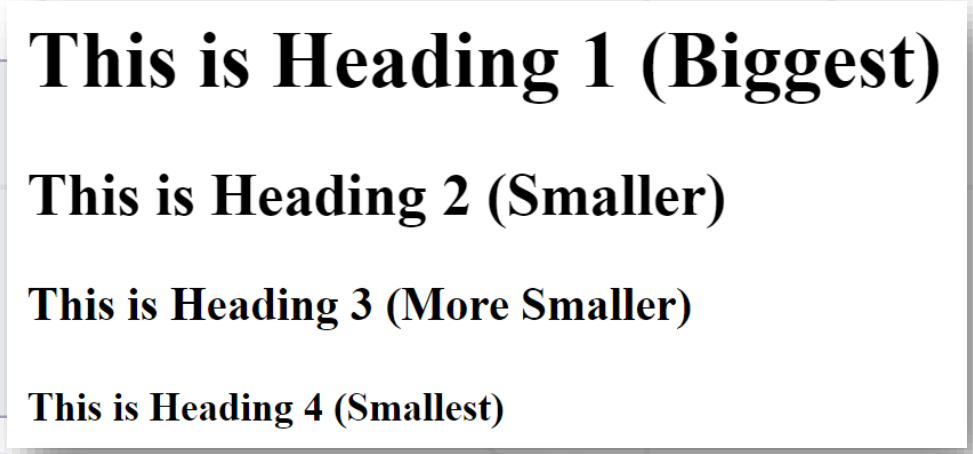
- There are many different documents on the web, and a browser can only display an HTML page **100%** **correctly** if it knows the **HTML type** and **version** used

Common HTML Tags

Headings and Paragraphs

- Headings: <h1> to <h6>

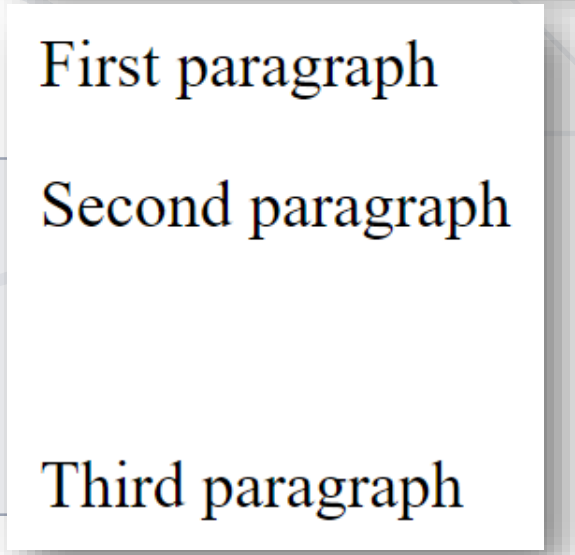
```
<h1>This is Heading 1 (Biggest)</h1>
<h2>This is Heading 2 (Smaller)</h2>
<h3>This is Heading 3 (More Smaller)</h3>
<h4>This is Heading 4 (Smallest)</h4>
```



- Paragraphs:

```
<p>First paragraph</p>
<p>Second paragraph</p>
<br /> <!-- empty line -->
<p>Third paragraph</p>
```

Comment



Hyperlinks

- External hyperlink

Specify the URL

```
<a href="https://softuni.bg">SoftUni</a>
```

SoftUni

- Local hyperlink

```
<h1 id="exercises">Exercises</h1>
…
See the <a href="#exercises" target="_blank">exercises</a>
```

Exercises

See the exercises

- Relative hyperlink

```
<a href="../2.%20HTML5-Overview.pptx">presentation</a>
```

presentation

Images

- Images are **external files**, inserted through the `<img>` tag

```
<img
  src="images/SoftUni-logo.png"
  width="400"
  height="313" />
```

- Embedded image (**data URI**)

```
<img src=" R0lGODlhEAAOAL…"/>
```

  - Example: https://codepen.io/snakov/pen/poNxXry

Ordered Lists: <ol> Tag

- Create an **O**rdered **L**ist
- Use `<ol></ol>`
  - Each holding `<li></li>`

```
<ol type="1">
  <li>One</li>
  <li>Two</li>
  <li>Three</li>
</ol>
```

1. One
2. Two
3. Three

- Attribute values for `type` are `1`, `A`, `a`, `I`, or `i`

1. One
2. Two
3. Three

A. One
B. Two
C. Three

a. One
b. Two
c. Three

I. One
II. Two
III. Three

i. One
ii. Two
iii. Three

Unordered Lists: <ul> Tag

- Create an Unordered List using :

```
<ul type="disc">
  <li>First item</li>
  <li>Second item</li>
  <li>Third item</li>
</ul>
```

- First item
- Second item
- Third item

- Attribute values for type are: disc, circle, square and none

- First item
- Second item
- Third item

- First item
- Second item
- Third item

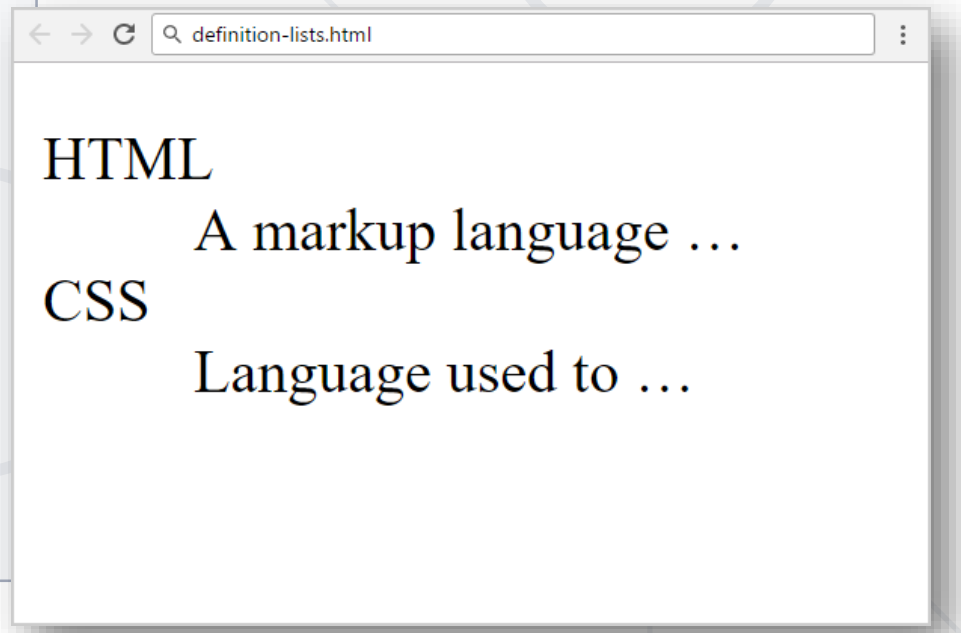
- First item
- Second item
- Third item

First item
Second item
Third item

Definition Lists: <dl> Tag

- Create definition lists using **<dl>**
  - Holds **terms** (**<dt>**) with their **definitions** (**<dd>**)

