

CSS – Grundlagen I



Übersicht

- Einleitung
 - CSS – Wozu und Warum?
 - Einbinden von CSS (extern vs intern vs inline)
- Selektoren
 - Aufbau von Stilen
 - Einfache Selektoren
 - Pseudo-class Selektoren
 - Pseudo-elements Selektoren
 - Attribut Selektoren
- Styling
 - Farben
 - Schrift
- CSS Box Modell
 - Margin
 - Padding
 - Border
 - Content

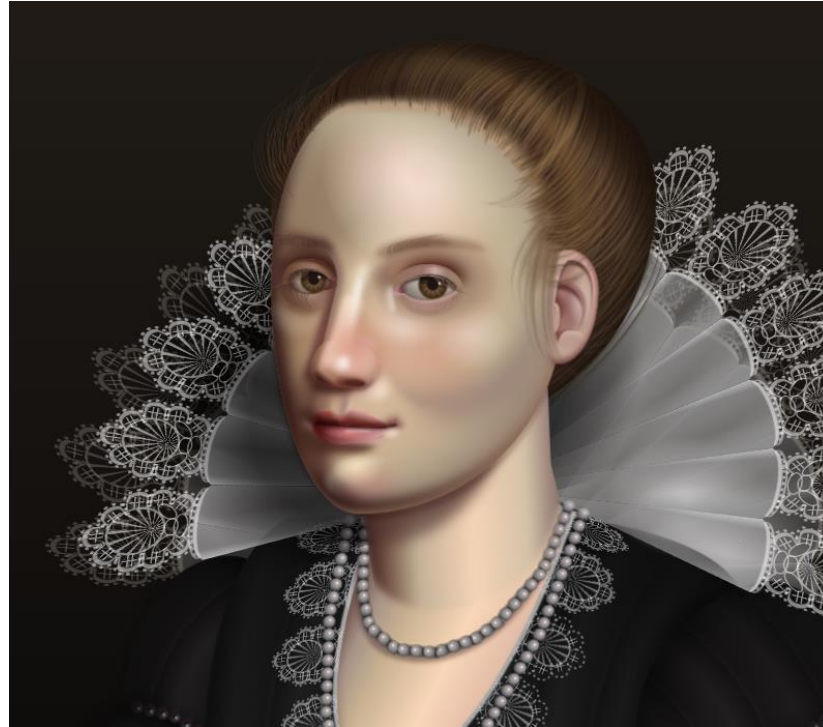


CSS Wozu und Warum?

- CSS steht für **Cascading Style Sheet**. Möchte man vom Standard-Aussehen einer Webseite im Browser abweichen, so kann man Stile für die genutzten Elemente definieren
- Ein Stil kann jeweils für ein oder mehrere Elemente gelten, je nach Definition.
- Es können Farben, Größen, Rahmenlinien, Schatten, Positionen, Hintergrundbilder und vieles weitere definiert werden.
- CSS Stile können entweder als Datei in die Webseite eingebunden werden, oder im HTML direkt definiert werden, letzteres sollte gut überlegt sein!



CSS Art



<https://css-art.com/pure-css-lace/>

Einbinden von CSS (extern)

- Mit einem externen Stylesheet kann mit einer Änderung an einer Stelle, das Aussehen mehrerer Seiten geändert werden
- Jede HTML-Seite muss eine Verweis auf die externe Stylesheet-Datei innerhalb des <link>-Elements im <head>-Bereich enthalten.

