

Tabellenlayout mit CSS

Frischkur für Zellen

Mit Cascading Stylesheets peppen Sie langweilige Tabellen auf. Dazu färben Sie Zeilen oder animieren sie. Darüber hinaus sorgen Sie dafür, dass Tabellen schneller laden

Niemand liest gern Tabellen. Oft wirken sie unübersichtlich und sind schlecht zu lesen. Wenn die Tabelle dann noch lange braucht, bis sie sich aufbaut, ist die Geduld des Surfers schnell verbraucht.

Doch es geht es auch anders: Wenige, geschickt eingesetzte CSS-Befehle (Cascading Stylesheets) bringen Farbe und Übersicht in Ihre Tabellen. Wir zeigen Ihnen im Folgenden, wie Sie Tabellenzeilen färben oder Zeilen so einrichten, dass sie die Farbe wechseln, wenn die Maus darüber fährt. Darüber hinaus können Sie Tabellen so präparieren, dass sie schneller laden. Mit Stylesheets haben Sie hierzu das richtige Werkzeug in der Hand.

ZEILEN UNTERSCHIEDLICH FÄRBEN

Lange Tabellenzeilen wirken meist unübersichtlich. Es strengt an, den Zeilen zu folgen und den Inhalt zu lesen. Hier bewährt es sich, den Hintergrund von Zeilen zu färben: Jede zweite Zeile erhält einen anderen Farbton, auch Zwischenüberschriften lassen sich so deutlich markieren.

Dafür ist nicht unbedingt CSS notwendig, denn das HTML-Attribut `bgcolor` gilt auch für das Zeilen-Tag `<tr>`. Allerdings ist das Attribut in HTML 4.0 als unerwünscht gekennzeichnet und soll in künftigen HTML-Versionen entfallen. Außerdem müssen Sie mit HTML jede Zeile ein-

zeln färben – das macht es mühsam, die Tabelle später zu ändern.

Mit CSS definieren Sie dagegen am Anfang der HTML-Datei eine Regel, deren Namen Sie dann in die `<tr>`-Tags eintragen – fertig. Sie geben die Farbe also nur an einer einzigen Stelle an, das ist später leicht zu ändern.

Zunächst definieren Sie im `<head>`-Bereich drei CSS-Regeln für zwei Zeilenfarben und für die erste Zeile in der Tabelle:

```
<style type="text/css">
.zeile1 {background-color:
#FF9900;}
.zeile2 {background-color:
#FFCC66;}
.erste {background-color:
yellow;}
</style>
```

Der führende Punkt vor den CSS-Regeln erzeugt einen so genannten Klassenselektor. Um diesen zu nutzen, tragen Sie nur

das Attribut `class="zeile1"` in einem HTML-Tag ein. Damit wendet der Browser diese Regel an. Eine Tabelle sieht dann folgendermaßen aus:

```
<table>
<tr class="erste"></tr>
<tr class="zeile1"></tr>
<tr class="zeile2"></tr>
</table>
```

Damit färben Sie den Kopf der Tabelle gelb, die zweite und dritte Zeile leuchten in verschiedenen Orangetönen. In der nächsten Zeile verwenden Sie wieder `class="zeile1"` und so weiter. Natürlich können Sie wie gezeigt auch zwei Zeilen gleich färben oder nur Zwischenüberschriften farblich hervorheben. Die Welt der Farben steht Ihnen offen.

Diese Lösung weist noch einen Makel auf: In der Tabelle trennen Streifen, die die Farbe des Hintergrunds annehmen, die Zellen. Um diese störenden Streifen zu entfernen, setzen Sie das HTML-Attribut `cellspacing="0"` im `<table>`-Tag ein. Zwar gibt es eine entsprechende CSS-Eigenschaft (`border-spacing`), diese unterstützen jedoch nur Netscape und Mozilla.

