

SELFHTML/Navigationshilfen HTML/XHTML Tabellen



## Gestaltung einer Tabelle



- ↓ [Zellenabstand und Zelleninnenabstand](#)
- ↓ [Regeln für den Außenrahmen](#)
- ↓ [Regeln für Gitternetzlinien](#)
- ↓ [Breiten- und Höhenangaben](#)
- ↓ [Zeilenumbruch in Zellen verhindern](#)
- ↓ [Ausrichtung von Zellen](#)
- ↓ [Hintergrundfarben und Hintergrundbilder](#)
- ↓ [Farben für Rand und Gitternetzlinien \(Microsoft\)](#)
- ↓ [Tabellen mit CSS gestalten](#)
- ↓ [Weitere Informationen](#)



## Zellenabstand und Zelleninnenabstand

Sie können den Abstand zwischen den Zeilen und Spalten einer Tabelle in Pixeln bestimmen. Ferner können Sie den Abstand zwischen Zellenrand und Zelleninhalt bestimmen.

### Beispiel:

[Anzeigebispiel: So sieht's aus](#)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
    "http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<title>Zellenabstand und Zelleninnenabstand in Tabellen</title>
</head>
<body>

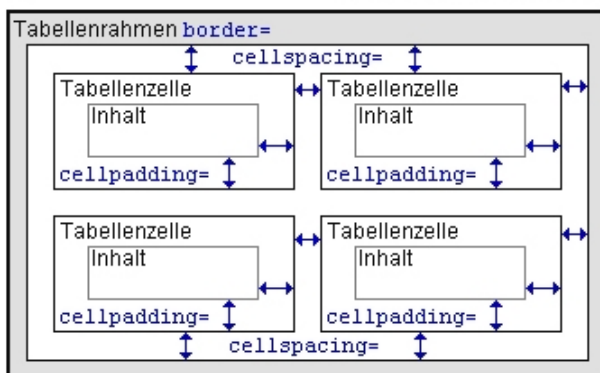
<h1>Weit auseinander</h1>

<table border="8" cellspacing="10" cellpadding="20">
  <tr>
    <th>Berlin</th>
    <th>Hamburg</th>
    <th>M&uuml;nchen</th>
  </tr>
  <tr>
    <td>Milj&ouml;h</td>
    <td>Kiez</td>
    <td>Bierdampf</td>
  </tr>
  <tr>
    <td>Buletten</td>
    <td>Frikadellen</td>
    <td>Fleischpflanzerl</td>
  </tr>
</table>

</body>
</html>
```

### Erläuterung:

Die folgende Grafik zeigt die Wirkungsweise der Attribute im einleitenden `<table>`-Tag:



Mit **cellpadding** bestimmen Sie den Innenabstand einer Zelle, also den Abstand zwischen Zellenrand und Zelleninhalt in Pixeln (*cellpadding* = Zelleninnenabstand). Mit **cellspacing** bestimmen Sie den Abstand der Zellen untereinander in Pixeln (*cellspacing* = Zellenabstand). Wenn Sie mit **border** einen sichtbaren Rahmen und sichtbare Gitternetzlinien in der Tabelle erzeugen, ist die Wirkungsweise von **cellpadding** und **cellspacing** besser nachvollziehbar. Bei blinden Tabellen erzeugen beide Attribute optisch gesehen einfach nur "Abstand".

↑ ↓



## Regeln für den Außenrahmen

Wenn Sie mit **border** einen sichtbaren Tabellenrahmen erzeugen, erhalten automatisch alle Seiten einen Rahmen. Sie können aber auch genau bestimmen, welche Seiten eines Außenrahmens angezeigt werden und welche nicht.

### Beispiel:

#### Anzeigebispiel: So sieht's aus

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
    "http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<title>Regeln für den Außenrahmen bei Tabellen</title>
</head>
<body>

<h1>Rahmenvereinbarung</h1>

<table border="3" frame="void">
  <tr>
    <td><b>Assoziation 1</b></td>
    <td><b>Assoziation 2</b></td>
    <td><b>Assoziation 3</b></td>
  </tr>
  <tr>
    <td>Berlin</td>
    <td>Hamburg</td>
    <td>München</td>
  </tr>
  <tr>
    <td>Miljö</td>
    <td>Kiez</td>
    <td>Bierdampf</td>
  </tr>
  <tr>
    <td>Buletten</td>
    <td>Frikadellen</td>
    <td>Fleischpflanzerl</td>
  </tr>
</table>

</body>
</html>
```