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

Einbinden von CSS (internal)

- Bei internem Styling wird das Aussehen einer speziellen Seite definiert. CSS Code wird innerhalb der HTML Datei geschrieben
- Das interne Stylesheet wird innerhalb des <style>-Elements im <head>-oder im <body> Bereich definiert.

```
<head>
<style>
body {
    background-color: linen;
}

h1 {
    color: maroon;
    margin-left: 40px;
}
</style>
</head>
```

Einbinden von CSS (inline)

- Ein Inline-Style kann verwendet werden, um einen einzigartigen Stil für ein einzelnes Element anzuwenden.
- Um Inline-Styles zu verwenden, fügen Sie das style-Attribut zum entsprechenden Element hinzu. Das style-Attribut kann jede CSS-Eigenschaft enthalten.

```
<h1 style="color:blue;text-align:center;">This is a heading</h1>  
<p style="color:red;">This is a paragraph.</p>
```

- Ein Inline-Style verliert viele der Vorteile eines Stylesheets (indem er Inhalt und Präsentation miteinander vermischt). Verwenden Sie diese Methode sparsam!

CSS - Selektoren



Aufbau von Stilen

- CSS Stil Definitionen bestehen aus sogenannten **Selektoren** und dazu gehörenden Stilen.
- Der **Selektor** definiert, für welches Element, bzw. für welche Elemente ein bestimmter Stil gilt.
- Auf den Selektor folgen **geschwungene Klammern**, im inneren von diesen sind die einzelnen Stile als **Key-Value Paare** definiert.
- Diese werden mit : (Doppelpunkt) getrennt und am Ende der Zeile mit ; (Semikolon) abgeschlossen.

```
h1 {  
    color: red;  
    text-decoration: underline;  
}  
  
h2 {  
    color: blue;  
}
```

Einfache Selektoren: Tag-Selektor

- Die einfachste Art des Selektors ist der **Tag-Selektor**
- Besteht aus Tag-Namen des Elements, welches man stylen möchte.
- Der Stil betrifft alle Elemente mit dem gegebenen Tag-Namen.

```
<h1>CSS Demo</h1>
<h2>Grundlagen</h2>
<h2>Selektoren</h2>
```

```
1 h1 {
2     color: red;
3     text-decoration: underline;
4 }
5
6 h2 {
7     color: blue;
8 }
```

Einfache Selektoren: Class-Selektor

- Möchte man einen Stil auf mehrere Elemente anwenden, aber nicht auf alle eines bestimmten Tag-Typs, so kann man Klassenselektoren verwenden.
- Dazu beginnt man den Selektor mit . (Punkt) und schreibt dahinter den Wert eines class-Attributs.
- HTML Elemente können mehrere Klassennamen haben, sie können im class-Attribut mit Leerzeichen getrennt angegeben werden..

```
1 <p class="lorem">Hallo</p>
2 <p class="lorem ipsum">Welt</p>
3 <p>Willkommen!</p>
```

```
1 .lorem {
2     font-weight: bold;
3 }
4
5 .ipsum {
6     color: red;
7 }
```

Einfache Selektoren: ID-Selektor

- Mit ID Selektoren greift man auf ein ganz bestimmtes Element zu,
- Selektor beginnt mit # gefolgt von id-Attribut eines Elements.
- Da ID-Attribute auf der gesamten Webseite eindeutig sein müssen (es darf jedes nur einmal geben), gilt dieser Stil nur für genau ein Element.

```
<h2 id="demo">Grundlagen</h2>
```

```
<h2>Selektoren</h2>
```

```
1 #demo {  
2     color: red;  
3 }
```

Pseudo-class Selektoren

- Pseudo Klassen beschreiben Zustände und Positionen von Elementen
Beispiel: hovers der Nutzer gerade mit der Maus über ein Element? Die n-te Stelle in einer Liste von Elementen,...
- Man schreibt sie hinter einen : (Doppelpunkt). Sie können auch Parameter haben, welche man in () übergibt.
- Meist kombiniert man Pseudo-Klassen mit einem anderen Selektor, indem man sie zusammen schreibt.
- Eine vollständige Liste der verfügbaren Pseudo-Klassen findet man [hier](#).