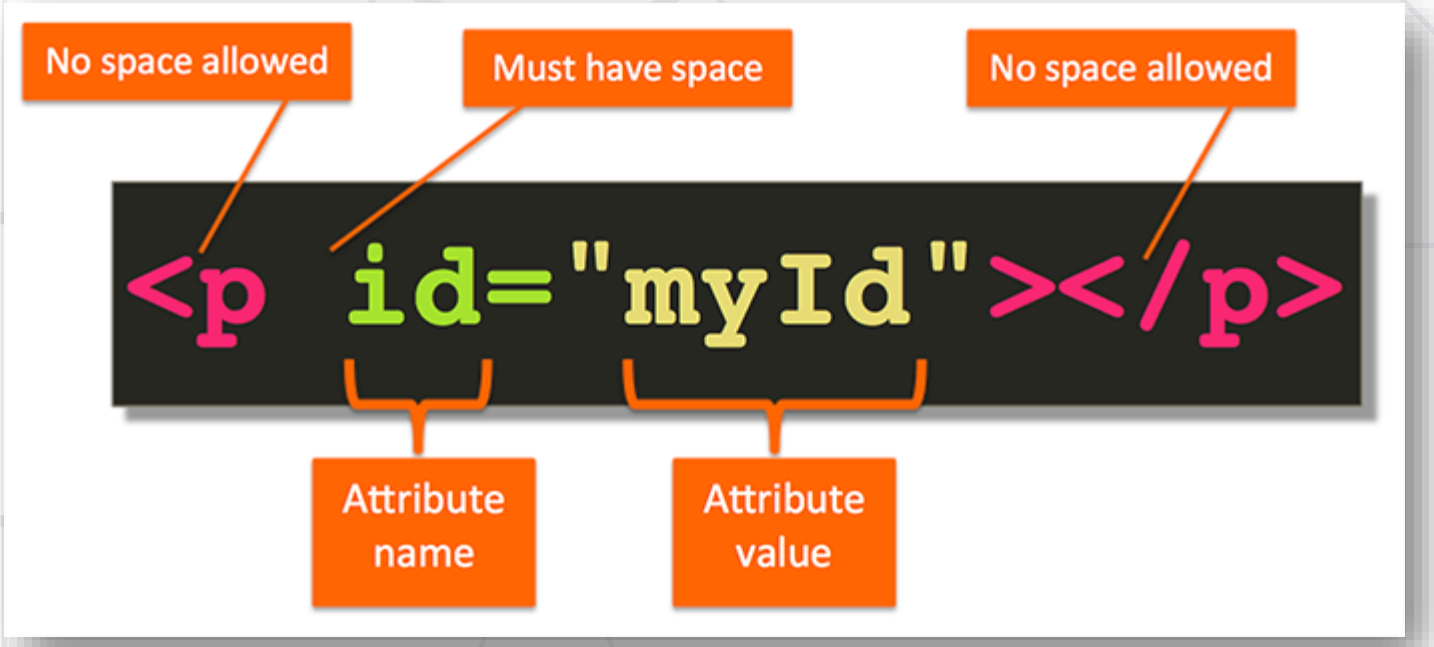
```
<dl>
  <dt>HTML</dt>
  <dd>A markup language …</dd>
  <dt>CSS</dt>
  <dd>Language used to …</dd>
</dl>
```



Common HTML Attributes

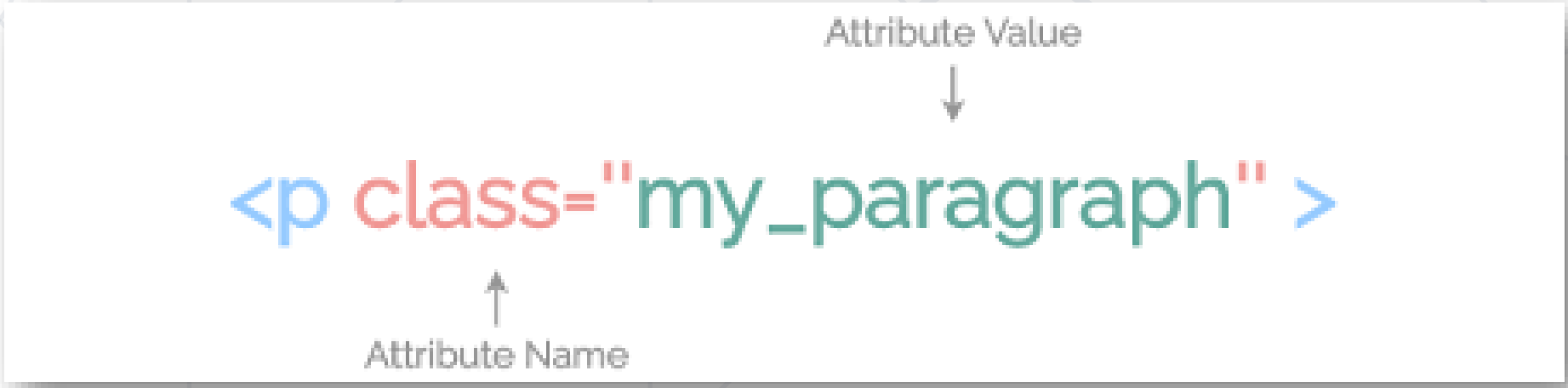
HTML Tag Attributes

- **Attributes** provide **additional** information about HTML elements
- Tags elements can have **attributes**



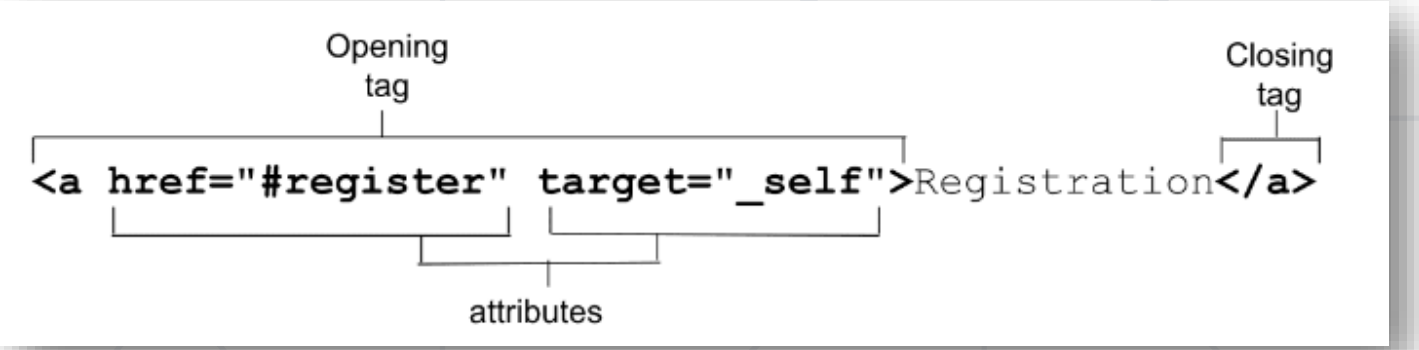
HTML Tag Attributes

- Attributes provide **additional** information about an **element**
- Attributes are **always** specified in the **start tag**
- Attributes come in **name**/**value pairs** like - **name="value"**

