

ARBEITEN MIT CSS

Ordnung halten

Je mehr Sie mit CSS arbeiten, desto länger und unübersichtlicher werden auch Ihre Stylesheets. Dieser Artikel gibt Ihnen nützliche Tipps zur Organisation von CSS-Regeln.

Allein am Kommentar im Dateinamen eines Stylesheets kann man nicht immer erkennen, zu welcher Site es gehört und welchen Zweck es erfüllt. Darum empfiehlt es sich, die wichtigsten Informationen in einem Kommentar am Anfang der CSS-Datei festzuhalten. Wie ausführlich dieser Kommentar ausfallen sollte, hängt vom Einsatzzweck ab. Zwei Beispiele sollen das verdeutlichen. Zunächst zeigen wir Ihnen die Minimalvariante:

```
/* Stylesheet für Artikel CHIP
Sonderheft Homepage, November
2006 */
Wenn Sie dieses Stylesheet in ein paar Wochen oder Monaten wieder öffnen, wissen Sie wenigstens noch, wann und wofür es geschrieben wurde. Die folgende Variante erledigt dieselbe Aufgabe, liefert aber mehr Informationen:
/* =====
CHIP Sonderheft Homepage -
Artikel - CSS: Styleguide für
Stylesheets
Datei: chip.css
Media: screen
Datum: 11. November 2006
Autor: Peter Müller
Mail: benachrichtigung@infotek-
ten.de
Aufbau: 1. Allgemeine Styles
- Block-Elemente
- Inline-Elemente
2. Styles für Layoutbereiche
```

```
- Wrapper
- Kopfbereich
- Navibereich
- Textbereich
- Fussbereich
===== */
```

Je nach Komplexität des Projekts können Sie beispielsweise auch das verwendete Farbschema oder eine Liste der Grafikdateien anführen, auf die das Sheet verlinkt.

Stylesheet in mehrere Abschnitte einteilen

Ein Stylesheet besteht aus CSS-Regeln, die auch Styles genannt werden. Es ist nicht selten mehrere hundert Zeilen lang. Versuchen Sie deshalb von Anfang an, das CSS so übersichtlich wie nur irgend möglich zu halten.

Der erste Schritt zu mehr Ordnung ist die Einteilung des Sheets in Abschnitte. Dabei bieten sich drei Hauptbereiche an:

- allgemeine Styles, die die Grundeinstellungen der Seiten definieren,
- Styles für Layout-Bereiche wie „#kopfbereich“ oder „#navibereich li“,
- alle anderen Styles, die nur auf bestimmten Seiten vorkommen (beispielsweise Kontaktformulare) oder nicht in die beiden anderen Abschnitte passen.

Den Beginn eines Abschnitts kennzeichnen Sie im Stylesheet durch einen mehrzeiligen Kommentar. Dabei können Sie auch beliebige Sonderzeichen verwenden. Wichtig ist nur, dass die gesamte Konstruktion mit „/*“ beginnt und mit „*/“ endet. Hier ein Beispiel:

```
/* =====
1. Allgemeine Styles
===== */
```

In diesem Abschnitt definieren Sie die allgemeinen, für die ganze Seite geltenden Einstellungen. Notieren Sie sich zuerst die Block/„body“-Elemente – Überschriften, Absätze, Listen, Zitate etc. – und dann die

Inline-Elemente, etwa Hyperlinks, Bilder und Elemente wie „strong“ und „em“.