<pre> 1 2 Lorem 3 Ipsum 4 Dolor 5 </pre>	<pre> 1 :nth-child(2) { 2 font-weight: bold; 3 } 4 5 li:hover { 6 color: red; 7 } </pre>
--	--

Pseudo-element Selektoren

- Pseudo Elemente sind Elemente die in HTML nicht als solche existieren, aber mit CSS an bestimmten Stellen angehängt werden können.
- Man schreibt sie mit :: (Zwei Doppelpunkte) und ihrem entsprechenden Namen.
- Sie werden beispielsweise verwendet, um eigene Arten von Aufzählungspunkten zu schaffen, oder um vor jedem Link im Text ein Icon anzuzeigen.
- Pseudo-Element Selektoren werden für gewöhnlich auch mit anderen Selektoren kombiniert.
- [Link](#) zur vollständigen Liste

```
1 <p>1.50</p>
2 <p>3.99</p>
3 <p>7.89</p>
```

```
1 p::after {
2     content: ' €';
3 }
```

[Attribut] Selektor

- Der [Attribut]-Selektor wird verwendet, um Elemente mit einem angegebenen Attribut auszuwählen.
- Das folgende Beispiel wählt alle <input>-Elemente mit dem Attribut placeholder=„Email“ aus:

```
<input type="email" placeholder="Email">
```

```
input[placeholder="Email"] {  
    background-color: yellow;  
}
```

CSS Selektoren Übung

- In deinen Übungsunterlagen unter CSS/Selectors/ findest du 3 Dateien
- Schreibe CSS um einen Stil wie in der Abbildung rechts zu erzielen
- Nutze sowohl internes als auch externes Einbinden des CSS Codes

CSS Selectors

Applying CSS to Different Parts of HTML

1. The element selector targets elements based on their HTML tag name.
2. Class selectors target elements based on the value of the class attribute.
3. ID selectors target elements based on the value of the id attribute.
4. Attribute selectors target elements based on their attributes and values.
5. The universal selector targets all elements.