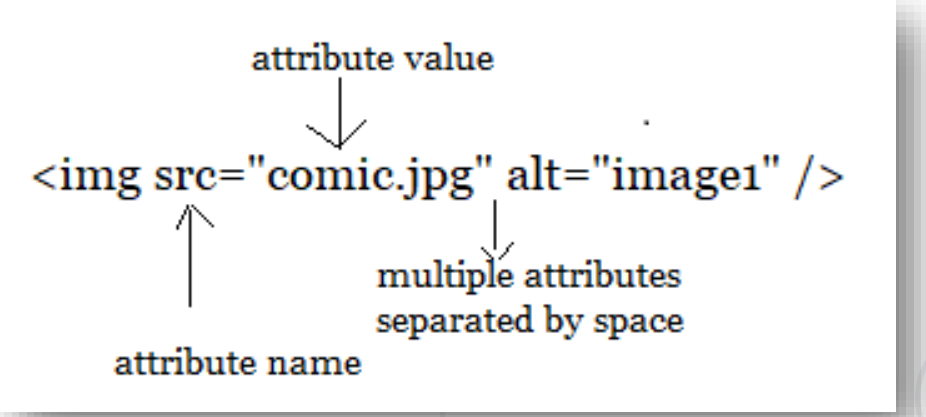


HTML Tag Attributes

- **href** – gives the tag the location information for the link



- **src** – tells the tag where to look for the image file



HTML Tag Attributes

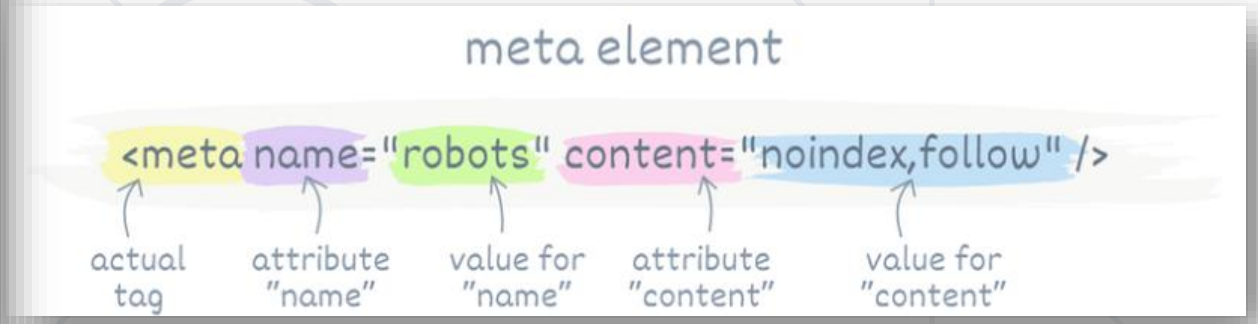
Software University

- Some tips
  - Always **Quote Attribute Values**. Attribute values should always be **enclosed** in quotes
  - **Double style quotes** are the **most common**, but **single style quotes** are also allowed
  - Be careful when **combining single** and **double** quotes, make sure you use **only one type**
  - Reference Documentation

HTML Metadata Section

- The following tags can be added to the head section:
  - <title>, <style>, <meta>, <link>, <script>, <noscript>
- Reference Documentation

```
<head>
    <meta charset="UTF-8">
    <title>Page Title</title>
    <base href="https://www.tutorialstonight">
    <link rel="stylesheet" href="style.css">
    <style>
      /* CSS properties */
    </style>
    <script src="script.js"></script>
    <script>
      // Javascript code
    </script>
</head>
<body>
```



Metadata Section

- Title

```
<head>
      <title>HTML Document title</title>
</head>
```

- Link
  - The <link> tag defines the relationship between a document and an external resource

```
<head>
      <link rel="stylesheet" type="text/css" href="mystyle.css">
</head>
```

Metadata Section

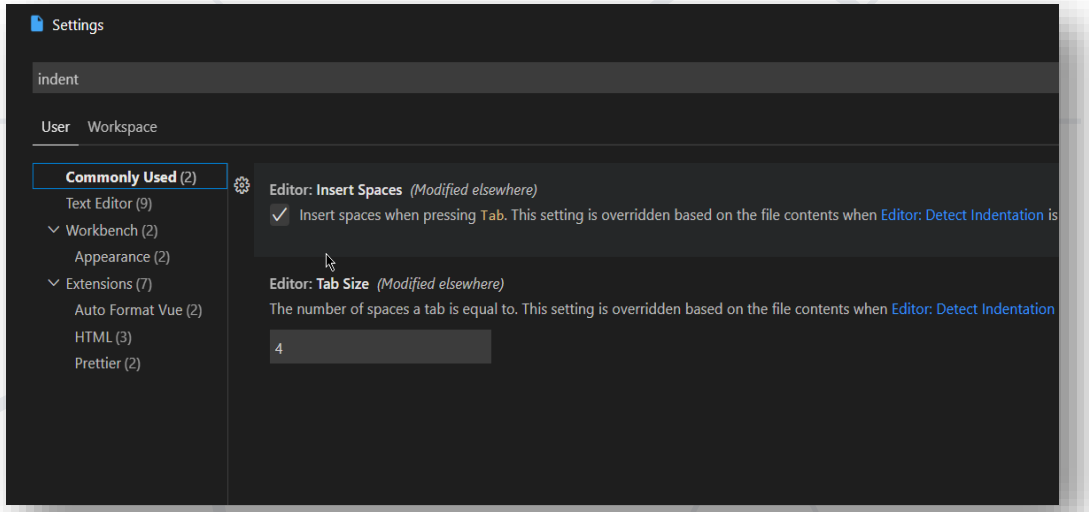
- Meta

```
<head>
      <!-- Define keywords for search engines: -->
      <meta name="keywords" content="HTML, CSS, XML, XHTML,
JavaScript">

      <!--Define a description of your web page:-->
      <meta name="description" content=”Courses on HTML and CSS">

      <!--Define the author of a page:-->
      <meta name="author" content=”Koko">
</head>
```

Indentation & Code formatting

Software University

- We will be using tabs that are 4 spaces long
- **Indentation is extremely important**
  - It helps with navigating the code
  - It helps to find mistakes faster
  - It makes debugging issues faster
- Bad indentation is shameful – when sharing



What is CSS?