Im zweiten Abschnitt folgen Styles, die sich nur auf einen bestimmten Bereich der Seite beziehen:

```
/* =====
2. Styles für Layoutbereiche
===== */
```

Am einfachsten führen Sie die Bereiche in der gleichen Reihenfolge auf, in der sie auch im HTML-Quelltext erscheinen. Bei den Beispielseiten aus dem CSS-Artikel ab 32 in diesem Heft wären das beispielsweise „#wrapper“, „#kopfbereich“, „#navibereich“, „#textbereich“ und „#fussbereich“. Verschachtelte Selektoren, die nur für einen bestimmten Bereich gelten, rücken Sie leicht ein:

```
/* WRAPPER */
#wrapper { ... }
/* KOPFBEREICH */
#kopfbereich { ... }
#kopfbereich h1 { ... }
#kopfbereich p { ... }
/* NAVIBEREICH */
#navibereich { ... }
```

Einen sinnvollen Aufbau für die Styles wählen

Darüber hinaus gibt es auch für die Definition der Stylesheets selbst einige Tricks, mit denen Sie die Übersicht verbessern können. Hier ein einfaches Beispiel:

```
body {
color: black;
background-color: silver;
font-family: Verdana, Arial,
Helvetica, sans-serif;
font-size: small;
padding: 10px;
}
```

In der ersten Zeile stehen der Selektor „body“ und die öffnende geschweifte Klammer. Danach folgt, etwas eingerückt, die erste Deklaration, die immer aus einer

Eigenschaft wie beispielsweise „color“, einem Doppelpunkt und einem Wert wie etwa „black“ besteht. Um die Styles möglichst übersichtlich zu halten, empfiehlt sich folgende Aufteilung:

- In jeder Zeile steht lediglich eine Deklaration, und jede Deklaration endet mit einem Semikolon.
- Eigenschaft und Wert werden durch einen Doppelpunkt getrennt, danach folgt ein Leerzeichen. Das ist zwar für das Funktionieren des Styles nicht zwingend erforderlich, erhöht aber die Lesbarkeit.
- Die geschweifte Klammer am Schluss des Listings bekommt eine eigene Zeile und schließt bündig mit dem ersten Buchstaben des Selektors ab.

Bei Styles mit lediglich einer Deklaration liegt es allerdings nahe, den gesamten Ausdruck in eine Zeile zu schreiben:

```
h1 { font-size: 150%; }
h1 {
font-size: 150%;
}
```

Der Einzeiler spart aber Platz und ist dennoch übersichtlich genug. Manche Autoren schreiben auch Styles mit nur einer Deklaration konsequent nach dem mehrzeiligen Schema. Das ist jedoch eine reine Geschmackssache.

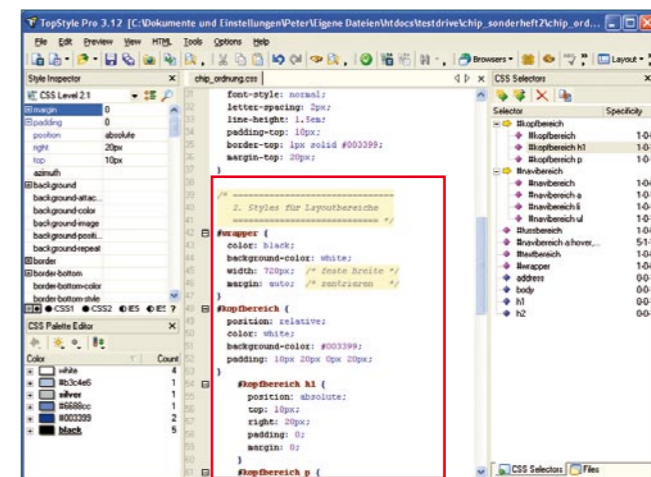
Reihenfolge der Deklarationen im Style festlegen

Zum Abschluss ein paar Anregungen zur Reihenfolge der Deklarationen innerhalb eines Styles. Schauen Sie sich dazu die folgende CSS-Regel an:

```
#kopfbereich {
position: relative;
font-size: 100%;
letter-spacing: 2px;
background-color: white;
border-top: 1px solid #003399;
padding-top: 10px;
margin-top: 20px;
}
```

Die Reihenfolge der Deklarationen orientiert sich am Aufbau des Box-Modells:

- Angaben wie „float“ oder „position“ und die zugehörigen Deklarationen wie „top“, „right“, „bottom“ und „left“ bestimmen die Position des Kästchens auf der Seite und stehen daher ganz am Anfang der Regel.



