

Clever gestalten

Sie wollen dem Design Ihrer Webseiten den letzten Schliff geben? Unser Workshop zeigt, wie Sie mit CSS perfekt Akzente setzen



HTML ist nicht der Weisheit letzter Schluss für die Gestaltung von Webseiten. Auf einzelne Elemente haben Sie nur sehr begrenzten Einfluss, und außerdem gelten viele HTML-Tags und -Attribute als veraltet. In den nächsten Versionen dieser Seitenbeschreibungssprache werden sie sicher entfernt.

Das ist nicht weiter schlimm, denn mit Cascading Stylesheets, kurz CSS, haben Sie viel mehr Freiheit. Sie dürfen etwa den Zeilenabstand ändern, die Hintergrundfarbe einzelner Wörter regeln oder Bilder an einer festen Stelle platzieren.

Zusätzlich sparen Sie sich eine Menge Arbeit bei eventuellen Anpassungen. So definieren Sie eine CSS-Regel nur einmal am Anfang der HTML-Datei, damit sie für die gesamte Website gilt. Noch einfacher

wird es, wenn Sie die CSS-Angaben in eine separate Datei auslagern, die dann alle Webseiten Ihres Online-Auftritts gemeinsam nutzen. Die gezeigten Beispiele liegen auf der Heft-CD unter „HomeP@ge“, „Praxis & Tuning“ für Sie bereit.

CSS im HTML-Tag

Die Grundlagen von CSS sind leicht zu verstehen. Als Erstes fügen Sie eine CSS-Eigenschaft und den gewünschten Wert in Ihren HTML-Code ein. Für die Farbe heißt die Eigenschaft schlicht **color**. Erlaubt sind wie in HTML Farbnamen (zum Beispiel **red**) oder RGB-Zahlenangaben (etwa **#0000FF** für Blau). Eine komplette CSS-Deklaration sieht damit so aus:

```
color: red;
```

Die Eigenschaft und den Wert trennen Sie durch einen Doppelpunkt und schließen die Angaben am Ende mit einem Strichpunkt ab.

Aber woher weiß der Browser, was er im Beispiel rot einfärben soll? Im einfachsten Fall fügen Sie die CSS-Deklaration direkt in das betroffene HTML-Tag ein. Mit dem folgenden Beispiel färben Sie einen ganzen Absatz rot ein:

```
<p style="color: red;">
Textabsatz ...
</p>
```

Mit dem Attribut **style** geben Sie die CSS-Deklaration in Anführungszeichen an. Damit zeigt der Browser alle Texte zwischen **<p>** und **</p>** in roter Schriftfarbe an.

Es ist durchaus erlaubt, mehrere CSS-Deklarationen in einem **style**-Attribut

anzugeben. Die Schriftgröße steuern Sie zum Beispiel mit `font-size: 18pt`. Damit erhalten Sie eine Schrift von 18 Punkt Zeichengröße in roter Farbe:

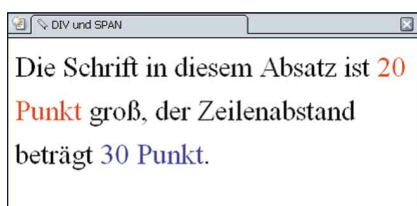
```
<p style="color: red; font-size: 18pt;">
```

<div> und

Was machen Sie, wenn Sie nicht einen ganzen Absatz, sondern nur einzelne Wörter formatieren wollen? Denn es ist immer ein HTML-Tag nötig, auf das Sie die CSS-Deklaration anwenden.

Speziell für diesen Zweck gibt es in HTML zwei besondere Tags: `<div>` und ``. Diese beiden Tags dienen nur dazu, besondere Bereiche für CSS zu markieren. `<div>` entspricht in etwa dem normalen Absatz-Tag. Davor und danach fügt der Browser also einen Zeilenumbruch ein. Mit dem Tag `` markieren Sie einzelne Elemente im Fließtext; es gibt dadurch keinen Zeilenumbruch.