Mit diesem Trick kleben aber die Texte in den Zellen zu nah aneinander. Bauen Sie daher mit dem Attribut `cellpadding` einen zusätzlichen Abstand ein. Das Tabellen-Tag sieht dann wie folgt aus:

```
<table border=0 cellspacing="0"
cellpadding="10">
```

Farbige Zeilen

Normale Darstellung

	Wochentag	Zeit	Halle
Kinder 3 - 6 Jahre	Montag	16.30 - 17.30	Donnerstr.
Kinder 4 - 6 Jahre	Montag	14.15 - 16.15	Frohnstr.
Kinder 5 - 8 Jahre	Dienstag	15.30 - 16.30	Pannerstr.
Kinder 5 - 9 Jahre	Donnerstag	16.30 - 18.00	Nierenhof
Kinder 6 - 8 Jahre	Montag	16.15 - 17.15	Frohnstr.
Kinder 8 - 10 Jahre	Montag	17.15 - 18.15	Frohnstr.
Gerätestunden für Mädchen ab 8 Jahre	Mittwoch	16.30 - 18.00	Frohnstraße
Turnen für Jungen im Alter von 7 bis 12 Jahren	Montag	16.00 - 18.00	Kuhstraße

Mit CSS-Formatierung

	Wochentag	Zeit	Halle
Kinder 3 - 6 Jahre	Montag	16.30 - 17.30	Donnerstr.
Kinder 4 - 6 Jahre	Montag	14.15 - 16.15	Frohnstr.
Kinder 5 - 8 Jahre	Dienstag	15.30 - 16.30	Pannerstr.
Kinder 5 - 9 Jahre	Donnerstag	16.30 - 18.00	Nierenhof
Kinder 6 - 8 Jahre	Montag	16.15 - 17.15	Frohnstr.
Kinder 8 - 10 Jahre	Montag	17.15 - 18.15	Frohnstr.
Gerätestunden für Mädchen ab 8 Jahre	Mittwoch	16.30 - 18.00	Frohnstraße
Turnen für Jungen im Alter von 7 bis 12 Jahren	Montag	16.00 - 18.00	Kuhstraße

Oben sehen Sie eine normale HTML-Tabelle, unten eine Tabelle, die mit CSS formatiert wurde

Wollen Sie die farbigen Tabellen auf mehreren Webseiten einsetzen, bietet es sich an, die CSS-Regeln in eine externe Datei auszulagern. Dazu speichern Sie die drei Zeilen aus dem Style-Bereich ohne die beiden Tags `<style>` und `</style>` in einer Textdatei. Nennen Sie diese beispielsweise `tabellen.css`.

Löschen Sie danach den gesamten CSS-Block von `<style>` bis `</style>` und fügen Sie statt dessen im Kopf der HTML-Datei folgende Zeile ein:

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

ANIMIERTE TABELLENZEILEN

In umfangreichen Tabellen kann der zweifarbige Hintergrund jedoch unangenehm wirken. Dann stechen die bunten Streifen heraus und man fühlt sich wie vor einer heruntergelassenen Jalousie. Hier gehen Sie dezent zu Werke. Dazu bieten Sie Ihren Besuchern eine kleine Animation an: Sobald diese den Mauszeiger über eine Tabellenzeile bewegen, färbt sich diese. Damit lässt sich eine große Tabelle leichter auslesen.

Das funktioniert mit CSS und zwei Event-Handlern. Ein Event-Handler löst eine Aktion aus, sobald ein festgelegtes Ereignis eintritt, der Handler `onMouseOver` wird beispielsweise aktiv, wenn sich der Mauszeiger über einen Bereich bewegt. Hingegen kommt `onMouseOut` zum Zug, wenn der Mauszeiger diesen Bereich wieder verlässt. Mit diesen Handlern gestalten Sie die Tabellenzeile wie folgt:

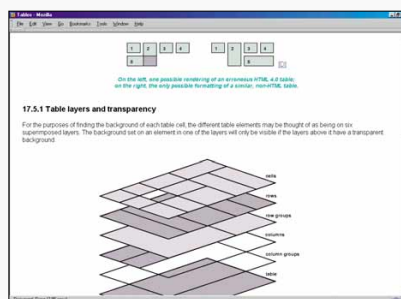
Animierte Tabellenzellen

	Jan	Peter
Januar	37	34
Februar	42	31
März	28	31
April	30	30
Mai	37	39
Juni	41	43
Juli	40	45
August	43	47
September	44	49
Oktober	47	52
November	48	54
Dezember	47	56

