

# CSS Grundlagen lernen - Anleitung für Anfänger

Querverweis:

[So werden Farben mittels](#)

[Hexadezimalcode angegeben](#)

CSS (Cascading Style Sheets) wird zum Formatieren von HTML verwendet und ist eine wichtige Grundlage für das optische Erscheinungsbild einer Webseite. Mit Hilfe von CSS Angaben in einem HTML Dokument, oder in einer gesonderten CSS- Datei hat man die Möglichkeit HTML zentral zu definieren, so dass Änderungen, die gemacht werden müssen, nur an einem Ort zu machen sind, und überall übernommen werden. CSS Formatierungen, die in einem HTML Dokument zutreffen sollen, werden im <head> gemacht.

Hier ein Beispiel, wie man alle Absätze einer Seite rot färbt:

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

**Erklärung:**

Selektor    Eigenschaft    Wert

p { color: red }

webbalance.de

Das was vor der geschweiften Klammer steht wird zu unserem Selektor. Das bedeutet, dass alle zutreffenden HTML Elemente entsprechend dem Klammerinhalt definiert werden. Zutreffende HTML Elemente sind hier alle Absätze (p-Tags). Der Inhalt der Klammer ist die CSS Deklaration, in diesem Fall werden allen p-Tags eine Farbe zugewiesen. Color ist unsere Eigenschaft, die einen Wert zugewiesen bekommt (Rot).

Daraus ergibt sich das Schema:

```
SELEKTOR { Eigenschaft: Wert }
```

Nun wird die CSS Angabe als eine solche definiert, sodass der Browser erkennen kann, dass es sich hierbei um CSS-Code handelt.

```
1 <style type="text/css">
2     <!--
3         p { color: red }
4     -->
5 </style>
```

...und in den <head> eingefügt. So könnte eine vollständige HTML-Datei mit CSS aussehen:

```
1 <html>
2 <head>
3     <style type="text/css">
4         <!--
5             p { color: red }
6         -->
7     </style>
8 </head>
9 <body>
10     <h1>So funktioniert's</h1>
11     <p>Erster Absatz</p>
12     <p>Zweiter Absatz</p>
13     <p>Dritter Absatz</p>
14 </body>
15 </html>
```

Hier sehen Sie das [Ergebnis](#)

Eine Besonderheit ist das direkte Formatieren mit Hilfe des `style`-Attributs (siehe zeile 12) :

```
1 <html>
2 <head>
3   <style type="text/css">
4     <!--
5       p { color: red }
6     -->
7   </style>
8 </head>
9 <body>
10   <h1>So funktioniert's</h1>
11   <p>Erster Absatz</p>
12   <p style="color:blue">Zweiter Absatz</p>
13   <p>Dritter Absatz</p>
14 </body>
15 </html>
```

Hier sehen Sie das [Ergebnis](#)

### Erklärung:

Wenn ein Element mit dem `style`-Attribut formatiert wird, wird diese, sehr direkte Methode priorisiert und überschreibt somit gleichbedeutende Eigenschaften die im Head, oder in einem [externen Stylesheet](#) definiert wurden.

## CSS Selektor Klasse

Eine weitere Anwendungsmöglichkeit von CSS ist das definieren von HTML Elementen über Ihre Klassenzugehörigkeit, dazu wird jedem gewünschten HTML Element einer Klasse zugeordnet, die dann möglichst Zentral (am Besten in einer [externen CSS Datei](#)) definiert wird. Hier ein Beispiel mit interner CSS Deklaration:

```
1 <html>
2 <head>
3   <style type="text/css">
4     <!--
5       .blau_und_dick { color: blue;
6                       font-weight:bolder
7                     }
8     -->
9   </style>
10 </head>
11 <body>
12   <h1> Klassen funktionieren so:</h1>
13   <p class="blau_und_dick">Erster Absatz</p>
14   <p>Zweiter Absatz</p>
15   <div class="blau_und_dick">div-container</div>
16 </body>
```

Hier sehen Sie das [Ergebnis](#)

### Erklärung:

In Zeile 13 und 15 wurden HTML Elemente der Klasse "blau\_und\_dick" zugewiesen. Die Klasse an sich wird mit vorangestelltem Punkt wie eine Normale CSS Deklaration angegeben.

### Ein Selektor - Mehrere Eigenschaften:

Wird einem Selektor mehrere Eigenschaften zugewiesen, so werden diese durch ein Semikolon getrennt. Im oberen Beispiel zwischen der Eigenschaft Color und font-weight.

# CSS Selektor ID

Die Deklaration von HTML Elementen mittels einer ID als Selektor unterscheidet sich von der mittels Klassen nur unwesentlich. Der einzige Unterschied besteht darin, dass innerhalb eines Dokumentes jede ID ein Unikat ist (jede ID darf nur einmal vorkommen) Eine ID wird mit voranstehendem Hash (#) definiert:

```
1 <html>
2 <head>
3   <style type="text/css">
4     <!--
5       #blau und dick { color: blue;
6                       font-weight:bold;
7                     }
8     -->
9   </style>
10 </head>
11 <body>
12   <h1> Klassen funktionieren so:</h1>
13   <p id="blau_und_dick">Erster Absatz</p>
14   <p>Zweiter Absatz</p>
15 </body>
```

Hier sehen Sie das [Ergebnis](#)

## Gruppieren von Selektoren

Selektoren können mittels Komma gruppiert werden, sodass alle vorangestellten Selektoren die selben Eigenschaften haben:

```
1 H1, H2, p, span { color:green }
```

Ohne Komma wirkt sich das gleich ganz anders aus:

Nehmen wir folgendes Beispiel, Sie benutzen häufig den strong-Tag, wollen aber nur wenn dieser innerhalb eines p-Tags vorkommt die Schriftfarbe ändern, so gehts:

```
1 <html>
2 <head>
3   <style type="text/css">
4     <!--
5       p strong { color: blue;
6                 font-weight:bold;
7               }
8     -->
9   </style>
10 </head>
11 <body>
12   <h1> Klassen funktionieren so:</h1>
13   <p>Erster Absatz mit <strong>dicker Auszeichnung</strong></p>
14   <p>Zweiter Absatz</p>
15   <strong>Dicke Auszeichnung</strong>
16 </body>
```

Hier sehen Sie das [Ergebnis](#)

### Erklärung:

Der Selektor in Zeile 5 sagt dem Browser, dass innerhalb eines Absatzes (p-Tag) jeder strong-Tag die Textfarbe Blau bekommt.

## In CSS-Code Kommentare setzen

So können Sie in einer CSS-Datei Kommentare setzen:

```
/* Kommentar, wird vom Browser nicht interpretiert, der Kommentar kann über
mehrere Zeilen gehen */
```