### Erläuterung:

Voraussetzung für all diese Angaben ist das Attribut `border` im einleitenden `<table>`-Tag. Dadurch wird ein Außenrahmen angezeigt.

Mit dem Attribut `frame` können Sie dann bestimmen, an welchen Seiten der Tabellenrahmen gezogen werden soll (*frame = Rahmen*).

Mit `frame="box"` (*box = Rechteck*) erreichen Sie, dass der Tabellenrahmen oben, links, rechts und unten sichtbar dargestellt wird (die Angabe ist identisch mit dem, was die Angabe `border` bewirkt - auch `frame="border"` ist erlaubt und bewirkt das Gleiche). Folgende andere Angaben sind möglich:

Mit `frame="void"` (*void = nichts*) wird überhaupt kein Tabellenrahmen angezeigt. Wenn Sie `border` angeben, werden jedoch die Gitternetzlinien der Tabelle sichtbar angezeigt. Die Tabelle sieht dann also aus wie ein an allen Seiten offenes Gitter (das obige Beispiel benutzt diese Variante).

Mit `frame="above"` (*above = oberhalb*) wird nur am oberen Rand der Tabelle eine Rahmenlinie angezeigt.

Mit `frame="below"` (*below = unterhalb*) wird nur am unteren Rand der Tabelle eine Rahmenlinie angezeigt.

Mit `frame="hsides"` (*hsides = horizontal sides = horizontale Seiten*) wird nur am oberen und am unteren Rand der Tabelle eine Rahmenlinie angezeigt.

Mit `frame="vsides"` (*vsides = vertical sides = vertikale Seiten*) wird nur am linken und am rechten Rand der Tabelle eine Rahmenlinie angezeigt.

Mit `frame="lhs"` (*lhs = left hand side = links*) wird nur am linken Rand der Tabelle eine Rahmenlinie angezeigt.

Mit `frame="rhs"` (*rhs = right hand side = rechts*) wird nur am rechten Rand der Tabelle eine Rahmenlinie angezeigt.

### Beachten Sie:

Netscape 4.x und Konqueror interpretieren diese Angaben noch nicht.



## Regeln für Gitternetzlinien

Sie können Regeln aufstellen, welche Gitternetzlinien einer Tabelle angezeigt werden sollen und welche nicht.

### Beispiel:

**Anzeigebispiel: So sieht's aus**

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
    "http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<title>Regeln für Gitternetzlinien bei Tabellen</title>
</head>
<body>

<h1>Gruppendynamik</h1>

<table border="1" rules="groups">
  <thead>
    <tr>
      <th>Assoziation 1</th>
      <th>Assoziation 2</th>
      <th>Assoziation 3</th>
    </tr>
  </thead>
  <tfoot>
    <tr>
      <td><i>betroffen:<br>4 Mio. Menschen</i></td>
      <td><i>betroffen:<br>2 Mio. Menschen</i></td>
      <td><i>betroffen:<br>1 Mio. Menschen</i></td>
    </tr>
  </tfoot>
  <tbody>
    <tr>
      <td>Berlin</td>
      <td>Hamburg</td>
      <td>München</td>
    </tr>
    <tr>
      <td>Miljöhöh</td>
      <td>Kiez</td>
      <td>Bierdampf</td>
    </tr>
  </tbody>
</table>
```

```

</tr>
<tr>
  <td>Buletten</td>
  <td>Frikadellen</td>
  <td>Fleischpflanzerl</td>
</tr>
</tbody>
</table>