Das Beispiel zeigt einen ganzen `<div>`-Absatz mit einer Schriftgröße von 20 Punkt und einem Zeilenabstand von 30 Punkt (`line-height: 30pt`;) formatiert. Zwei Abschnitte im Text haben blaue beziehungsweise rote Schriftfarbe:



In einem Absatz sind mit `` zwei Bereiche mit unterschiedlicher Schriftfarbe formatiert

```
<div style="font-size: 20pt; line-height: 30pt;">
  Die Schrift in diesem Absatz ist
  <span style="color: red;">20
  Punkt</span>
  groß, der Zeilenabstand beträgt
  <span style="color: blue;">30
  Punkt</span>.
</div>
```

Stylesheet-Bereich

Manchmal ist es nützlich, eine CSS-Deklaration direkt in ein HTML-Tag einzu-

Interessante Webadressen

Insgesamt gibt es weit über hundert CSS-Eigenschaften (CSS Level 1 und CSS Level 2), die eine Vielzahl von Werten erlauben. Allerdings unterstützen nicht alle Browser sämtliche Möglichkeiten. Eine Übersicht der CSS-Eigenschaften mit Angaben zur Browser-Unterstützung liefern folgende deutschsprachige Webadressen:

■ <http://selfhtml.teamone.de>

Self-HTML: Die Referenz schlechthin für HTML und CSS

■ www.bjoernsworld.de

Bjoernsworld.de: umfangreiche Details zu CSS mit Tipps zu häufigen anzutreffenden Fehlern

fügen. Besser ist es aber, die Deklaration ganz am Anfang der HTML-Datei zu schreiben. Die Deklaration gilt dann für das gesamte HTML-Dokument. Wenn Sie etwas ändern wollen, brauchen Sie das nur an einer Stelle und nicht an dutzenden quer durch den HTML-Code zu tun. Dazu definieren Sie im Kopf der HTML-Datei einen Stylesheet-Bereich:

```
<head>
  <style type="text/css">
    h1 {color: red;}
  </style>
</head>
```

Das Tag-Paar `<style>` und `</style>` schließt den Bereich ein. Alles dazwischen interpretiert der Browser nicht als Text, sondern als CSS-Angaben.

Das Beispiel enthält die CSS-Deklaration `color: red`, die Sie bereits kennen. Sie ist in geschweifte Klammern eingefasst, und davor steht ein so genannter Selektor. Er sagt dem Browser, auf welche Teile der Webseite er die CSS-Angaben anwenden soll. `h1` bedeutet somit, dass der Browser sämtliche Überschriften erster Ordnung (die mit dem Tag `<h1>` markiert sind) im Dokument rot einfärben soll. Selektor und CSS-Deklaration zusammen nennt man übrigens CSS-Regel.

Gesammelte Eigenschaften

In vielen Fällen wollen Sie eine Deklaration auf verschiedene HTML-Tags anwenden. Dann müssen Sie nicht mehrere komplette CSS-Regeln tippen, sondern Sie tragen stattdessen einfach alle HTML-Tags als Selektor ein:

```
h1, h2, h3, h4, h5, h6 {color: red;}
```

Damit erfassen Sie sämtliche Überschriften von `<h1>` bis `<h6>` und färben sie rot. Natürlich dürfen Sie auch andere Tags kombinieren, beispielsweise `<p>`, `<div>` und `<table>`.

Genauso funktioniert es, mehrere CSS-Deklarationen auf einmal anzuführen. Trennen Sie die einzelnen Angaben einfach durch einen Strichpunkt. Der Quelltext wird übersichtlicher, wenn Sie jede Deklaration in eine eigene Zeile schreiben.

```
<style type="text/css">
  h1, h2, h3, h4, h5, h6 {
    color: red;
  }
  p {
    font-family: sans-serif;
    font-size: 9pt;
    text-indent: 5mm;
    text-align: justify;
  }
</style>
```

Das erzeugt Überschriften in roter Schrift, der restliche Text innerhalb von Absätzen erhält eine Schriftart ohne Serifen ►



Schnell zum einheitlichen Layout: Über eine globale CSS-Regel sind alle Absätze eingerückt und im Blocksatz dargestellt

(**font-family: sans-serif;**), welche neun Punkt groß ist.