CSS – Styling



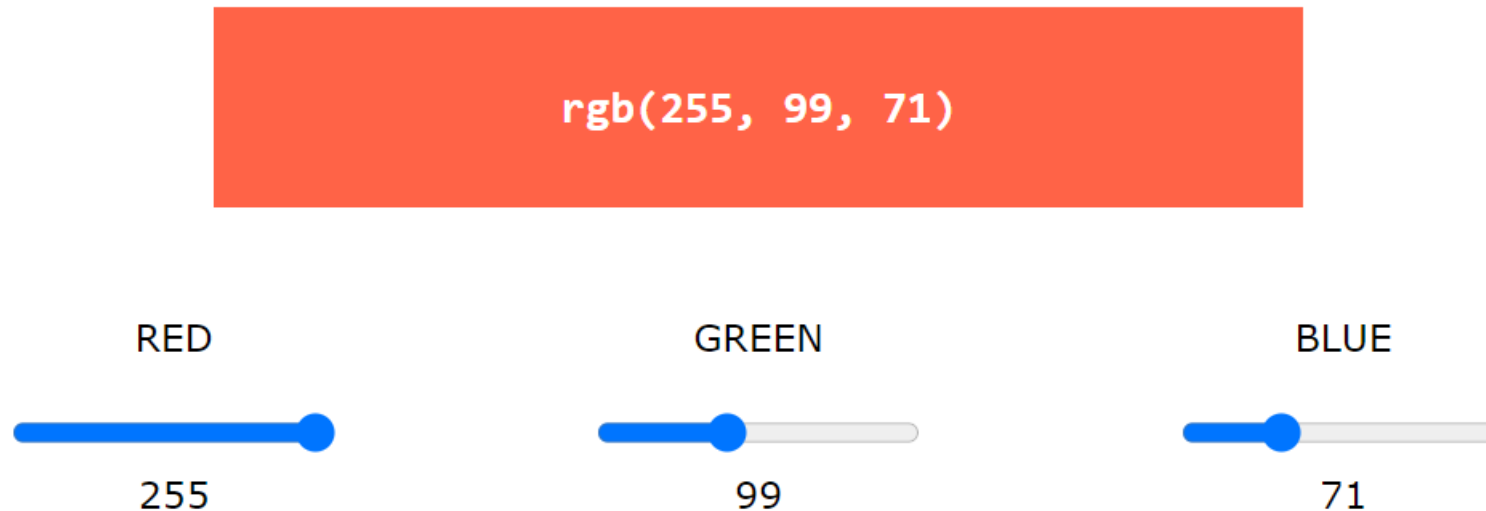
Farben

- Farben werden in CSS mithilfe vordefinierter Farbnamen oder RGB-, HEX-, HSL-Werte spezifiziert.
- Eine Liste aller Farbnamen findest du [hier](#)
- RGB:
Jeder Parameter (Rot, Grün und Blau) definiert die Intensität der Farbe zwischen 0 und 255
- HEX:
Ein hexadezimaler Wert hat die Form: #rrggbb.
Dabei sind rr (rot), gg (grün) und bb (blau) hexadezimale Werte zwischen 00 und ff (entspricht dezimal 0-255)
- HSL:
Unter Verwendung von Farben im HSL-Format erfolgt die Angabe in der folgenden Form:
hsl(Farbwinkel, Sättigung, Helligkeit)



RGB – (ROT, GRÜN, BLAU)

- Jeder Parameter (Rot, Grün und Blau) definiert die Intensität der Farbe zwischen 0 und 255
- https://www.w3schools.com/css/css_colors_rgb.asp



Schrift-, Hintergrund- und Rahmen-Farbe

- Folgende CSS Properties werden verwendet um Schrift-, Hintergrund- und Rahmen-Farbe zu definieren
- Schrift: color
- Hintergrund: background-color
- Rahmen-Farbe: border-color
- Rahmen: border: 10px solid green

```
<style>
  p {
    background-color: blue;
    color: red;
    border-color: green;
    border: 10px solid green;
  }
</style>
```

Lorem ipsum dolor sit amet consectetur adipisicing elit. Reiciendis quos officia laudantium? Nam molestiae repudiandae iure, eos amet blanditiis aut consequatur, ratione distinctio, error repellendus laborum! Ullam veritatis cum ab.

Schriftgröße

- Schriftgröße wird über das Property „**font-size**“ definiert
- Die Schriftgröße kann über 4 verschiedenen Einheiten definiert werden
- Statische Schriftgröße
 - Pixel (px): 0,25 mm
 - Point (pt): 0,35 mm
- Dynamische Schriftgrößen
 - 1 em: 1 x Schriftgröße des Parent Elements
 - 1 rem: 1 x Schriftgröße des root Elements

1 Pixel = 0.25 mm

1 Point = 0.35 mm

1em = 100% des parent elements

1rem = 100% des root element (html)

Schriftstärke

- Die Schriftstärke „**font-weight**“ kann durch Angabe einer Zahl oder durch Keywords definiert werden
- Zahl: 100, 200, ... , 800, 900
- Keywords:
 - normal = 400
 - bold = 700 oder 800 abhängig von font-family
 - lighter = font-weight des parent-elements – 100
 - bolder = font-weight des parent-elements + 100

normal

bold

font-weight:400

100 lighter than parent element

100 bolder than parent element

Schriftart

- In CSS verwendet man die Eigenschaft "**font-family**", um die Schriftart eines Textes anzugeben
- Wenn der Schriftartenname aus mehreren Wörtern besteht, muss er in Anführungszeichen stehen, wie zum Beispiel: "Times New Roman",
- Es sollten mehrere Schriftnamen als "Fallback"-System verwendet werden
 - font-family: "Times New Roman", serif;
- Beginne mit der gewünschten Schriftart und ende mit einer generischen Schriftart
 - Serif
 - Sans-serif
 - Monospace
 - Kursiv
 - Fantasy

Serif

Sans Serif

Cursive

Monospace

Fantasy

Serif vs Sans-serif

- Serifen-Schriftarten haben an den Enden jeder Buchstabe eine kleine Strichführung. Sie erzeugen ein Gefühl von Formalität und Eleganz.
- Serifenlose Schriftarten haben klare Linien (keine kleinen Striche). Sie erzeugen einen modernen und minimalistischen Look.
- Auf Computerbildschirmen gelten serifenlose Schriftarten als leichter lesbar als serifenbetonte Schriftarten.

