

CSS

Programmieren

D. Langhanki



© D. Langhanki

Bedeutung

Cascading StyleSheets

Unterstützt Gestaltung von Website(s)

Ziel: Trennung von Struktur und Gestaltung

Struktur → HTML, Gestaltung → CSS.

Vergleichbar z. B. mit Formatvorlagen in Word.

Nicht:

jedes Element im HTML-Dok. einzeln gestalten,

Sondern:

über CSS Formatierung aller Elemente einer bestimmten Gruppe festlegen

© D. Langhanki

2

Die wichtigsten Ziele von CSS

Darstellung von demselben Inhalt für **verschiedene** Ausgabemedien: PC, Handy, Drucker...

Gleichartige Gestaltung von **mehreren** HTML-Seiten

Trennung von Struktur und Gestaltung, damit Unterstützung der Barrierefreiheit

© D. Langhanki

3

Syntax

Ein Titel, mit CSS gestaltet

```

1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
2 "http://www.w3.org/TR/html4/loose.dtd">
3 <html lang="en">
4 <head>
5 <meta http-equiv="content-type" content="text/html; charset=utf-8">
6 <title>CSS-Beispiel</title>

```

```

h1 {
  font-family: sans-serif;
  font-size: 20pt;
  color: #668833;
  border-bottom: 1px solid red;
}

```

23 </body>

24 </html>

Deklaration

Selektor | **Eigenschaft (Attribut)** | **Wert (Attributwert)**

h1 { color : #668833 ; }

© D. Langhanki

4

Einbindung von CSS: 3 Möglichkeiten

- lokal → CSS-Anweisungen stehen direkt im HTML-Tag
- zentral → CSS-Anweisungen stehen im head im HTML-Dokument
- extern → CSS-Anweisungen stehen in weiterem Dokument. Im head wird darauf referenziert.

© D. Langhans

5

Lokale Einbindung (oder „Inline“)

```
<body style="background-color: #111111;" ... </body>
```

```
<body bgcolor = "#111111"> ... </body>
```

HTML ohne CSS

© D. Langhans

6

Zentrale Einbindung (oder „style-tag“)

```
<head>
...
<style type="text/css">
  body {
    background-color: #111111;
  }
  h1 {
    color: #cc0000;
    font-size: 18px;
  }
</style>
...
</head>
<body> ... </body>
```

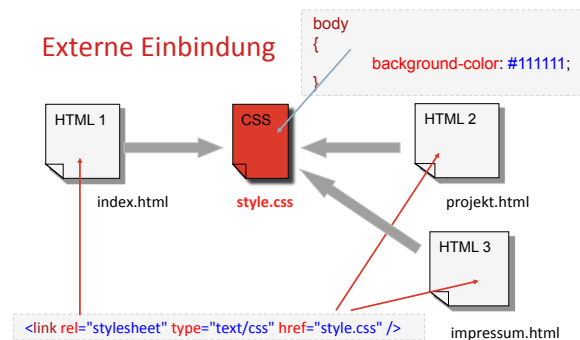
Semikolon:

- bei einem Attribut optional
- bei mehreren notwendig!

© D. Langhans

7

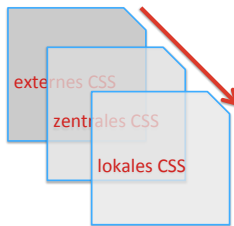
Externe Einbindung



© D. Langhans

8

Überschreiben



Ansonsten gilt erstmal:
die Reihenfolge entscheidet.
Der unten definierte Selektor
überschreibt den oberen.

```
h2{ color: #5566AA;}  
...  
h2{ color: #8811EE;}
```