Cascading Style Sheets

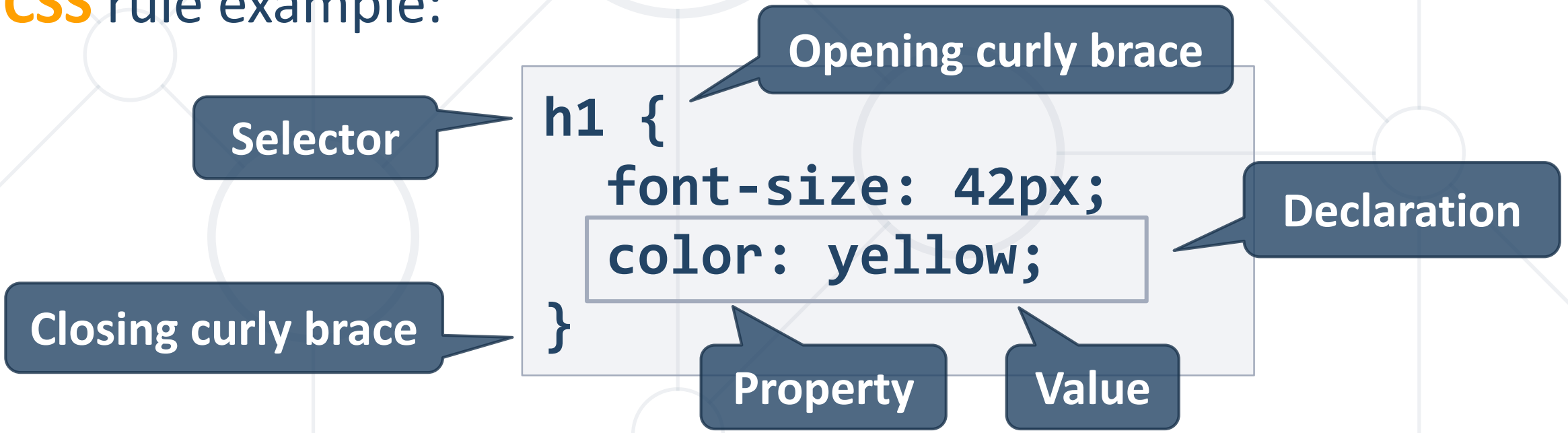
What is CSS?

Software University

- CSS stands for **Cascading Style Sheets**
- Styles define the visual presentation of HTML elements
- CSS solved a problem
- HTML was never intended to contain tags for formatting a document.
- With CSS the separation between semantic content and visual presentation can be achieved again.

What is CSS?

- **CSS** defines **styling** of the HTML elements
  - Specifies fonts, colors, margins, sizes, positioning, floating, …
  - CSS **rules** format: `selector { prop1:val1; prop2:val2; … }`
- **CSS** rule example:



Combining HTML and CSS Files (External Style)

Software University

using-css.html

```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" type="text/css"
href="styles.css">
  </head>
  <body id="modern"><p>
    This is a <span class="special">special
drink</span> for <span class="special">special
drinkers</span>.</p>
  </body>
</html>
```

styles.css

```
.special {
  font-style: italic;
  font-weight: bold;
  color: blue;
}

#modern {
  background: #EEE;
}

p {
  font-size: 24pt;
}
```

CSS Syntax – Selectors and Rules

CSS Selectors

Software University

- In CSS, selectors are used to target the HTML elements on our web pages that we want to style
- There are a wide variety of CSS selectors available, allowing for fine-grained precision when selecting elements to style. A CSS selector is the first part of a CSS Rule
- Reference Documentation

CSS Selectors

- Select elements by **name**

```
<h1>Page Title</h1>
```

```
h1 { color: blue; }
```

- Select by **class** name

```
<p class="odd">Text</p>
```

```
.odd { font-size: 10px; }
```

- Select by element **id**

```
<span id="login">Go</span>
```

```
#login { width: 150px; }
```

- Select **element** with certain **class**

```
<a class="login">Login</a>
```

```
a.login { width: 80px; }
```

CSS Selectors

- **Type selectors**
  - The CSS type selector matches elements by node name. In other words, it selects all elements of the given type within a document
  - Reference Documentation

```
[tag-name] {
      [declaration]
      [declaration]
}
```

```
h1 {
      color: red;
      font-weight: bold;
}
```

CSS Selectors

- **Class selectors**
  - The CSS class selector matches elements based on the contents of their class attribute
  - Reference Documentation

```
.[tag-name] {
      [declaration]
      [declaration]
}
```

```
.[class-name] {
      color: red;
      font-weight: bold;
}
```

CSS Selectors

Software University

- **ID selectors**
  - The CSS ID selector matches an element based on the value of the element's id attribute. For the element to be selected, its id attribute must match exactly the value given in the selector.
  - Reference Documentation

```
#[tag-name] {
      [declaration]
      [declaration]
}
```

```
#[id-name] {
      color: red;
      font-weight: bold;
}
```

CSS Selectors

- **Attribute selectors**
  - The CSS attribute selector matches elements based on the element having a given attribute explicitly set, with options for defining an attribute value or substring value match
  - Reference Documentation

```
[attribute="value"] {
      [declaration]
      [declaration]
}
```

```
[href="https://softuni.bg"] {
      color: red;
      font-weight: bold;
}
```

CSS Selectors

- **Universal selector**
  - The CSS universal selector (*) matches elements of any type.
  - Reference Documentation

```
* {
      [declaration]
      [declaration]
}
```

```
* {
      color: red;
      font-weight: bold;
}
```

Combined CSS Selectors

```
<section id="news">
  <h1>Hot News</h1>
  <article>
    <h1>New Release!</h1>
    <p>Today we released
      <b class="red">ver. 7</b>
      of our unique software …
    </p>
  </article>
  <p>Published: <span
class="date">1/1/2021</span></p>
</section>
```

```
#news > h1
```

```
#news article > h1
```

```
#news article p
```

```
#news p > b.red
```

```
#news span.date
```

```
article>p, section>p
```

CSS Combinators

- **Descendant combinator**
  - The descendant combinator — typically represented by a single space (" ") character — combines two selectors such that elements matched by the second selector are selected if they have an ancestor (parent, parent's parent, parent's parent's parent, etc.) element matching the first selector
  - Reference Documentation

```
header p {
      color: red;
      font-weight: bold;
}
```

CSS Combinators

- **Selector list**
  - The CSS selector list (,) selects all the matching nodes. A selector list is a comma-separated list of selectors
  - Reference Documentation

```
header, p, div {
      color: red;
      font-weight: bold;
}
```

CSS Combinators

- **Child combinator**
  - The child combinator (>) is placed between two CSS selectors. It matches only those elements matched by the second selector that are the direct children of elements matched by the first
  - Reference Documentation

```
header > p {
      color: red;
      font-weight: bold;
}
```

Adding CSS to our HTML Document

Adding CSS to our HTML documents

Software University

- There are three ways of inserting a style sheet:
  - External style sheet
  - Internal style sheet
  - Inline style

Adding CSS to our HTML documents

Software University

- **External Style Sheet**
  - An external style sheet is ideal when the style is applied to many pages. With an external style sheet, you can change the look of an entire Web site by changing one file. Each page must link to the style sheet using the tag
  - The tag goes inside the head section:

```
<head>
      <link rel="stylesheet" type="text/css" href="mystyle.css">
</head>
```

Adding CSS to our HTML documents

Software University

- **Internal Style Sheet**
  - An internal style sheet should be used when a single document has a unique style. You define internal styles in the head section of an HTML page, by using the style tag

```
<head>
      <style>
            body {
                  …
            }
      </style>
</head>
```