F

Sans-serif

F

Serif

F

Serif
(red serifs)

Google Fonts

- Wenn du keine der Standard-Schriftarten in HTML verwenden möchtest, kannst du Google Fonts nutzen.
- Google Fonts sind kostenlos und bieten mehr als 1000 Schriftarten zur Auswahl.



Google Fonts

<https://fonts.google.com/>

Text Ausrichtung

- Die Eigenschaft "**text-align**" wird verwendet, um die horizontale Ausrichtung eines Textes festzulegen.
- Ein Text kann folgendermaßen ausgerichtet werden.

- Linksbündig: left
- Rechtsbündig: right
- Zentriert: center
- Blocksatz: justify

This is a left-aligned paragraph.

This is a center-aligned paragraph.

This is a left-to-right direction paragraph.

This is a right-aligned paragraph.

.This is a right-to-left direction paragraph

.noitcerid txet tfel-ot-thgir si sihT

- Die Eigenschaften "**direction**" und "**unicode-bidi**" können verwendet werden, um die Textrichtung eines Elements zu ändern.

Zusätzliche Text Formatierung

- Die Eigenschaft "**text-decoration-line**" wird verwendet, um eine dekorative Linie zu einem Text hinzuzufügen
- Die Eigenschaft "**text-transform**" kann verwendet werden, um den Text in Großbuchstaben, Kleinbuchstaben oder den ersten Buchstaben jedes Wortes in Großbuchstaben umzuwandeln (kapitalisieren)
- Die Eigenschaft "**text-shadow**" fügt Schatten zum Text hinzu
- Die Eigenschaft "**line-height**" wird verwendet, um den Abstand zwischen Zeilen festzulegen.

Overline text decoration

~~Line-through text decoration~~

Underline text decoration

Overline and underline text decoration.

Note: It is not recommended to underline text that is not a link, as this often confuses the reader.

THIS TEXT IS TRANSFORMED TO UPPERCASE.

this text is transformed to lowercase.

This Text Is Capitalized.

This text has a shadow

This is a paragraph with a smaller line-height.
This is a paragraph with a smaller line-height.

This is a paragraph with a bigger line-height.

This is a paragraph with a bigger line-height.

Hintergrund

- Hintergrund-Eigenschaften in CSS ermöglichen die Steuerung des Erscheinungsbilds des Hintergrunds eines Elements.
- **Background-Color:** Legt eine einheitliche Hintergrundfarbe für ein Element fest.
Beispiel: background-color: #3498db;
- **Background-Image:** Setzt ein Bild als Hintergrund für ein Element.
Beispiel: background-image: url('hintergrund.jpg');
- **Background-Size:** Definiert die Größe des Hintergrundbildes.
Beispiel: background-size: cover;
- **Background-Repeat:** Bestimmt, wie das Hintergrundbild wiederholt wird.
Beispiel: background-repeat: no-repeat;
- **Background-Position:** Legt die Startposition des Hintergrundbildes fest.
Beispiel: background-position: center;
- **Background-Attachment:** Steuert, ob das Hintergrundbild mit dem Inhalt scrollt
Beispiel: background-attachment: fixed;
- Mehrere Hintergründe: CSS erlaubt das Anwenden mehrerer Hintergründe auf ein Element.

```
background-image: url('bg1.jpg'), url('bg2.jpg');  
background-size: cover, auto;  
background-repeat: no-repeat, repeat-x;
```

