Inhaltsverzeichnis

[1 Basic HTML 2](#_Toc526962918)

[1.1 Ordnerstruktur: 2](#_Toc526962919)

[1.2 Relative Links / Absolute Links 2](#_Toc526962920)

[1.3 Kommentare 2](#_Toc526962921)

[1.4 HTML Grundstruktur 2](#_Toc526962922)

[1.5 Das kommt alles in <HEAD>: 3](#_Toc526962923)

[1.6 ID's und Klassen 3](#_Toc526962924)

[1.7 Box-Modell 3](#_Toc526962925)

[1.8 HMTL Tags 4](#_Toc526962926)

[1.9 Sectioning Elemente 6](#_Toc526962927)

[2 Basic CSS 7](#_Toc526962928)

[2.1 Arten der Deklaration 7](#_Toc526962929)

[2.2 Alles rund um die Schrift 7](#_Toc526962930)

[2.3 Positionierung 9](#_Toc526962931)

[2.4 Rahmen 9](#_Toc526962932)

[2.5 Unsortiert 9](#_Toc526962933)

[3 Basic JavaScript 10](#_Toc526962934)

[3.1 Coole Sachen 10](#_Toc526962935)

# Basic HTML

## Ordnerstruktur:

* **index.html**
* **pages**
  + kontakt.html
  + news.html
  + ueberuns.html
* **css**
  + main.css
  + navigation.css
* **img**
  + titelbild.png
  + logo.svg
* **js**
  + main.js

## Relative Links / Absolute Links

**../index.html**

//aus einem Ordner zurück ins Grundverzeichnis

**../css/main.css**

//aus einem Ordner zurück und zum Stylesheet

**img/titelbild.png**

//von index.html auf Bilder verweisen

http://ch.viva.tv

//absoluter Link zur viva Website

## Kommentare

HTML:

**<!-- Kommentar -->**

CSS:

**/\* Dieser Kommentar kann über mehrere Zeilen gehen \*/**

**// single line comment**

// .class {

text-align: center;

color: green;

}//kommentiert die komplette .class aus

.class {

text-align: center;

// color: green;

} //kommentiert nur color: green aus

## HTML Grundstruktur

<!DOCTYPE HTML>

<html>

<head>

<meta charset="utf-8">

<title>Dies ist der Titel</title>

</head>

<body>

<h1>Das hier ist eine Überschrift</h1>

</body>

</html>

## Das kommt alles in <HEAD>:

<link href="css/main.css" rel="Stylesheet">

//deklaration vom Stylesheet mit rel="stylesheet"

<link href="img/icon.png" rel="icon">

//deklaration vom Website icon mit rel="icon"

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

//dient für das Erkennen von der Grösse des Displays (Wichtig für Resonsive)

<meta name="author" content="Marcel Gertsch">

//deklariert den Autor der Website

<title>Junioren Team Bern</title>

//deklariert den Titel der Website

<script src="js/scripts.js"></script>

//deklariert, woher es das Script findet

## ID's und Klassen

**ID**

Für ein Tag kann maximal eine ID vergeben werden. Diese werden innerhalb vom Tag mittels id="NameDerID" festgelegt. ID's können für Links verwendet werden [**<a** href="#NameDerID"**>**] oder mit Javascript angesprochen werden [document.getElementById("NameDerID")].

Um ID's zu stylen spricht man diese im CSS-File mit **#NameDerID** an.

**Klassen**

Ein Tag kann x-beliebige Klassen besitzen. Diese werden dem Tag mit class="NameDerKlasse" zugewiesen. In Javascript werden Klassen mit [document.getElementsByClassName("NameDerKlasse")] angesprochen.

Um Klassen optisch zu verändern, spricht man diese im Stylesheet Mittels **.NameDerKlasse** an.

**Weder Klassen, noch ID's dürfen mit einer Zahl beginnen.**

## Box-Modell

**Content** // Inhalt der Box, in welchem Bilder und Text sind.

**Padding** // schafft Platz um den Text. Hintergrundfarbe wird übernommen.

**Border** // Rahmen halt.

**Margin** // schafft ebenfalls Platz um den Text. Unterschied: Die Hintergrundfarbe wird nicht übernommen.

Die drei Elemente, welche zum Modell gehören (P, B, M) können in jede Richtung einzeln gestylt werden. Die Reihenfolge ist oben, rechts, unten, links. Bei zwei Werten ist es oben & unten, rechts & links. Beispiel:  
margin: 5px 10px 8px 20px;

****padding: 10px 40px;

## HMTL Tags

**<a>** // wird als Link verwendet

**<a** href="https://viva.de" target="\_blank"**>**Link**</a>**

href="pages/kontakt.html" // relativer Link

"#Hierherbitte" // relativer Link zu einer ID

"https://viva.de" // absoluter Link

target="\_blank" // öffnet in neuem Tab

"\_self" // öffnet sich im selben Tab (Default)

"\_parent" // lädt die Antwort in den *frameset parent* –braucht man, falls Links im iframe sind