Interessant ist die CSS-Angabe **text-indent**. Damit rückt der Browser die erste Zeile im Absatz um fünf Millimeter ein. Mit reinem HTML wäre dieser Effekt nur mit Klimmzügen zu realisieren (unsichtbare Grafik als Platzhalter). **text-align: justify** sorgt schließlich für den Blocksatz, also einen sauberen linken und rechten Rand beim Text.

Nur Klassen formatieren

In vielen Fällen bringt es keine Vorteile, einfach die Absätze oder Überschriften im Text einheitlich zu formatieren. Sie wollen vielleicht einen wichtigen Absatz bewusst besonders hervorheben oder mit gutem Grund nicht alle Überschriften dritter Ordnung im selben Stil gestalten.

Für diesen Effekt ändern Sie die CSS-Regeln ein klein wenig ab. Markieren Sie im HTML-Dokument alle Tags, die Sie mit einer bestimmten CSS-Regel formatieren wollen. Dazu tragen Sie statt des HTML-Tags als Selektor einen Punkt gefolgt von einer Bezeichnung ein. Das nennt man einen Klassenselektor:

```
.rot {color: red;}
```

Diese CSS-Regel enthält keine Angabe, auf welche Tags der Browser sie anwenden soll. Deshalb müssen Sie selbst die betreffenden Tags kennzeichnen. Durch diese Zusammengehörigkeit bilden sich ver-

schiedene Klassen. Somit wird auch klar, warum es Klassenselektor heißt.

```
<h1 class="rot">Text</h1>
```

So fügen Sie in ein Tag das Attribut **class** ein. Den Namen des Klassenselektors, im Beispiel **rot** (ohne den Punkt am Anfang), geben Sie als Wert in Anführungszeichen an. Jetzt weiß der Browser, dass er die Überschrift rot formatieren soll. Übrigens bleibt die Wahl des Namens für den Selektor Ihnen überlassen. Sie könnten ihn genauso gut *ueberschrift* nennen.

Sorgfältig planen

Das **class**-Attribut fügen Sie in jedes HTML-Tag ein, das Text in roter Schrift aufweisen soll. Da es sich nicht unbedingt um eine Überschrift handeln muss, haben Sie jetzt ein universelles CSS-Format für roten Text. Sie sollten allerdings genau überlegen, an welchen Stellen Sie ein CSS-Format mit **class** einfügen. Denn wenn Sie später beispielsweise die Farbe der Überschriften verändern wollen und dazu die CSS-Regel am Dateianfang anpassen, beeinflusst das auch alle anderen Stellen mit **class="rot"**.

Praktischer ist es, von Anfang an zwei CSS-Regeln für rote Überschriften und rote Textstellen zu definieren, etwa **.rotH** und **.rotText**. Beachten Sie außerdem, dass in den Klassennamen nur die Buchstaben a bis z und A bis Z erlaubt sind – keine Zahlen oder Sonderzeichen.

Die erste Zeile: Pseudo-Element

Manche Elemente einer Webseite lassen sich nicht über ein HTML-Tag ansprechen. Das typische Beispiel dafür ist die erste Zeile eines Absatzes. Welche Buchstaben zur ersten Zeile gehören, ändert sich je nach Browser und Fensterbreite. Für diese Fälle kennt CSS so genannte Pseudo-Elemente. Mit **:first-letter** (erster Buchstabe) und **:first-line** (erste Zeile) bauen Sie einen Selektor zusammen:

```
p:first-letter {
    font-size: 300%;
    float: left;
}
```

Das Beispiel erzeugt ein Initial am Textanfang. Die Eigenschaft **float: left** bewirkt, dass der Text im Absatz rechts am Initial vorbeifließt.

