

CSS-Anweisungen

Aufbau einer CSS-Anweisung

Eine CSS-Anweisung setzt sich zusammen aus einem **Selektor** und den **Eigenschaften**, die der Selektor annehmen soll. Die zugewiesenen Eigenschaften werden in geschweiften Klammern { } hinter den Selektor geschrieben. Innerhalb der Klammern wird zunächst die Eigenschaft genannt, gefolgt von einem Doppelpunkt, dem Wert der Eigenschaft und einem Semikolon.

```
Selektor x {Eigenschaft: Wert;}
```

Der Selektor kann auch als Gruppe von Elementen angegeben werden. Das hat den Vorteil, dass man eine Anweisung, die für mehrere Elemente gleichermaßen gelten soll, im CSS nur einmal definieren muss.

Beispiel 1: Ein Selektor mit mehreren Eigenschaften

```
Selektor y {  
    Eigenschaft1: Wert-a;  
    Eigenschaft2: Wert-b;  
}
```

Beispiel 2: Eine Gruppe von Selektoren mit einer Eigenschaft

```
Selektor y, Selektor z {  
    Eigenschaft: Wert-c;  
}
```

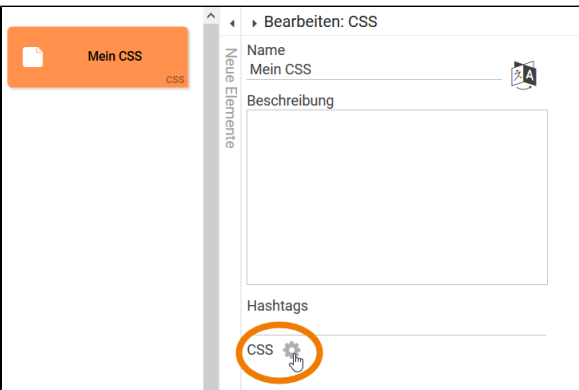
Bei der Eingabe der CSS-Anweisung sind Sie frei in der Gestaltung: Solange die Syntax eingehalten wird, können Sie den Leerraum um die Anweisung herum frei gestalten. Um Syntaxfehler zu vermeiden und die Übersichtlichkeit der Anweisungen zu erhöhen, werden zum Beispiel die Eigenschaften häufig in einzelne Zeilen geschrieben.

Beispiel 3: CSS-Anweisung für die Überschrift h1, jede Eigenschaft auf einer eigenen Zeile

```
h1 {  
    font-size: 1.8em;  
    margin: 0 5 px;  
    color: #3F84B4;  
    font-weight: bold;  
}
```

CSS-Anweisungen im CSS-Modell eintragen

Ihre Anweisungen geben Sie direkt im Editor des CSS-Modells ein.



Klicken Sie auf das Zahnrad **CSS**, um den Editor zu öffnen.

On this Page:

- [Aufbau einer CSS-Anweisung](#)
- [CSS-Anweisungen im CSS-Modell eintragen](#)
 - [CSS-Anweisungen zur feldgenauen Layout-Definition](#)
 - [Layout-Definition über den Feldbezeichner](#)
 - [Layout-Definition über eine CSS-Zusatzklasse](#)
 - [Häufig genutzte CSS-Eigenschaften](#)

Related Pages:

- [CSS-Elemente identifizieren](#)
- [CSS-Modell anlegen](#)
- [Widget-Tabelle](#)

<div data-bbox="157 138 735 323"><div>CSS Mein CSS</div><div></div><div><div>Speichern</div><div>Abbrechen</div></div></div>	<p>Der Editor öffnet sich in einem Pop-up-Fenster und ist direkt eingabebereit.</p>
<div data-bbox="157 413 735 737"><div>CSS Mein CSS</div><div><pre>1= body { 2 background-image: url(css/images/dark_red.png); 3 background-size: cover; 4 background-attachment: fixed; 5 background-position: center top; 6 background-repeat: no-repeat; 7 background-color: #f0f7f8; 8 position: relative; 9 border-top: 75px solid rgba(255,255,255,0.95); 10 } 11 12= body:before { 13 background-image: url(img/logos/logo.png); 14 background-size: contain; 15 background-position: center; 16 background-repeat: no-repeat; 17 background-color: transparent; 18 position: absolute; 19 /top: 30px;/ 20 top: -50px; 21 left: 30px; 22 height: 240px; 23 width: 130px; 24 content: ""; 25 }</pre></div><div><div>Speichern</div><div>Abbrechen</div></div></div>	<p>Tragen Sie Ihre Layoutvorgaben ein.</p> <p>Die entsprechenden Selektoren für Formularfelder können Sie über die Entwicklungswerkzeuge identifizieren oder in der Widget-Tabelle nachschauen.</p> <p>Klicken Sie im Anschluss auf Speichern.</p>

```
1 body {
2   background-color: #f0f0f0;
3   background-image: url(css/images/dark_red.png);
4   background-size: cover;
5   background-attachment: fixed;
6   background-position: center top;
7   background-repeat: no-repeat;
8   background-color: #f0f0f0;
9   position: relative;
10  border-top: 7px solid rgb(255,255,0,95);
11 }
```