Adding CSS to our HTML documents

Software University

- **Inline Styles**
  - An inline style loses many of the advantages of style sheets by mixing content with presentation
  - Do not use this method unless you have no other choice!
  - *To use inline styles, you use the style attribute in the relevant tag. The style attribute can contain any CSS property*

```
<div style="color: red;">
      …
</div>
```

Inline CSS Style

- The `style` attribute defines **inline CSS**

Attribute "style"

```
<h1 style="color:blue">This is a blue …
</h1>
```

Property

Value


```
<h2 style="color:red; font-size:2.1em">
  This is a red …
</h2>
```

Multiple CSS declarations

Basic CSS Selectors

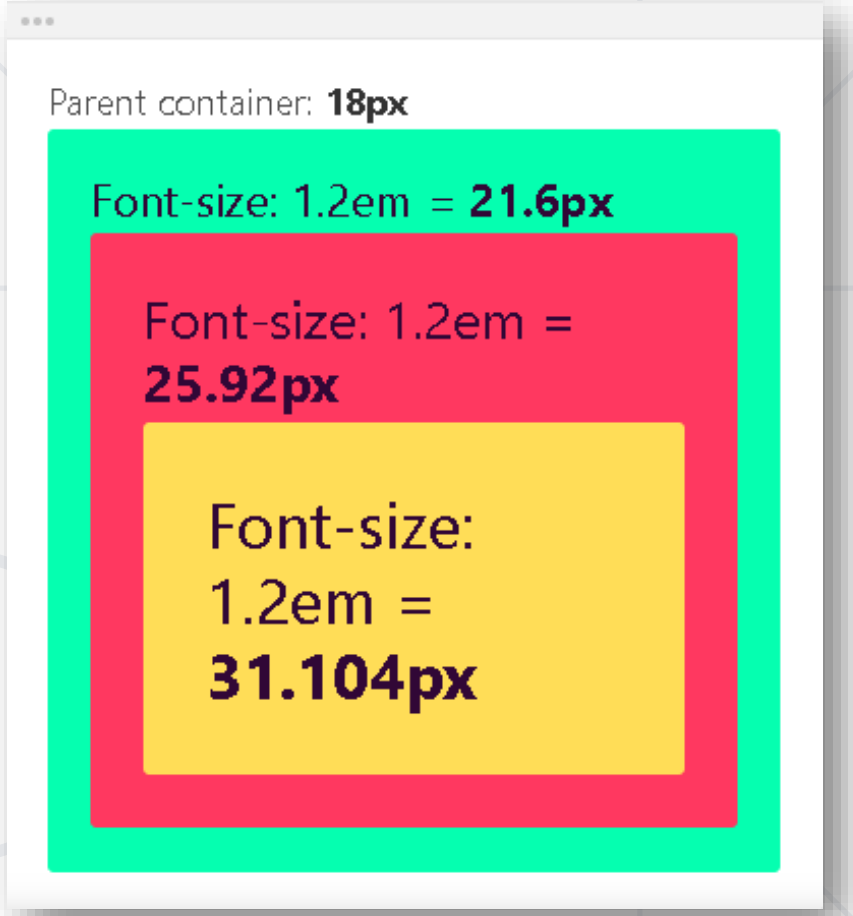
Font Size

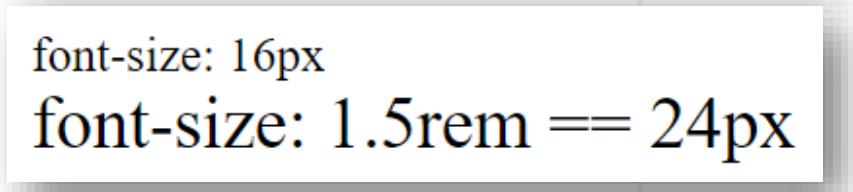
- `font-size` – defines the text size
  - `px` / `pt` values (e. g. **18px** / **24pt**)
    - 1px == 0.75pt == 1/96 inch
  - `em` values – relative to the original size, multiplied by a **scale** factor

```
font-size: 1.2em;
```

  - `rem` values – relative to the HTML root size (the `<html>` element)

```
font-size: 1.5rem;
```





Font Weight: Thin / Normal / Bold

- `font-weight` defines how weight is the font
  - Thin, normal, **bold**, or value [100 … 900]

Value	Name
100	thin
300	light
400	normal
700	bold

```
font-weight: thin;
```

```
font-weight: 300;
```

```
font-weight: 400;
```

```
font-weight: bold;
```

Font Style: Normal / Italic

- `font-style` – defines how much the text is ***slanted***
  - `normal` – the text is not slanted

```
font-style: normal;
```

Normal font style

  - `italic` – the letters are slightly slanted

```
font-style: italic;
```

Italic font style

  - `oblique` – the letters are more slanted than italic

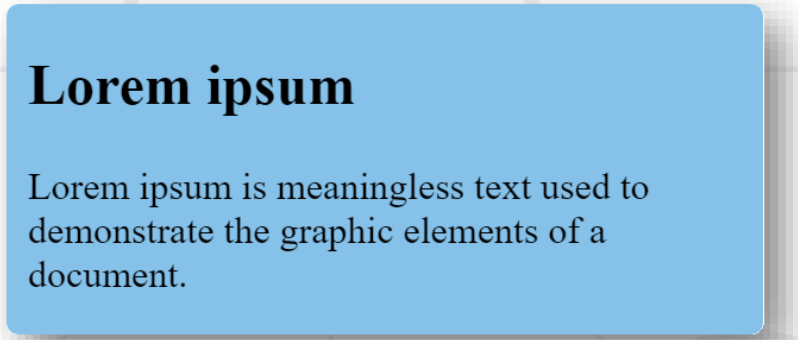
```
font-style: oblique;
```

Oblique font style

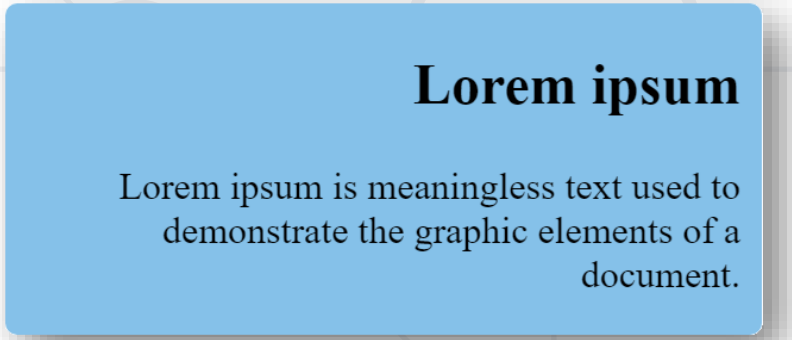
Text Align: Left / Right / Center / Justify