</body>
</html>

```

### Erläuterung:

Voraussetzung für all diese Angaben ist das Attribut `border` im einleitenden `<table>`-Tag. Dadurch werden Gitternetzlinien angezeigt.

Mit `rules` können Sie im einleitenden `<table>`-Tag Regeln für die Gitternetzlinien bestimmen (*rules = Richtschnüre, hier: Gitternetzlinien*).

Folgende Angaben sind dabei möglich:

Mit `rules="none"` (*none = keine*) werden überhaupt keine Linien gezogen, der Außenrahmen der Tabelle wird jedoch angezeigt.

Mit `rules="rows"` (*rows = Reihen*) werden Linien zwischen allen Tabellenzellen gezogen, nicht jedoch zwischen den Spalten der Tabelle.

Mit `rules="cols"` (*cols = columns = Spalten*) werden Linien zwischen allen Tabellenspalten gezogen, nicht jedoch zwischen den Zeilen der Tabelle.

Mit `rules="groups"` (*groups = Gruppen*) werden Linien zwischen Kopf, Körper und Fuß einer Tabelle gezogen (siehe [☰ Kopf, Körper und Fuß einer Tabelle definieren](#) - das obige Beispiel verwendet diese Angabe).

Mit `rules="all"` (*all = alle*) werden Linien zwischen allen Tabellenzellen gezogen (Voreinstellung).

### Beachten Sie:

Netscape 4.x, Konqueror und Safari Interpretieren diese Angaben noch nicht.





## Breiten- und Höhenangaben

Sie können für die gesamte Tabelle, sowie für einzelne Spalten und Zeilen geeignete Angaben zu Breite und Höhe notieren. Dadurch schaffen Sie "Raum".

Breitenangaben werden durch das Attribut `width` notiert, Höhenangaben durch das Attribut `height` (*width = Breite, height = Höhe*). Bei allen Angaben sind absolute Zahlenangaben erlaubt, die Breite bzw. Höhe als Anzahl Pixel bestimmen, sowie prozentuale Angaben, die am Ende ein Prozentzeichen (%) haben.

Die einzige von allen Angaben, die nach der HTML-Variante Strict erlaubt ist, ist die Gesamtbreite der Tabelle. Andere Angaben sind als *deprecated* gekennzeichnet und sollen künftig aus dem HTML-Standard entfallen. Solche Angaben sind auch mit Hilfe von Stylesheets möglich. Und schließlich gibt es auch noch Angaben, die zwar von einigen Browsern interpretiert werden, aber nicht mal zum HTML-Standard in der Variante Transitional gehören. Die folgende Tabelle listet auf, welche Angaben möglich sind, und welchen Status die Angabe in Bezug auf HTML-Standard und Browser-Kompatibilität hat.

Angabe (Notation)	Status	Bedeutung
<code>&lt;table width="..."&gt;</code>		Bestimmt die Breite der gesamten Tabelle in Bezug auf den verfügbaren Raum im Elternelement. Das Elternelement kann das "Dokument" sein, also das <code>body</code> -Element, oder auch ein anderes Element, innerhalb dessen eine Tabelle vorkommen kann, z.B. ein <code>div</code> -Element oder eine Tabellenzelle einer äußeren Tabelle.
<code>&lt;table height="..."&gt;</code>		Bestimmt die Höhe der gesamten Tabelle in Bezug auf das Anzeigefenster. Diese Angabe wird zwar von den Browsern schon lange unterstützt und wird in der Praxis häufig benutzt,

<code>&lt;th width="..."&gt; oder &lt;td width="..."&gt;</code>		<p>gehörte jedoch nie zum HTML-Standard.</p> <p>Bestimmt die Breite einer Tabellenzelle und damit die Breite aller Zellen der gleichen Spalte. Als <i>deprecated</i> eingestuft. Ersetzbar durch CSS-Notationen wie <code>&lt;th style="width:100px"&gt;</code>. Solche Angaben werden allerdings nur von neueren Browsern interpretiert.</p>
<code>&lt;th height=&gt; oder &lt;td height=&gt;</code>		<p>Bestimmt die Höhe einer Tabellenzelle und damit die Höhe aller Zellen der gleichen Tabellenzelle. Als <i>deprecated</i> eingestuft. Ersetzbar durch CSS-Notationen wie <code>&lt;td style="height:100px"&gt;</code>. Solche Angaben werden allerdings nur von neueren Browsern interpretiert.</p>

Einige Browser Interpretieren darüber hinaus weitere Angaben wie `<tr height=>`. Solche Angaben sind allesamt nicht HTML-Standard-konform und sollten nicht mehr verwendet werden.

### Beispiel:

#### Anzeigebispiel: So sieht's aus

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>Breiten- und Höhenangaben bei Tabellen</title>
</head>
<body>

<h1>Breit und hoch mit HTML</h1>

<table border="1" width="60%">
  <tr>
    <td width="50%" height="100">Berlin</td>
    <td>Hamburg</td>
    <td>München</td>
  </tr>
  <tr>
    <td>Miljah</td>
    <td>Kiez</td>
    <td>Bierdampf</td>
  </tr>
  <tr>
    <td>Buletten</td>
    <td>Frikadellen</td>
    <td>Fleischpflanzerl</td>
  </tr>
</table>

<h1>Breit und hoch mit CSS</h1>

<table border="1" style="width:60%">
  <tr>
    <td style="width:50%; height:100px">Berlin</td>
    <td>Hamburg</td>
    <td>München</td>
  </tr>
  <tr>
    <td>Miljah</td>
    <td>Kiez</td>
  </tr>

