

Ablauf

1	13. Oktober	Einführung + HTML
2	20. Oktober	CSS Selektoren
3	27. Oktober	CSS Grid
4	3. November	Geräte, Formate, Pixel-Bilder
5	10. November	Vektor-Bilder
6	17. November	Einheiten + Animation
7	24. November	Interaktion
8	1. Dezember	Typografie
9	8. Dezember	Farbe
10	15. Dezember	Dokumentation
11	22. Dezember	Briefing
12	5. Januar	Semesterprojekt
13	12. Januar	Semesterprojekt
14	19. Januar	Semesterprojekt
15	26. Januar	Präsentation

Recap

- 1 Werkzeuge: Email, Mattermost, Seafile, VS Code, Live Server
- 2 Warum Webentwicklung lernen?
- 3 Was sind Webseiten?
- 4 HTML
- 5 Übungen von Woche 1

Javascinot

Cascading StV/E Sheets

Syntax

Der Selektor spricht ein oder mehrer HTML-Elemente an

Selektor { Eigenschaft: Wert;

Die Eigenschaft ist ein bestimmer gestalterischer Aspekt, zum Beispiel Schriftgröße oder Farbe Der Wert setzt einen gestalterischen Aspekt auf einen bestimmten Wert, zum Beispiel 20px.

Syntax

Wir sprechen alle -Elemente in unserem HTML-Dokument an.



Wir setzen die Schriftfarbe auf Rot.

```
h1 {
  color: red;
  font-weight: 900;
h2 {
  color: pink;
  font-style: italic;
  border: 3px solid green;
```

CSS-Datei einbinden

```
index.html

...
<head>
    <title>Meine tolle Seite</title>
    <link rel="stylesheet" href="style.css"/>
</head>
...
```

Your friendly neighbourhood selectors

Der Element-Selector

```
Inhalt hier
```

```
p {
  color: red;
}
```

Der Class-Selector

```
   Inhalt hier
```

```
.product-details {
  color: red;
}
```

Der ID-Selector

```
   Inhalt hier
```

```
#article-intro {
  color: red;
}
```

Les fancy selectors 1/2

Die hover- und active-Selektoren

```
<button>
  Jetzt Kaufen!
</button>
```

```
button {
 color: red;
button:hover {
  color: green;
button:active {
  background: red;
```

Les fancy selectors 2/2

Der nth-of-type-Selector

```
>
 Hier ist mein erster Absatz
>
 Und mein zweiter Absatz
>
 Absatz Nummer 3
>
 Und Nummer 4
```

```
p:nth-of-type(3){
   color: red;
}
p:nth-of-type(odd){
   color: yellow;
}
p:nth-of-type(2n){
   color: green;
}
```

Wir können Selektoren kombinieren, um Kinder von bestimmten Elementen anzusprechen....

```
>
 Dieser Absatz
 steht alleine
<article>
 >
   Dieser Absatz steht
   in einem Artikel
 </article>
```

```
p {
  color: green;
}

article p {
  color: red;
}
```

...oder Elemente, auf die mehrere Eigenschaften zutreffen (UND-Relation)

```
<div class="intro">
</div>
```

```
.intro {
  color: green;
p.intro {
  color: red;
p.intro#special {
  color: blue;
```

Komplexe Selektoren lesen wir von rechts nach links

...innerhalb von <article>-Elementen mit einem class-Attribut mit dem Wert project ...innerhalb von Elementen mit einem class-Attribut mit dem Wert intro Alle <a>-Elemente, über denen gerade der Mauszeiger schwebt

```
article.project .intro a:hover {
  color: red;
```