CSS Übung Styling

- Übungsunterlagen: CSS/Styling
- Schreibe internen CSS Code um Farben und Schriftarten wie im Bild rechts zu erzielen
- Eine Schritt für Schritt Anleitung findest du im index.html File

Important CSS Properties

Color

Font Size

Font Weight

Font Family

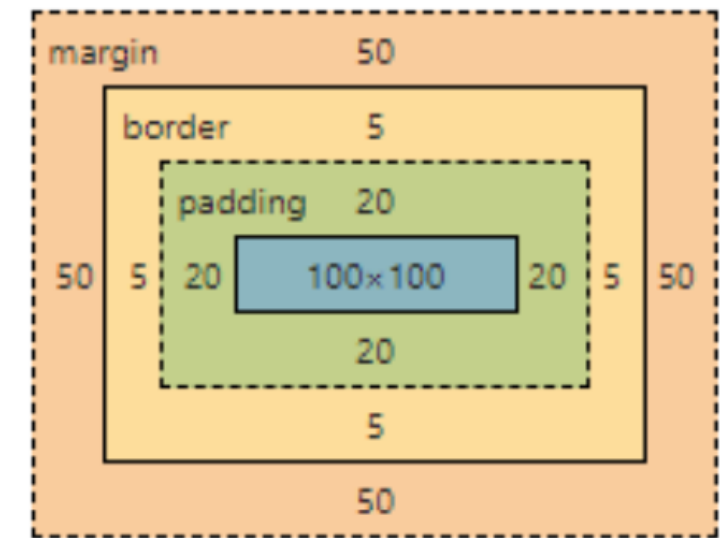
Text Align

CSS – Box Modell



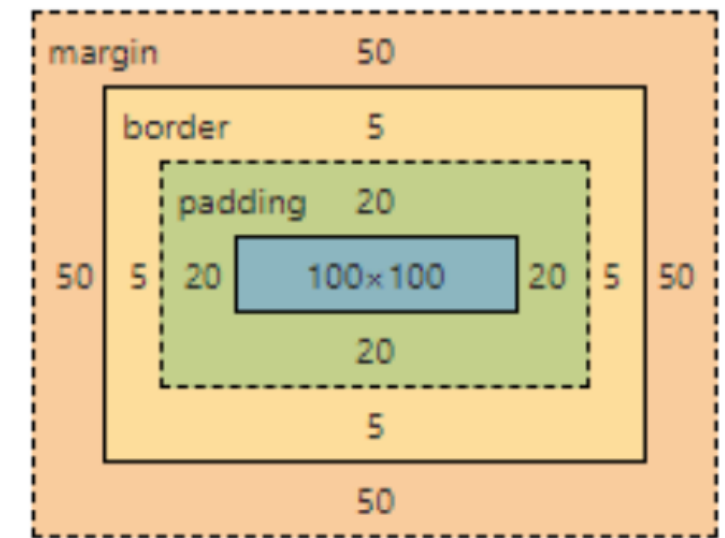
CSS Box Modell

- Das Box Model beschreibt, auf welche Weise ein Element Platz einnimmt, wo und wie es Platz reserviert. Von innen nach außen hat jedes Element folgende Bereiche:
- **content**: Bereich in dem Inhalt platziert wird
- **padding**: Abstand zum Rand des Elements
- **border**: Rand des Elements
- **margin**: Abstand zu anderen Elementen