- `text-align`
  - Defines the **horizontal alignment**

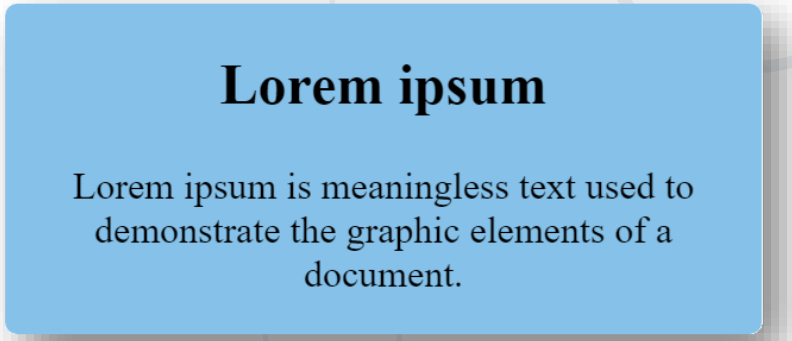
```
text-align: left;
```



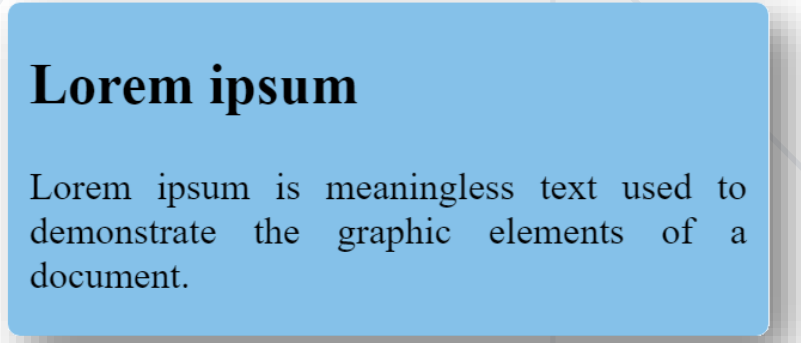

```
text-align: right;
```




```
text-align: center;
```




```
text-align: justify;
```

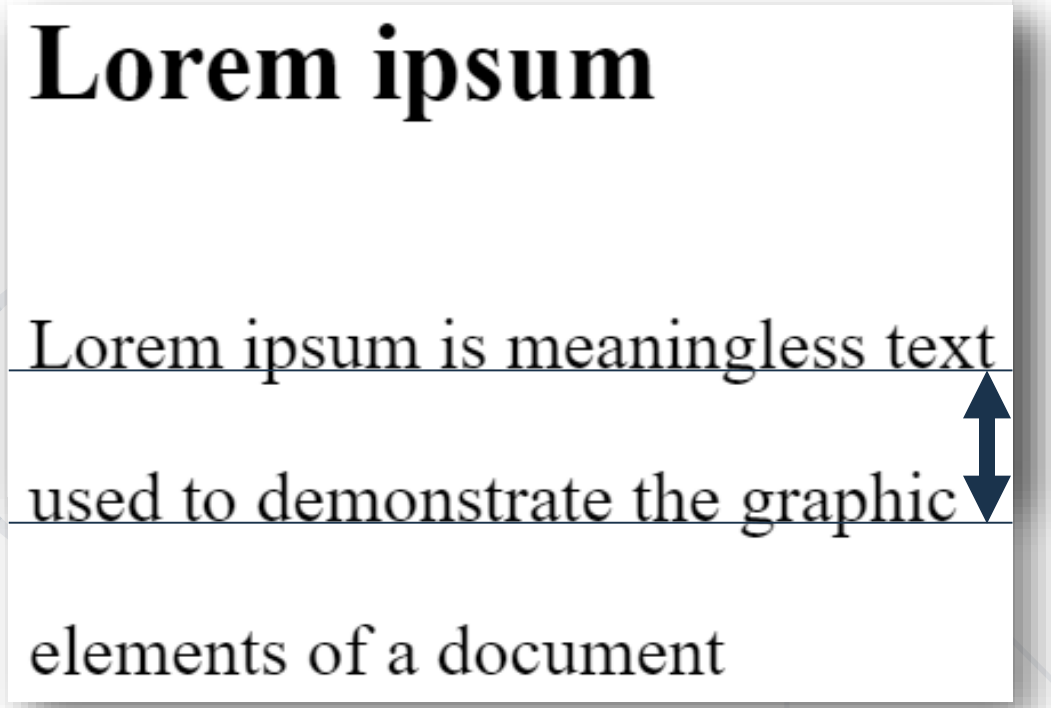


Line Height

- `line-height` – defines the **height** of a single line of text
  - Measures: **unitless** / **pt** / **px** / **em** / **rem**

```
<article>
  <h1>Lorem ipsum</h1>
    <p> Lorem ipsum is
meaningless text used to
demonstrate the graphic
elements of a document.</p>
</article>
```

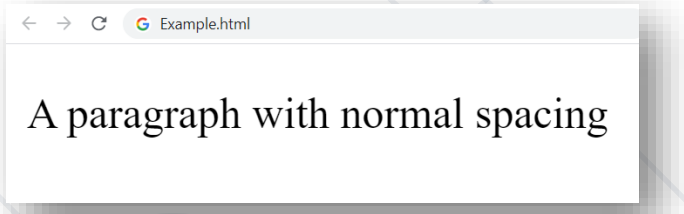
```
p { line-height: 2.1; }
```



Letter Spacing

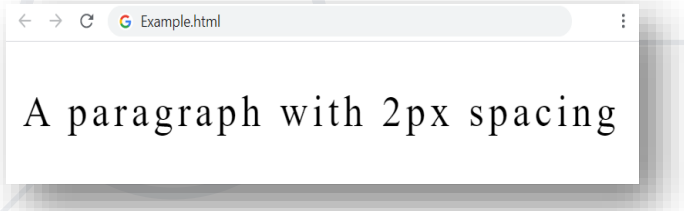
- `letter-spacing` - defines the spacing between the characters of a block of text
  - `normal` - the spacing between the characters is normal

```
letter-spacing: normal;
```



  - Using `pixels`

```
letter-spacing: 2px;
```



Text Decoration

- `text-decoration` - defines how the text content of the element is decorated: `overline`, `underline`, `line-through`
  - `none` - removes any text decoration

```
text-decoration: none;
```

  - `line-through` - draws a line across the text

```
text-decoration: line-through;
```