Fährt ein Besucher mit der Maus über eine Tabellenzeile, färbt der Browser sie

WIE SIE HÄUFIGE CSS-FEHLER ERKENNEN UND VERMEIDEN

Bei der Arbeit mit CSS-Regeln lauern einige Tücken: Als Trennzeichen zwischen CSS-Eigenschaft und Wert dient der Doppelpunkt. Wer oft HTML-Tags eingibt, tippt hier gerne versehentlich ein Gleichheitszeichen wie bei HTML-Attributen. Der Internet Explorer schluckt das, Netscape und Mozilla ignorieren damit jedoch den gesamten CSS-Befehl. Achten Sie bei CSS-Angaben zudem auf die Schreibweise in `class`- und `id`-Attributen: CSS unterscheidet Groß- und Kleinschreibung und erlaubt keine führenden Zahlen. Eine Angabe wie `class="1zeile"` bereitet Probleme, `class="erstezeile"` ist besser.



Umfangreiche Informationen über Tabellen und CSS finden Sie auf der Seite www.w3.org/TR/REC-CSS2/tables.html

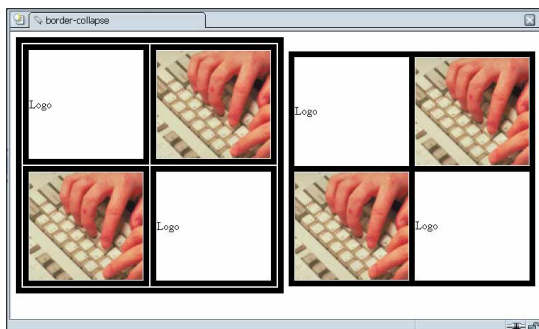
```
<tr onMouseOver="..."
onMouseOut="...">
```

Aber wie bringen Sie den Browser dazu, die Zeile zu färben? Dazu weisen Sie in den Anführungszeichen hinter dem Handler einer Aktion einen Wert zu:

```
this.style.backgroundColor='red';
```

Das Schlüsselwort `this` adressiert das aktuelle Element, also das Zeilen-Tag `<tr>`. Mit `style` legen Sie fest, dass Sie eine CSS-Eigenschaft ändern wollen, und diese Eigenschaft ist `backgroundColor`. Damit weisen Sie einen neuen Farbwert zu, im Beispiel `'red'`.

Beachten Sie die Schreibweise der CSS-Eigenschaft: Zum einen fehlt der Trennstrich in der Art `background-color`, den Sie in CSS einfügen müssten. Diesen ►



Mit zusammengefassten Rahmenlinien wie auf der rechten Seite sieht die Tabelle deutlich besser aus

Strich interpretiert HTML aber als Trennzeichen – und der CSS-Befehl würde nicht funktionieren. Zum anderen muss das `c` in `Color` groß geschrieben sein, die Browser ignorieren sonst die Angabe.

Bei `onMouseOut` setzen Sie die Farbe wieder auf den Standardwert, im Beispiel `'silver'`. Damit nimmt das Tabellen-Tag folgende Form an:

```
<tr onMouseOver="this.style.backgroundColor='red';" onMouseOut="this.style.backgroundColor='silver';">
```

Stellen Sie schließlich die Grundfarbe der Tabellenzellen auf `silver`.

An den Zellenrändern erscheinen noch hässliche Trennstriche in der Farbe des Seitenhintergrunds. Diese blenden Sie mit einem zusätzlichen Attribut aus:

```
<table cellspacing="0">
```

Die Animation funktioniert in allen aktuellen Browsern außer Opera.

RAHMENLINIEN AUFPEPPEN

Die Trennlinien sehen nicht schön aus: Jede Zelle hat einen eigenen Rahmen, also trennen zwei Linien die Zellen. Gerade bei dickeren Linien sieht das hässlich aus.