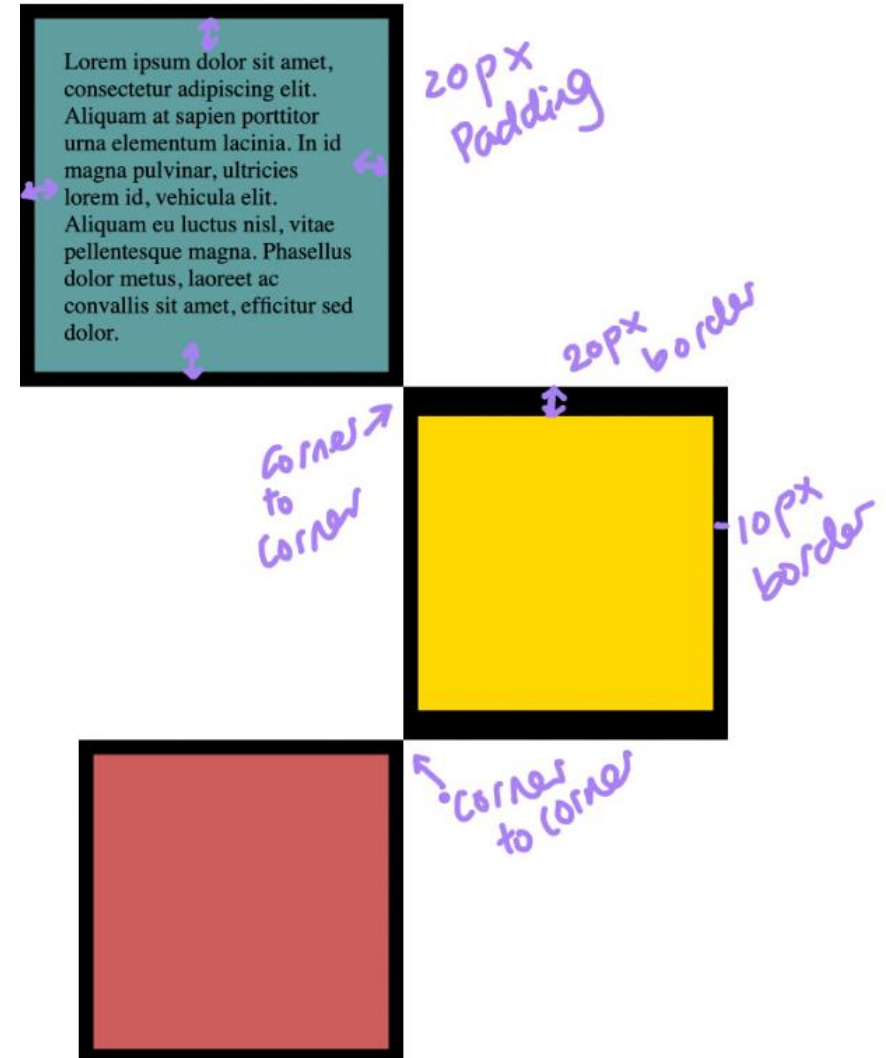
CSS Box Modell Einheiten

- **Pixel (px):** Die Einheit "px" (Pixel) ist eine feste Einheit und gibt die Anzahl der Bildpunkte auf dem Bildschirm an. Zum Beispiel: 10px, 20px, usw.
- **Em:** Die Einheit "em" ist relativ zur aktuellen Schriftgröße des Elements. Zum Beispiel: 1em, 1.5em, usw.
- **Rem:** Die Einheit "rem" ist relativ zur Schriftgröße des Wurzelements (normalerweise des <html>-Elements). Zum Beispiel: 1rem, 1.5rem, usw.
- **Prozent (%):** Prozentuale Werte beziehen sich auf einen Prozentsatz der Breite des übergeordneten Elements. Zum Beispiel: 10%, 50%, usw.
- **Viewport Width (vw) und Viewport Height (vh):** Diese Einheiten sind relativ zur Breite bzw. Höhe des sichtbaren Bildschirms. Zum Beispiel: 10vw, 5vh, usw.



CSS Übung Box Modell

- Übungsunterlagen: CSS/Box Model/
- Verwende dein Wissen über das CSS Box Modell und Farben, um das Bild rechts zu reproduzieren
- Eine TODO Liste und Hinweise findest du im index.html File als Kommentar



CSS 1 Abschlussübung Motivationsposter

- Erstelle ein Motivationsposter ähnlich zu folgendem. Nutze dafür dein Wissen über Selektoren, Styling und das Box Modell



Viel Erfolg beim Entwickeln!