```



```

    <td>Bierdampf</td>
  </tr>
  <tr>
    <td>Buletten</td>
    <td>Frikadellen</td>
    <td>Fleischpflanzerl</td>
  </tr>
</table>

</body>
</html>

```

### Erläuterung:

Im Beispiel wird zweimal die gleiche Tabelle notiert - im ersten Fall kommen die HTML-Attribute `width` und `height` an erlaubten Stellen zum Einsatz - nach der HTML-Variante Transitional. Die zweite Tabelle benutzt entsprechende CSS-Eigenschaften zur Angabe von Breite und Höhe und genügt damit auch den Anforderungen an die HTML-Variante Strict. Interpretiert wird die zweite Tabelle allerdings erst vom Internet Explorer ab Version 4.x und von Netscape ab Version 6.

### Beachten Sie:

Die Angaben zu Breite und Höhe sind nur wirksam, wenn der Tabellen- bzw. Zelleninhalt kleiner ist, sodass leerer Raum entsteht. Wenn der Tabelleninhalt größer ist, werden die Angaben zu Breite und Höhe außer Kraft gesetzt. Die Tabelle bzw. Zelle (und mit ihr Spalte und Zeile) werden dann so dimensioniert, dass der gesamte Inhalt angezeigt wird. Einige Browser - z.B. Netscape bis einschließlich Version 4.x - setzen Angaben zu Breite und Höhe allerdings leider je nach Tabelleninhalt stärker außer Kraft als es nötig wäre.

Da die Angabe zu `width` in einer Tabellenzelle spaltenweit gilt, brauchen Sie sie nur einmal pro Spalte zu notieren. Am sinnvollsten ist es, die Angabe in der ersten Zeile der Tabelle zu notieren. Ebenso ist es mit der Angabe zu `height` in einer Tabellenzelle. Da diese Angabe für die ganze zugehörige Tabellenzelle gilt, ist es am sinnvollsten, sie in der ersten Zeile der Zeile zu notieren. Im obigen Beispiel wird durch `<td width="50%" height="100">` in der ersten Zeile also sowohl die Breite der ersten Spalte als auch die Höhe der ersten Zeile definiert.

Verwenden Sie Pixelangaben und Prozentangaben immer so, dass keine Konflikte entstehen. Im obigen Beispiel werden etwa alle Breiten prozentual bestimmt, und nur die Höhe wird mit Pixeln festgelegt. So kann der Browser im Beispiel für die gesamte Tabelle eine Breite von 60% Prozent zum Elmentelement ermitteln und für die erste Spalte wiederum 50% davon. Die Angabe von 100 Pixeln Zeilenhöhe kommt damit nicht in Konflikt.