Um den Rahmen schön zu designen, setzen Sie die CSS-Eigenschaft `border-collapse: collapse` ein. Diese fasst die beiden Linien zwischen zwei Zellen zu einer Linie zusammen. Dazu ergänzen Sie das Tabellen-Tag um folgende CSS-Angabe:

```
<table style="border-collapse: collapse; border="8">
```

Das Attribut `width`, mit dem Sie die Breite der Linien festlegen, lässt sich ebenfalls

durch eine CSS-Eigenschaft ersetzen. Allerdings genügt es nicht, die Eigenschaft `border` im Tabellen-Tag anzuführen. Denn dann gilt sie nur für den äußeren Rahmen der Tabelle, nicht aber für die Rahmen aller Zellen. Sie müssen also jeder Zelle eine Breite zuweisen. Dazu definieren Sie eine CSS-Regel im Kopf der HTML-Datei:

```
<style type="text/css">td, tr, table {border:solid 8px black;}
```

```
</style>
```

Damit sind sämtliche Rahmenlinien acht Pixel dick, schwarz und durchgezogen.

TABELLE FLOTT LADEN

Komplexe Tabellen brauchen viel Zeit, bis sie komplett geladen sind. Das liegt daran, wie Browser Tabellen anzeigen: Sie richten die Größe der Zellen nach dem Inhalt ein. Erst wenn die komplette HTML-Datei und sämtliche Bilder geladen sind, steht das Layout fest. Das ist eigentlich praktisch, weil der Browser die Tabelle an die aktuelle Fenstergröße anpasst – wenn man so lange warten will ...

	Jan	Peter
Januar	37	34
Februar	42	31
...
Dezember	47	56
Donandampfschiffahrtsgesellschaft	Zwei	Drei

	Jan	Peter
Januar	37	34
Februar	42	31
...
Dezember	47	56
Donandampfschiffahrtsgesellschaft	Zwei	Drei

Mit dem fixierten Tabellenlayout setzt der Browser die Spaltenbreite nach der ersten Zeile

CSS 2 bietet eine Option, den Aufbau zu beschleunigen: `table-layout: fixed` im `<table>`-Tag schaltet in den Fixed-Modus für Tabellen um:

```
<table style="table-layout: fixed">
```

Damit übernimmt der Browser die Breite aus der ersten Tabellenzeile in alle weiteren Zellen und kennt somit bereits nach der ersten Zeile der Tabelle die Breite aller

Spalten. Der Preis, den Sie dafür zahlen: Passt ein Bild oder ein langes Wort nicht in die Zelle, schneidet der Internet Explorer den Inhalt am Rand ab und Netscape schiebt den Inhalt in die nächste Zelle. Was Sie dagegen tun, lesen Sie im nächsten Abschnitt „Überhang vermeiden“.

Der Browser ermittelt die Breite der Zellen nach folgendem Schema: Ist in einer Spaltengruppe über `width` eine Breite festgelegt, gilt diese. Dabei ist es unerheblich, ob Sie die Breite mit CSS-Eigenschaften oder HTML-Attributen angeben. Falls nicht, gilt `width` aus der ersten Zelle in der ersten Zeile. Gibt es keine `width`-Angaben, verteilt der Browser die Breite der Tabelle gleichmäßig auf die Spalten.

Geben Sie auf jeden Fall die gesamte Breite der Tabelle in `<table>` an. Der Internet Explorer geht sonst von 100 Prozent Breite aus, Netscape 6 schaltet dagegen zum normalen Tabellenlayout um.

ÜBERHANG VERMEIDEN

Wie gesagt, die vorgeschlagene Lösung enthält eine Fußangel: Ist der Inhalt einer Zelle zu breit, schneidet ihn der Internet Explorer ab und Netscape lässt den Überhang in die nächste Zelle hineinragen.

Dagegen gibt es ein Mittel: Damit auch Netscape den Überhang abschneidet, können Sie mit der CSS-Eigenschaft `overflow: hidden` genau regeln, wie der Browser mit Zellen mit zu viel Inhalt umgeht. Ergänzen Sie die Zellen entsprechend. Um einen Umbruch zu verhindern, verwenden Sie

```
<td style="overflow: hidden">
```

Die Zelle behält so ihre Breite. Natürlich können Sie auch im `<head>`-Bereich eine CSS-Regel definieren, die für alle Tabellenzellen gilt:

```
<style type="text/css">td, th {overflow: hidden;}</style>
```

Damit verhindern Sie auf der ganzen Seite, dass Netscape mit einem Tabellenüberhang das Layout zerstört. Das Ergebnis sieht dann aus wie in dem Bild links zum fixierten Tabellenlayout. ■

Wolfgang Nefzger/ad