Damit Sie bei den vielen Klassenselektoren den Überblick nicht verlieren, fügen Sie Kommentare ein. Ein solcher beginnt mit **/*** und endet mit ***/**. Alle Zeichen dazwischen ignoriert der Browser. Auf den ersten Blick wirkt das vielleicht unnötig, aber Sie erleichtern sich mit diesen Eingaben die Arbeit in der Zukunft. Bearbeiten Sie nach einem halben Jahr an Ihrer Homepage eine umfangreiche CSS-Datei, sind Sie über die Kommentare froh. Schnell bekommen Sie einen Überblick über Ihr Website-Layout.

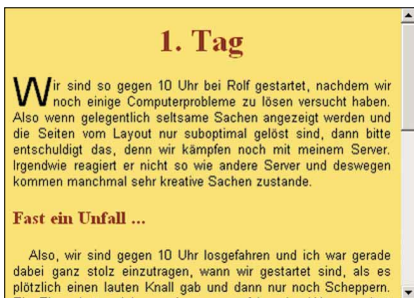
Wichtige CSS-Eigenschaften

CSS-Eigenschaft	Beispiel	Wirkung
background-color	background-color: green;	Hintergrundfarbe des Elements; erlaubt sind Farbnamen und Farbwerte im RGB-Format (zum Beispiel #FF00BB)
background-image	background-image: url(bild.jpg);	Hintergrundbild des Elements; den Dateinamen müssen Sie mit der Schreibweise url(...) angeben
color	color: red;	Schriftfarbe
float	float: left	Regelt den Textfluss um das Element (bei left fließt der Text rechts vorbei)
font-family	font-family: serif;	Schriftfamilie für den Text (serif : mit Serifen, sans-serif : ohne Serifen, cursive : kursive Schrift, fantasy : Schmuckschrift oder Sonderzeichen, monospace : keine proportionale Schrift)
font-size	font-size: 12pt;	Schriftgröße, im Beispiel mit Punktangabe
text-align	text-align: justify;	Ausrichtung für Text (left : linksbündig, right : rechtsbündig, justify : Blocksatz)
text-indent	text-indent: 1cm;	Erste Zeile des Absatzes einen Zentimeter einrücken

Gemeinsam nutzen

Die CSS-Regeln im Stylesheet-Bereich vereinfachen die Pflege einer HTML-Datei schon erheblich. Ein Projekt für eine komplette Website besteht aber meist aus vielen HTML-Dateien, die Sie dann einzeln anpassen müssen. Auch dafür bietet CSS eine praktische Lösung: Lagern Sie die CSS-Regeln in eine externe Datei aus, die von allen HTML-Seiten gemeinsam genutzt wird.

Dazu erzeugen Sie zuerst eine neue Datei namens *format.css* für die CSS-Regeln. Der Dateiname ist beliebig, und mit der



Diese Seite verwendet eine externe CSS-Datei mit Anweisungen für Überschrift und Textinitial

Erweiterung *css* vermeiden Sie Verwechslungen. In diese Datei kopieren Sie den Inhalt des Stylesheet-Bereichs, also ohne die HTML-Zeilen `<style ...>` und `</style>`.

In der HTML-Datei fügen Sie in den Kopf zwischen `<head>` und `</head>` folgende Zeile ein:

```
<link href="format.css"
rel="stylesheet" type="text/css">
```

Die einzige Anpassung betrifft den Dateinamen hinter **href**. Tragen Sie hier den Namen Ihrer CSS-Datei ein.

In den meisten Fällen ist es sinnvoll, gleich mehrere CSS-Dateien anzulegen. Das erleichtert es Ihnen, in die einzelnen HTML-Seiten nur die externen CSS-Regeln einzubinden, die Sie wirklich benötigen. Dabei haben Sie die Möglichkeit, beliebig viele CSS-Dateien in einem einzelnen HTML-File zu verwenden. Schreiben Sie dazu einfach für jede Datei ein `<link>`-Tag. ■

Wolfgang Nefzger/Johann Sedlbauer
homepage@com-online.de

Anzeige