CSS geordnet: So sieht das Stylesheet aus dem Artikel ab 32 aus. Verschachtelte Selektoren, etwa im „kopfbereich“, rücken Sie leicht ein.

- Nach einer optionalen Leerzeile folgen die Eigenschaften für den Inhaltsbereich, also die Deklarationen wie beispielsweise „font-size“, „font-style“, „letter-spacing“ und so weiter.

- Nach einer weiteren optionalen Leerzeile listet die Regel die Definitionen für das Kästchen selbst auf: „width“, „height“, „background“, „padding“, „border“ und „margin“. Diese Reihenfolge sollte übrigens immer eingehalten werden.

In der am Box-Modell orientierten Sortierung beschreiben die Deklarationen also zuerst den Standort auf der Seite und dann, von innen nach außen, das Element selbst. So verinnerlichen Sie beim Ordnen der Deklarationen ganz nebenbei auch den Aufbau des Box-Modells und können sich das Aussehen des Elements vor Ihrem inneren Auge besser vorstellen.

Eine andere, weit verbreitete Methode zum Auflisten von Deklarationen ist die alphabetische Sortierung. Das folgende Listing beschreibt dieselbe CSS-Regel wie in der Textspalte zuvor:

```
#kopfbereich {
background-color: white;
border-top: 1px solid #003399;
font-size: 100%;
letter-spacing: 2px;
margin-top: 20px;
padding-top: 10px;
position: relative;
}
```

```
font-size: 100%;
letter-spacing: 2px;
margin-top: 20px;
padding-top: 10px;
position: relative;
}
```

Der Vorteil der alphabetischen Sortierung ist, dass die Reihenfolge immer eindeutig und leicht einzuhalten ist. Ihr Nachteil liegt darin, dass verwandte Eigenschaften wie „background“ und „padding“ auseinandergerissen werden und wichtige Positionsangaben wie etwa „float“ oder „position“ zwischen den anderen Eigenschaften vergraben liegen.

Probieren Sie aus, welche der beiden Methoden Ihnen besser gefällt: die alphabetische Sortierung oder das Box-Modell. Sie können sich auch etwas Neues ausdenken, wichtig ist lediglich, dass Sie dem einmal gewählten Schema treu bleiben. Nur so ist gewährleistet, dass Sie sich im Quelltext Ihrer Websites auch später noch zurechtfinden. Wie penibel Sie den selbst gewählten Regeln folgen, hängt natürlich in jedem Fall auch vom Umfang des Projekts ab und von Ihrem eigenen Ordnungsbedürfnis.

Peter Müller

TIPPS

Schnelle Kommentare

Kommentare stehen zwischen den Markierungen „/*“ und „*/“. Über den Ziffernblock auf der Tastatur sind die Zeichen schnell eingegeben, da die beiden zuständigen Tasten direkt nebeneinander liegen: Den Schrägstrich erreichen Sie über das Divisionszeichen, das Sternchen über einen Druck auf die Taste mit dem Multiplikationszeichen.

Weitere Infos

Das Box-Modell beschreibt den Aufbau eines HTML-Elements und ist eine der wichtigsten Grundlagen zum Arbeiten mit CSS. Im CSS-Artikel aus dem Sonderheft „Die perfekte Homepage“ auf der CD finden Sie eine genauere Erklärung. Ein Schema zum Ausdrucken gibt es im Internet unter der Adresse <http://little-boxes.de/download.php>.

AUF EINEN BLICK

→ Arbeiten mit CSS

Deklarationen logisch oder alphabetisch ordnen 41

Kommentare schnell eingeben 41

Alle Tools auf CD

PDF: Artikel aus dem Sonderheft „Die perfekte Homepage“ © Vollversionen