SPEICHERN ABBRECHEN

Nutzen Sie **Kommentare**, um das CSS zu strukturieren und Anweisungen für bestimmte Elemente später leichter wiederzufinden.

Die Eingabe der Zeichenfolge **/*** kommentiert den nachfolgenden Inhalt aus, d.h. er wird bei der Anwendung des CSS ignoriert.

Beenden Sie den Kommentar mit ***/**, damit alle nachfolgenden Anweisungen berücksichtigt werden.



Ein Stylesheet darf beliebig viele Anweisungen enthalten.

Aber: Ein CSS-Modell wird von oben nach unten durchlaufen. Anweisungen, die weiter unten stehen, können die oben getroffenen Vorgaben übersteuern.

CSS-Anweisungen zur feldgenauen Layout-Definition

In einem CSS-Modell können Sie das Layout feldgenau festlegen. Dies kann hilfreich sein, wenn beispielsweise ein Feldbezeichner aufgrund des Hintergrunds schwer lesbar ist.


Beispiel: Die schwarze Überschrift der Eingabefelder **Vorname** und **Nachname** ist aufgrund des dunklen Hintergrunds nicht lesbar

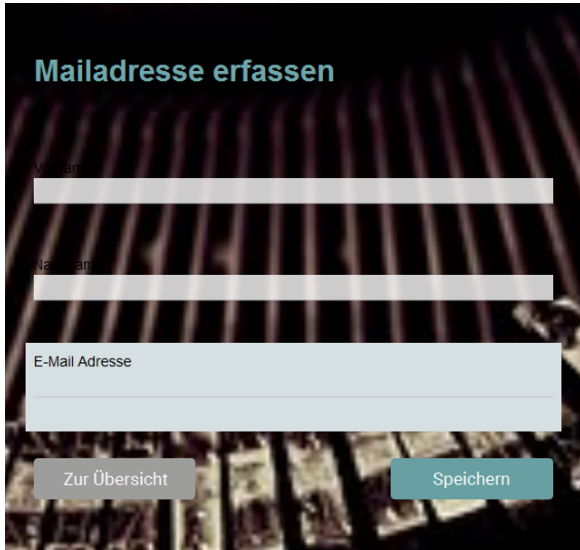



Für eine feldgenaue Definition des Layouts gibt es in einem CSS zwei Varianten: Man kann das Layout für ein Feld mit Hilfe seines eindeutigen Feldbezeichners festlegen oder eine Zusatzklasse definieren.

Layout-Definition über den Feldbezeichner

In dieser Variante wird das zu ändernde Feld im CSS über eine eindeutige ID, seinen Feldbezeichner, angesprochen.



	<p>Im Formular-Editor wurden die Eingabfelder Vorname, Nachname und E-Mail-Adresse angelegt.</p>
--	--

