

Styling with CSS

Pablo Klaschka im Rahmen des Frontend Workshops

Kontakt: contact@pabloklaschka.de



CSS? Was ist das denn jetzt schon wieder?!

- CSS \equiv „Cascading Style Sheets“
- Deklaratives Styling von Websites (aber auch anderen Dingen)
- Zwei Möglichkeiten der Nutzung:
 - .css-Dateien werden in HTML eingebunden
 - CSS steht in `<style>[...]</style>` Element



Ein kleines Beispiel

```
h1 {  
  font-size: 32px;  
  color: #ff0000; /* Red headlines */  
  margin-top: 32px;  
}
```

Selektiere alle <h1> Elemente

Mache die Schrift 32px groß

Mache die Schrift rot

Und einen Abstand nach oben von 32px



Selektoren

Name	HTML	CSS	Beschreibung
Typ	<code><h1>Hallo Welt</h1></code>	<code>h1 { }</code>	
Klasse	<code> Hallo Welt </code>	<code>.test { }</code>	
ID	<code></code>	<code>#logo { }</code>	Nur ein Element pro Seite darf die gleiche ID besitzen!
Attribut	<code> Hallo Welt </code>	<code>[lorem] { }</code>	Selektiert alle Elemente, die das @lorem-Attribut besitzen.
"Alles"	n/a	*	Selektiert sämtliche Elemente



Selektoren verbinden

Name	HTML	CSS	Beschreibung
Child	<code><a><c></code> Ich werde gestylt <code></c></code> <code><c>Ich nicht</c></code>	<code>a c { }</code>	Hier wird <code><c></code> selektiert, wenn es ein direktes oder indirektes Kind von <code><a></code> ist.
Direct Child	<code><a></code> Ich werde gestylt <code></code> <code><a><c></code> Ich nicht <code></c></code>	<code>a > b { }</code>	<code></code> wird selektiert, wenn es ein direktes Kindelement von <code><a></code> ist.



Und wenn mehrere Selektoren zutreffen?

- Spezifität
- sehr vereinfacht ausgedrückt: Der „längere“ (bzw. spezifischere) Selektor gewinnt
- Ids zählen mehr als Attribute
- Attribute zählen mehr als Klassen
- Klassen zählen mehr als Typen
- Im Zweifelsfall die Zuweisung „weiter unten“ im Code!
- Mehr Infos unter <https://developer.mozilla.org/de/docs/Web/CSS/Spezifit%C3%A4t>

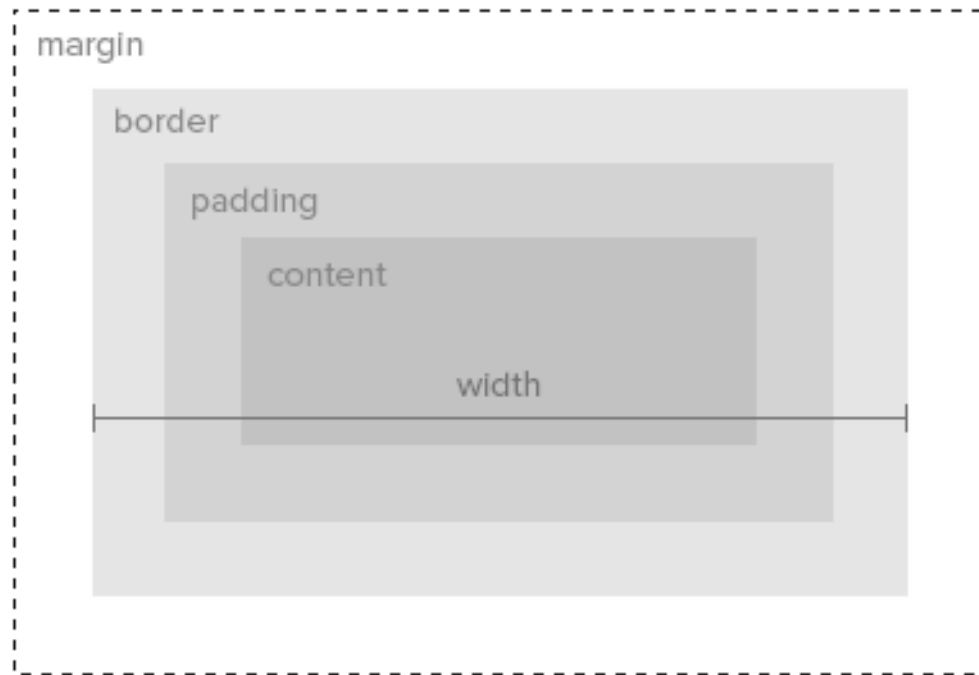


Kaskadieren

- Beispiel:
 - HTML: `<div><p>Test</p></div>`
 - CSS: `div { font-size: 32px; }`
- Ergebnis: div-font-size kaskadiert => gilt auch für `<p>`



Everything's a box? Das Box-Model



Grafik von <https://www.abeautifulsite.net/box-sizing-border-box-explained>

- Block-Elemente halten sich an das Box-Model
- Elemente haben eine (je feste oder variable) Breite und Höhe
- Um die Kindelemente gibt es das “padding”, den Innenabstand
- Darum den Rahmen (border)
- Separation von umgebenden Elementen durch Außenabstand (margin)

Alles Boxen?

Nein, weil es auch Inline-Elemente gibt 🤪

- Inline-Elemente fließen im Textfluss “mit”
- Sie verhalten sich wie Wörter im Text
- Kein Padding oder Margin (außer inline-block)
- Elemente, die standardmäßig „inline“ sind:
 - `<code>`
 - ``
 - ``
 - ``



Einheiten in CSS

- px: Ein “Pixel”
- em: Eine “font-size”-Einheit
- %: % des Parents (width: 100%; => 100% der Parent-Element-Breite)
- rem: Eine “font-size”-Einheit basierend auf dem Root-Element (<html>)
- Farben: Hexadezimal, z.B. #ff0000 für Rot
u.v.m. (bewusst ausgelassen!)



Wichtige CSS-Eigenschaften

Eigenschaft	Beispielwerte	Beschreibung
color	red, #ff0000, blue	Schriftfarbe
background	#ff0000, url(bg-image.png)	Hintergrundfarbe oder -bild
font-size	16px, 3rem, 20 %	Schriftgröße
width	" "	Breite des Block-Elements
height	" "	Höhe des Block-Elements
margin margin-left margin-right margin-top margin-bottom	" "	Außenabstand des Block-Elements



Wichtige CSS-Eigenschaften

Eigenschaft	Beispielwerte	Beschreibung
padding padding-left padding-right padding-top padding-bottom	" "	Innenabstand des Elements: Abstand der Kindelemente des Elements von den von der Box (width, height) aufgezogenen Dimensionen
display	block, inline, inline-block	Lässt Umwandlung von Block- in Inlineelemente (und andersherum) zu.
text-align	left, right, center, justified	Textausrichtung
line-height	24 px, 1.5 em, 150 %	Zeilenhöhe



Live-Coding

Basierend auf bisherigen Ergebnissen.
Checkpoint: day1-02-html-and-css



Ausblick

- “display: flex” und “display: grid”: Das kontrolliertere Inline für Block-Elemente
- Media-Queries: Styles für kleine (oder große?) Bildschirme
- Pseudo-Attribute (:not, :checked, :hover u.Ä.): Andere Farbe wenn der Cursor drüber schwebt

