HTML Essential training

# Introduction

HTML is simple so it can run on all machines

# Formatting

## Regions

<p> opening tag

</p> closing tag

Elements can be nested 🡪 Document Object Model (DOM) tree (the hierarchy)

<p> Paragraph

<h1> … <h6> headlines. Choose level not on looks but on importance (looks will be changed with css)

Having a headline hierarchy across the whole team/project creates consistency

## Text

<i> visual only italics (I want to mark this as a quote)

<em> empaphasizes italics (I want to emphasize this word)

<strong> importance, seriousness, urgency (I want to emphasize this word)

<b> bold (without meaning, only style)

## Lists

<ul> Unordered list (all the same dots)

<ol> Ordered list (1, 2, 3, …)

<li> list item

<dl> Definition: key-value pairs <dt> … <dd>

## Quotes

<cite>

<blockquote> block context

<q> quote inline

Inline elemts vs block elements

## Time

<time>

<time datetime=”2025-05-08”> 8 May 2025 </time>

Also possible for hours, time and timezone

## Attributes

Add parameters to elements

<elementname attributename=”” >

## Code

How to style something to look like code

<code> inline

How to write ‘<H1>’ inside <code> without converting it to a headline?

&lt; and &gt; 🡪 HTML entities

<br> line break (needs to closing)

<pre> html will respect the spacing

## Subscripts, Superscripts

<sub> 2

<sup>

<small> mark something with small meaning (e.g. fineprint)

# Understanding the Power of HTML

## Debugging HTML

Use Dev tools in Browser to look at the DOM

## Attributes

Global Attributes that can be used on any element

<p class=””> attach class to that element so style can be changed for all instances of that class 🡪 CSS

<p id=””> reference element with unique ID 🡪 for javascript

<… contenteditable=”true”> user can edit it (which wont be saved 🡪 need javascript)

<… lang=””> language

<… dir=””> text direction

## ARIA Roles

Can be added to elements

For accessibility tree (screenreader)

<.. aria-label=”Hello World”>

## Formatting HTML

<!-- …

comments

-->

Lower case elements

Early developed elements: if not closing tag is needed, we don’t write it. Modern: <newelement />

## Weird Character

Use entity to display HTML characters: &amp; 🡪 &

Also insert special symbols

&nbsp; non-breaking space (e.g. for names)

# Links

## Links

<a href=page.html”>Link</a> anker

Can wrap around an image

Direct link: needs https…

## URL Path

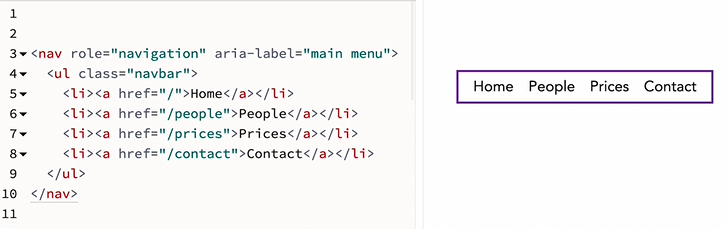
Relative link 🡪 to another page of the same page locally

../blog/march-9.html

If you just go to a folder (../images), HTML will look for an index.html file 🡪 you can have pretty links when naming the folders

href=folder/index.html = href=folder

## Navigation



<nav> marks the navigation element

List of links can also be put in footer or other places

# Images

<img src=“image.jpg“ alt=“black dog” width=”400” height=”300”>

Size in pixels 🡪 so the webpage can be created/spaced before all images are loaded

## Formats

High Quality + Small File Size

Gif: large area of single color, limited color space, can be transparent, can be animated

SVG: Logos, Vectorfile, very small file size

JPG: compress photographs (don’t put 5mb pics on webpage)

PNG: newer, has transparency

## Responsive Images

To scale for large and small screens depending on the user

Create multiple copies of the images in different resolution and let the user device decide which to download (depending on pixel density 1x – nx)

<img … srcset=”file1.jpg 2x, file2.jpg 3x, …”>

## Responsive Width

Choose file depending on width of screen

<img … srcset=”file1.jpg 480w, file2.jpg 960w, …”>

Size attribute: choose how much of the screen is occupied by the image

## Responsive Pictures

Different Photos/ratio/… for different devices

Picture element with multiple sources



## Figure and figcaption

<figure>

<img>

<figcaption> some caption </figcaption>

</figure>

In order to link the caption and the picture together