~~text~~

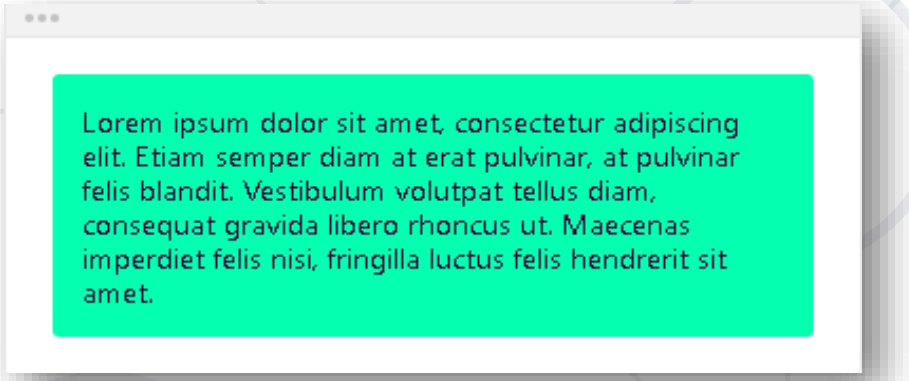
Text Indent

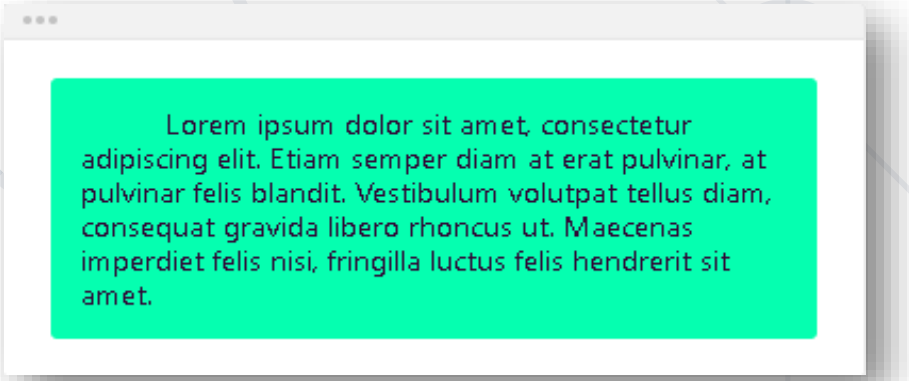
- `text-indent` - defines the indentation of the element's first line of text
  - The text is not indented

```
text-indent: 0;
```

  - The text is indented

```
text-indent: 40px;
```

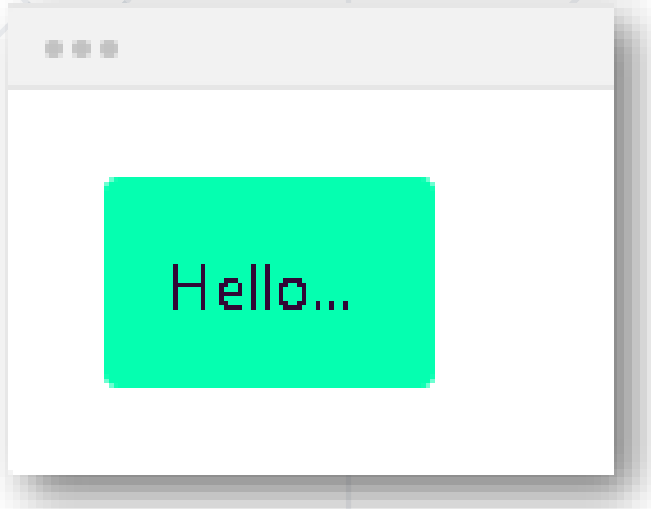




Text Overflow

- `text-overflow` - defines how the hidden text content behaves if it's overflowing
  - `ellipsis` - the overflowing content is replaced by `. . .`

```
text-overflow: clip;
```



Text Transform

- `text-transform` - specifies how to capitalize text
  - `capitalize` - turns the **first letter** of each word into a capital letter

```
text-transform: capitalize;
```

It Is A Long Established Fact That A Reader Will Be Distracted.

  - `uppercase` - turns all characters to uppercase

```
text-transform: uppercase;
```

IT IS A LONG ESTABLISHED FACT THAT A READER WILL BE DISTRACTED.

  - `lowercase` - turns all characters to lowercase

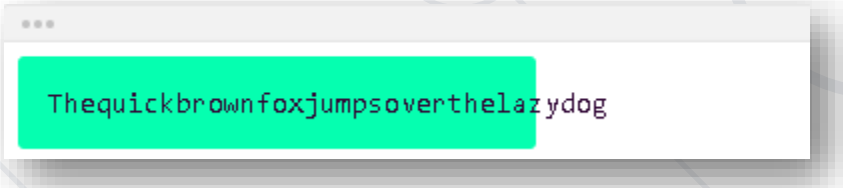
```
text-transform: lowercase;
```

it is a long established fact that a reader will be distracted.

Word Break

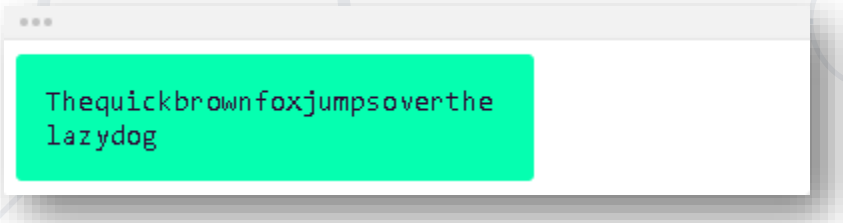
- `word-break` - defines how words should break when reaching the end of line
  - `normal` - words with no space will **NOT** break

```
word-break: normal;
```



  - `break-all` - words with no space will **break** as soon as they reach the end of a line

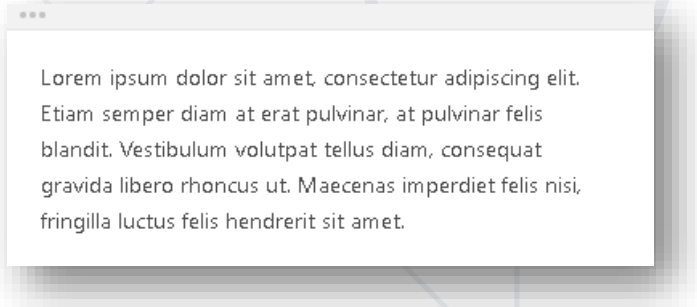
```
word-break: break-all;
```



Text Shadow

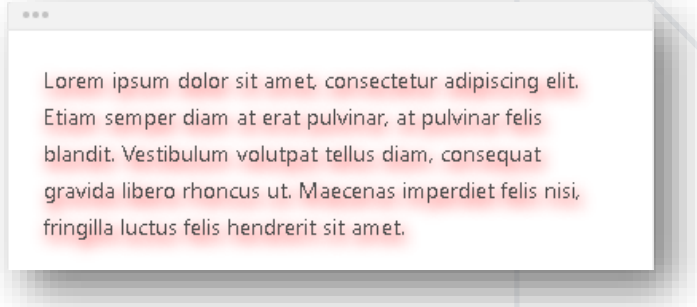
- `Text-shadow` - defines the shadow of the text content
  - None - the text content has **no shadow**

```
p {
    text-shadow: none;
}
```



  - Text-shadow: `<horizontal>` `<vertical>` `<blur>` `<color>`

```
p {
    text-shadow: 2px 4px 10px red;
}
```



Text Color

- 140+ **predefined color names** (e. g. **green**, **red**, **blue**, **coral**, …)

```
color: red;
```

Lorem ipsum

- **Hexadecimal** color code in format **#RGB** or **#RRGGBB**:

```
color: #05ffb0;
```

Lorem ipsum

- Decimal `rgb()` color codes (red, green, blue values):

```
color: rgb(125, 125, 255);
```

Lorem ipsum

- Decimal `rgba()` color codes (red, green, blue, alpha opacity):

```
color: rgba(255, 0, 0, 0.5);
```

Lorem ipsum

Background Color

- `background-color` - defines the color of the background
  - `transparent`

```
background-color: transparent;
```