Zusammenfassung

```
<article>

      Inhalt hier

</article>
```



Aufgabe 1

- 1 Lade die Vorlage Woche-2/Aufgabe-1 herunter.
- 2 Öffne die index.html und erstelle 30 <div>s mit beliebigem Textinhalt. Gebe den <div>s per CSS eine Breite und Höhe, Hintergrundfarbe, Schriftgröße und Abstand.
- 2 Gebe fünf <div>s ein beliebiges class-Attribut. Dann gib ihnen mithilfe des Class-Selekors eine andere Hintergrundfarbe.
- 4 Vergib über den Pseudoselektor :nth-of-type(...) weitere Eigenschaften wie border-radius oder transform an jedes dritte Element. Führe das so fort, dass mit möglichst wenig CSS eine möglichst unterschiedliche Gruppe an Objekten entsteht.
- 5 Lade das Ergebnis unter Abgaben/Woche-2/Aufgabe-1/Dein_Name hoch.

CSS: Die Cascade

```
<article>
 Dieser Absatz steht
   in einem Artikel
 </article>
>
 Dieser Absatz
 steht alleine
```

```
article {
  color: red;
```

2/2 Styling Order

```
<article>

     Welche Farbe habe ich?

</article>
```

```
article {
  color: green;
p
  color: pink;
article p {
  color: purple;
.intro {
  color: red;
.intro {
  color: orange;
#test {
  color: brown;
#test {
  color: blue !important;
```

Styling inherited from nearest The definitive guide to CSS styling order parent element Includes CSS stylings for SVG Child styling (if exist) has higher priority even though inherited parent styling contains important keyword Styling methods Inherited styles has the lowest Inheritance priority among styling methods Ordering, selectors or specificity and important keyword does not Inline attributes (for SVG) apply to SVG inline attributes Specificity has higher priority than Style sheet ordering, with tags, classes and ID in ascending priority. CSS codes to the right or bottom Embedded / External styles has higher priority and will be #myID { color: green; } applied. Specificity .MyClass { color: blue; } p { color: red; } p { color: red; color: blue; } Tags (eg: p, div) Element will be styled with Blue will be applied green because ID specificity Ordering has the highest priority, superceding ordering rules. color: red; Classes (eg: .MyClass) color: blue; Within specificity, ordering rules still applies. Ordering Blue will be applied #myID { color: red; } #myID { color: blue; } p { color: red; } ID: (eg: #myID) p { color: blue; } Blue will be applied. Blue will be applied Ordering Inline styles has higher priority than For each style, ordering rules style sheets, and within inline continue to apply, from left to right styles, ordering rules applies. and top to bottom. <style> color: blue"> Inline styles #myID { color: red } </stvle> Blue will be applied <style> Important keyword #myID { color: blue } </style> Important keyword in inline styles Style sheet

Inline styles

has higher priority than the same

keyword in style sheets.

Blue will be applied

Aufgabe 2

- 1 Lade die Vorlage Woche-2/Aufgabe-2 von Seafile herunter.
- 2 Nimm dein HTML des Guerrilla-Girls-Plakats von Woche 1 (oder die Vorlage aus dem Seafile-Ordner) und verwende CSS, um deine Version genau wie das Original aussehen zu lassen. Modifiziere dein HTML falls nötig. Recherchiere die nötigen CSS-Eigenschaften.

Bonus: Binde das externe Stylesheet von der URL https://use.typekit.net/vxf0pzp.css ein. Wenn du das richtig gemacht hast, kannst du die Schrift Futura in deiner eigenen CSS-Datei als futura-pt und futura-pt-condensed verwenden.

3 Lade den fertigen Order unter Abgaben/ Woche-2/Dein_Name hoch.

Literatur

Zum Thema CSS

Geoff Graham (2018): CSS Basics: The Second "S" in CSS. css-tricks.com/css-basics-second-s-css

Thomas Yip (2018): *The "C" in CSS: The Cascade* css-tricks.com/the-c-in-css-the-cascade

Chris Coyier (2010): *How nth-child Works*. css-tricks.com/how-nth-child-works

Für Historiker:innen

Karl Gerstner (1964): *Designing Programmes* (Facsimile Edition 2019), Lars Müller.