**<b> & <strong>** // fetter Text

**<b>**Text, der bold ist**</b>**

**<br>** // Textumbruch

**<p>**Text auf einer Zeile**<br>**zweite Zeile**</p>**

**<code>** // diese Schriftart

**<code>**Text in der Konsolenschriftart**</code>**

**<dl>** // Beschreibungsliste (description list)

**<dl>**

**<dt>**Marcel**</dt>** // Definition Begriff

**<dd>**die geilste Sau ever**</dd>** // Definition Beschreibung

**</dl>**

**<del>** // durchgestrichener Text

**<del>**Text, der durchgestrichen ist**</del>**

**<div>** // dient als Behälter für jegliche Inhalte

**<div>**

**<img** src="img/logo.svg"**>**

**</div>**

**<form>** // Formular (u dont say)

**<form>**

**<input** type="text"**>** // normaler Text (Name)

type="email" // verlangt "@" und "."

type="password" // zeigt Punkte anstelle von den Zeichen

type="color" // Color Picker

type="button" // selbsterklärend

type="checkbox" // Salami oder keine Salami?

type="file" // Um eine Datei anzuhängen

type="submit" // Absendebutton

type="reset" // löscht jeglichen Inhalt des Formulars

**<input** type="radio" name="Auswahl"**>** // möchtest du Salami oder

**<input** type="radio" name="Auswahl"**>** // Margherita? – name muss gleich sein

**<textarea></textarea>**

required // erforderlich oder nicht

name="Vorname" // Beispiel für type="text"

value="Absenden" // Defaultwert für etwas festlegen

**</form>**

**<h1>** // Überschriften

**<h1>**Haupttitel**</h1>**

**<h2>**Subtitel**</h2>**

...

**<h6>**kleinste Überschrift**</h6>**

**<hr>** // horizontale Trennlinie

**<p>**Text über der Trennung**<hr>**Text unter der Trennung**</p>**

**<i> & <em>** // kursiver Text

**<i>**Das ist ein Text in "italic"**</i>**

**<iframe>** // iframe dient dazu, andere Dokumente im aktuellen Dokument darzustellen

**<iframe** src="https://teamstocker.github.io" width="50px" height="50px "**></iframe>**

**<img>** // offensichtlich, ein Bild.

**<img** src="img/logo.svg" alt="Logo" width="50px" height="auto" draggable="false"**>**

src = Pfad zur Bilddatei

alt = Falls das Bild nicht geladen wird, erscheint dieser Text

width = Breite des Bildes (px, em, %)

height = Höhe des Bildes (px, em, %)

draggable = ziehbar oder nicht (true, false, auto)

**<mark>** // gelb markiert

**<mark>**Dies ist ein markierter Text**</mark>**

**<nav>** // keine Eigenschaften – wird für die Navigation gebraucht

**<nav>**

**<ul>**

**<li><a** href="index.html"**>**home**</a></li>**

**<li><a** href="pages/news.html"**>**News**</a></li>**

**<li><a** href="pages/team.html"**>**Team**</a></li>**

**<li><a** href="pages/saisonplan.html"**>**Saisonplan**</a></li>**

**<li><a** href="pages/kontakt.html"**>**Kontakt**</a></li>**

**</ul>**

**</nav>**

**<ol>** // geordnete Liste

**<ol** type="1" start="1"**>** // dies sind die Defaultwerte

**<li>**Element 1**</li>**

**<li>**Element 2**</li>**

**</ol>**

type="1" // Normales zählen

"A" // zählt mit grossen Buchstaben

"a" // zählt mit kleinen Buchstaben

"I" // römische grosse Zeichen

"i" // Römische kleine Zeichen

reversed // zählt vom höchsten nach unten

start="10" // beginnt bei 10 zu zählen

**<p>** // ganz normaler Text

**<p>**Hallo das ist 1 Text**</p>**

// p hat ein Default padding von 16px oben und unten

**<sub>** // tiefgestellt

**<sub>**Das ist ein tiefgestellter Text**</sub>**

**<sup>** // hochgestellt

**<sup>**Das ist ein hochgestellter Text**</sup>**

<table>

**<u> & <ins>** // unterstrichener Text

**<u>**Dieser Text ist unterstichen**</u>**

**<ul>** // ungeordnete Liste (zu gebrauchen bei Navigationen)

**<ul>**

**<li>**QL**</li>**

**<li>**1337**</li>**

**</ul>**

// jegliches styling folgt

## Sectioning Elemente

Diese Elemente haben Standardgemäss keine speziellen Eigenschaften. Sie werden verwendet, um den Code einer Website übersichtlicher zu gestalten. Folgendermassen kann ein Beispiel aussehen:

# Basic CSS

## Arten der Deklaration

.lxbauzz {-} // verändert jedes Element mit der Class lxbauzz

#triggered {-} // verändert jedes Element mit der ID triggered

a {-} // verändert jedes Element mit dem a-Tag

.lxbauzz**,** #triggered {-} // verändert jedes Element mit der Class lxbauzz und der ID triggered

p#triggered {-} // verändert jedes p Element, wenn es die ID triggered hat

.lxbauzz#triggered {-} // triggered baut auf der Klasse lxbauzz auf

a[target] {-} // jedes a-Tag mit dem Attribut "target" wird gestylt

[target="\_blank"] // jeder Tag, welches target="\_blank" beinhaltet

ol[start~="5"] // jeder ol-Tag, welcher das Attribut start mit dem Wert 5 hat

a[class^=lx] // jeder a-Tag, welcher mit ha beginnt

a[class$=lx] // jeder a-Tag, welcher mit ha endet

a[class\*=lx] // jeder a-Tag, welcher irgendwo lx beinhaltet

div p {-} // verändert jedes p Element, welches in einem div ist

div **>** p {-} // verändert jedes p Element, welches parent ein div ist

nav ul li a || nav > a // same thing

div **+** p {-} // verändert jedes p Element, welches unmittelbar nach einem div folgt

div **~** p {-} // stylt jedes p, wenn irgendwann vorher ein div benutzt wurde

a:hover // falls mit der Maus über dem a-Element ist

a:active // Wenn jemand gerade auf das Element klickt.

a:visited // aktiv, falls die Person diesen Link bereits besucht hat

a::before { content="- " } // fügt vor jedem a-Tag ein – ein

a::after { content="lul" } // fügt vor jedem a-Tag ein lul ein

p:first-child {-} // stylt immer das erste p Element pro parent

p:last-child {-} // stylt immer das letzte p Element pro parent

p:first-of-type {-} // stylt das aller aller erste p Element

li:nth-child(2) {-} // stylt das zweite Element einer Liste

li:nth-child(even) {-} // stylt jede Zeile mit einer geraden Zahl

li:nth-child(3n+0) {-} // stylt jede dritte Zeile, beginnend bei 0

p::first-letter {-} // stylt den ersten Buchstaben pro p Element

::selection {-} // stylt alles, sofern es markiert wurde

## Alles rund um die Schrift

@import url("https://fonts.googleapis.com/css?family=Ubuntu"); // importiert eine Schriftart

**font-family**:

'Ubuntu' // Schriftart deklarieren

sans-serif; // ohne Serifen

serif; // mit Serifen

monospace; // Diese Schriftart

**font-size**:

16px; // standard Textschriftgrösse

1em; // standard Textschriftgrösse

**font-style**:

italic; // kursiv

oblique; // 8-12° verzerrt

normal;

**font-weight**: // deklariert die Dicke der Schrift

light; // dünn

normal;

bold; // dick

**font-variant**:

small-caps; // Das hier (Kapitälchen)

normal;

**text-align**:

center; // zentriert

left; // linksbündig

right; // rechtsbündig

justify; // Blocksatz

**text-decoration**:

none;

underline; // unterstrichen

overline; // überstrichen lol

line-through; // durchgestrichen

STYLING: dashed; dotted; double; wavy; solid;

Beispiel: **underline overline wavy #ff0000;**

**text-shadow**:

horizontal (+ rechts / - links) vertikal (+ unten / - oben) blurr Farbe;

-10px 5px 20px #000000;

**text-transform**:

capitalize; // 1. Buchstabe von jedem Wort gross

lowercase; // alles klein

uppercase; // alles gross

none;

**line-height**: // dient zur vertikalen Zentrierung des Textes

16px; // setzt die Zeilenhöhe auf 16px

1em; // setzt die Höhe auf 1em

**color**: // verändert die Textfarbe

**background-color**: // verändert die Hintergrundfarbe

## Positionierung

**position**:

static; // defaultwert

relative; // Positionierung relativ zu ihrer normalen Position

(mit top: 50px; ist das Element 50px weiter unten als es normalerweise wäre)

absolute; // orientiert sich an dem Viewport

(mit top: 50px; ist das Element 50px von ganz oben)

fixed; // das Element bleibt beim Scrollen wo es ist

sticky; // bleibt oben kleben, sobald gescrollt wurde

**float**: // damit kann man Elemente nebeneinander darstellen lassen

left; // Ausrichtung links

right; // Ausrichtung rechts

none;

## Rahmen

**border-radius**: 10px; // rundet die Ecken ab

**border-color**: #ff0000; // färbt den Rahmen

**border-width**: 2px; // deklariert die dicke vom Rahmen

**border-style**:

STYLING: dotted; dashed; solid; double; groove; ridge; inset; hidden;

abgekürzt: **border: 1px solid #000000;**

## Unsortiert

**z-index**: // höchste Nummer im Vordergrund, kleinste im Hintergrund

**cursor**: all-scroll; ew-resize; grab; grabbing; move; ns-resize; none; not-allowed; pointer; progress; text; url; & noch ein paar weitere...

**::selection {**

color: white;

background: black;

**}** // wenn etwas selektiert wird

# Basic JavaScript

## Coole Sachen

// Beim draufklicken wirst du eine Seite zurückgeleitet.

<a onclick="history.back(-1)">Eine Seite zurück</a>

// Weiterleiten auf eine bestimmte Website

window.location = "https://github.com/thebauzz"