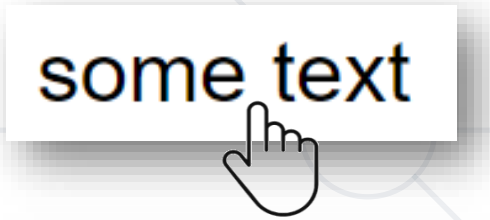
  - Specify the background color with:
    - HEX
    - RGB/RGBA
    - Named color

```
background-color: navy;
```

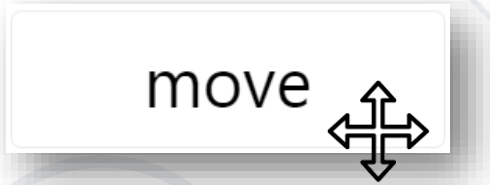
Mouse Cursor

- Sets the **mouse cursor** when hovering the element:

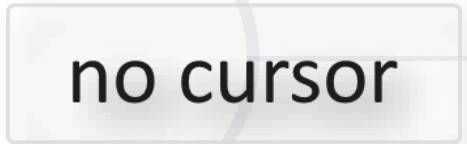
```
cursor: pointer;
```



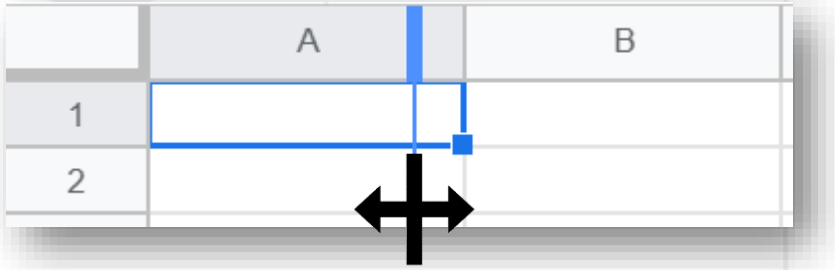

```
cursor: move;
```




```
cursor: none;
```




```
cursor: col-resize;
```



Outline

- Outline-width - defines the **width** of the element's outlines
- Outline-style - defines the **style** of the element's outlines
- Outline-color - defines the **color** of the element's outlines

```
p {
    outline: 4px dotted red;
}
```

Resources

- https://en.wikipedia.org/wiki/Markup_language
- https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/How_the_Web_works
- https://developer.mozilla.org/en-US/docs/Web/HTML
- https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/HTML_basics
- https://developer.mozilla.org/en-US/docs/Web/CSS
- https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/CSS_basics

Summary

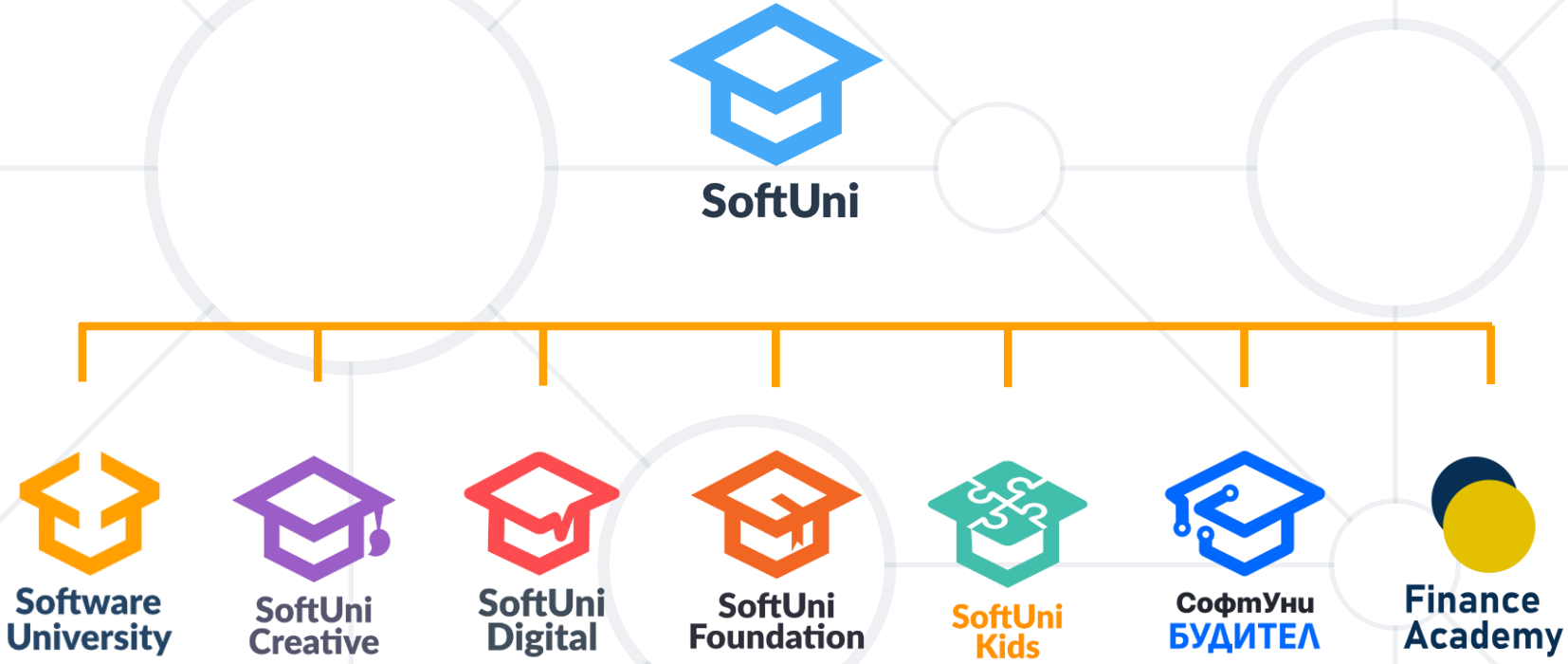
- What is HTML?

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>HTML Example</title>
  </head>
  <body>
    <h1>Hello HTML!</h1>
    <p>HTML describes formatted text using <strong>tags</strong>. Visit the
      <a href="https://softuni.bg/trainings/courses">SoftUni HTML course to
    learn more.</a></p>
  </body>
</html>
```

- CSS styles may be: external, inline, embedded

```
.my-list p { text-align: center; }
```

Questions?



SoftUni Diamond Partners

Software University

SmartIT

Coca-Cola HBC Bulgaria

SUPER HOSTING .BG

createX

Postbank

Решения за твоето утре

DRAFT KINGS

VIVACOM

AMBITIONED

INDEAVR

Serving the high achievers

SOFTWARE GROUP

PHAR VISION

License

- This course (slides, examples, demos, exercises, homework, documents, videos and other assets) is **copyrighted content**
- Unauthorized copy, reproduction or use is illegal
- © SoftUni – https://about.softuni.bg/
- © Software University – https://softuni.bg

Trainings @ Software University (SoftUni)

- Software University – High-Quality Education, Profession and Job for Software Developers
  - softuni.bg, about.softuni.bg
- Software University Foundation
  - softuni.foundation
- Software University @ Facebook
  - facebook.com/SoftwareUniversity