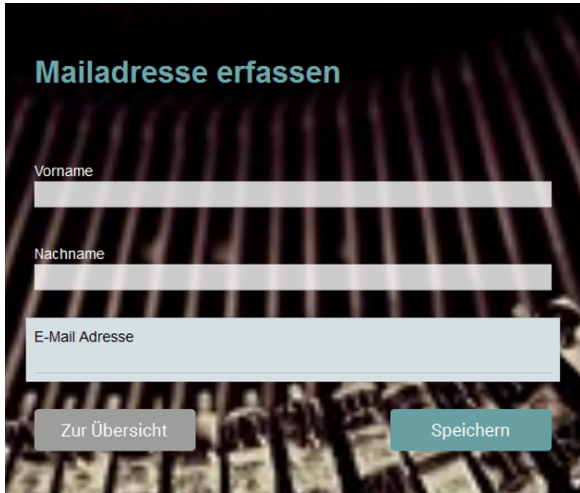
	<p>Das zugewiesene CSS enthält ein sehr dunkles Hintergrundbild. Die Überschriften der Felder Vorname und Nachname sind kaum lesbar.</p> <p>Damit man Vorname gut lesen kann, soll zunächst nur für dieses eine Feld die Schriftfarbe auf weiß geändert werden.</p>
<p>.lbn-Vorname</p>	<p>Der Feldbezeichner für die Überschrift eines Widgets lautet lb-. Über die Erweiterung n für name wird auf ein konkretes Formularfeld verwiesen.</p> <p>Die eindeutige ID für das Feld Vorname in unserem Beispiel ist also lbn-Vorname.</p>
<pre>369 /* Nur die Überschrift des Eingabefeldes "Vorname" auf weiß abändern */ 370 .lbn-Vorname { 371 color:white; 372 }</pre>	<p>Im CSS wird jetzt für das entsprechende Feld die Schriftfarbe weiß definiert.</p>

	Bei der Ausführung des Formulars ist die Überschrift nun in weiß viel besser lesbar.
---	--

Layout-Definition über eine CSS-Zusatzklasse

Eine zweite Variante, das Layout für ein bestimmtes Formularfeld festzulegen, ist das [Zuweisen einer Zusatzklasse](#). Zusatzklassen haben den Vorteil, dass man sie auch für weitere Felder verwenden kann, sollte dies im Laufe der Layoutarbeiten nötig sein. In unserem Beispiel sind von dem dunklen Hintergrundbild gleich zwei Eingabefelder betroffen. Statt im CSS für beide Felder jeweils eine weiße Überschrift zu definieren, reicht es aus, eine CSS-Zusatzklasse zu erstellen.

<code>.ueberschrift_weiss</code>	Im CSS werden Klassen mit einem vorangestellten Punkt angesprochen.
<pre>376 /* Definition einer Zusatzklasse namens ".ueberschrift_weiss" */ 377 .ueberschrift_weiss { 378 color:white; 379 }</pre>	Im CSS-Modell unseres Beispiels wird die Zusatzklasse <code>.ueberschrift_weiss</code> definiert.
	Anschließend wird die Zusatzklasse im Formular-Editor sowohl dem Eingabefeld 'Vorname...' als auch dem Eingabefeld 'Nachname' zugeordnet.
	... als auch dem Eingabefeld 'Nachname' zugeordnet.

	<p>Bei der Ausführung des Formulars werden beide Überschriften in weiß angezeigt.</p>
---	---

Häufig genutzte CSS-Eigenschaften

CSS-Eigenschaft	Auswirkung
color	Beschreibt die (Vordergrund-)Farbe, im allgemeinen die Schriftfarbe eines Elementes.
background	Passen Sie über background den Hintergrund Ihrer Elemente und Formulare an. background setzt sich aus mehreren Eigenschaften zusammen: background-color, background-image, background-repeat, background-attachment und background-position.
font	Beschreibt die Eigenschaften der verwendeten Schrift in der Reihenfolge: font-style, font-variant, font-weight, font-size und font-family.
!important	Diese Eigenschaft kennzeichnet einen Wert als wichtig. Da CSS-Anweisungen von oben nach unten durchlaufen werden, gilt im Normalfall der zuletzt festgelegte Wert. Deklariert man einen Wert mit Hilfe von !important als wichtig, so wird der Wert nicht durch eine später folgende, gleich gewichtete Anweisung überschrieben.
opacity	Mit dieser Eigenschaft kann man die Transparenz eines Elements einstellen. Die Eingabe erfolgt als Zahl zwischen 0 und 1 (nicht in Prozent!), wobei 1 für 100%ige Deckkraft steht, also keinerlei Transparenz.



Weiterführende Erläuterungen zu den Themen CSS, Syntax von CSS-Anweisungen, Selektoren und Eigenschaften finden Sie auf der Seite wiki.selfhtml.org oder unter www.w3schools.com (in englischer Sprache). Ein deutschsprachiges CSS-Tutorial bietet die Seite <http://de.html.net>.