Eine übersichtlichere Lösung, die Breite von Tabellenspalten zu definieren, steht mit der Möglichkeit zur Verfügung, [Spalten vorzudefinieren](#). Die hier beschriebene Möglichkeit hat dagegen den Vorteil, auch von älteren Browsern interpretiert zu werden.

↑ ↓



## Zeilenbruch in Zellen verhindern

Sie können festlegen, dass beim Inhalt einer Zelle kein automatischer Zeilenbruch erfolgen darf. Die Spalte der Tabelle wird bei der Anzeige entsprechend breit dimensioniert. Das entsprechende Attribut ist jedoch als *deprecated* eingestuft und soll künftig aus dem HTML-Standard entfallen.

### Beispiel:

[Anzeigebelsolel: So sieht's aus](#)

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>Zeilenbruch in Tabellenzellen verhindern</title>
</head>
<body>

<h1>Lange N&auml;chte</h1>

<table border="1">
  <tr>
    <td nowrap>
      Die langen HTML-N&auml;chte von Berlin sind die l&auml;ngsten N&auml;chte &uuml;berhaupt.
      Die langen CSS-N&auml;chte von Berlin sind ebenfalls die l&auml;ngsten N&auml;chte und sogar

```

```

    noch l&auml;nger als die HTML-N&auml;chte.
  </td>
</tr>
</table>

</body>
</html>

```

### Erläuterung:

Mit dem Attribut `nowrap` (ohne weitere Wertzuweisung) in einem einleitenden `<th>`- oder `<td>`-Tag verhindern Sie, dass der Text innerhalb der Zelle automatisch umbrochen wird. Die Zelle und damit ihre gesamte zugehörige Spalte wird so breit wie erforderlich.

Die entsprechende Eigenschaft in CSS lautet `white-space`. Diese wird jedoch erst ab Internet Explorer 5.5 und Netscape 6 interpretiert.

### Beachten Sie:

Wenn Sie **XHTML-Standard-konform** arbeiten, müssen Sie das `nowrap`-Attribut in der Form `nowrap="nowrap"` notieren. Weitere Informationen dazu im Kapitel [XHTML und HTML](#).



## Ausrichtung von Zellen

Zelleninhalte von Tabellen können aufgrund unterschiedlicher Inhalte oder durch Angaben zu Breite und Höhe mehr Raum in Anspruch nehmen, als es ihr Inhalt erfordert. Für diesen Fall können Sie Zelleninhalte sowohl horizontal (linksbündig, zentriert, rechtsbündig) als auch vertikal (obenbündig, mittig, untenbündig) am Zellenrand ausrichten. Die entsprechenden Angaben sind sowohl auf einzelne Zellen (also `th`- und `td`-Elemente) anwendbar, als auch auf Tabellenbereiche wie Tabellenzellen (`tr`), Tabellenkopf (`thead`), Tabellenkörper (`tbody`) und Tabellenfuß (`tfoot`). In diesem Fall gelten die Angaben für alle Zellen des jeweiligen Bereichs.

### Beispiel:

#### Anzeigebispiel: So sieht's aus

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>Ausrichtung von Zelleninhalten</title>
</head>
<body>

<h1>Richtungsweisend</h1>

<table border="1">
  <tr>
    <th align="left"   valign="top" width="200" height="75">links oben</th>
    <th align="center" valign="top" width="200">zentriert oben</th>
    <th align="right"  valign="top" width="200">rechts oben</th>
  </tr>
  <tr>
    <td align="left"   valign="middle" height="75">links mittig</td>
    <td align="center" valign="middle">zentriert mittig</td>
    <td align="right"  valign="middle">rechts mittig </td>
  </tr>
  <tr>
    <td align="left"   valign="bottom" height="75">links unten</td>
    <td align="center" valign="bottom">zentriert unten</td>
    <td align="right"  valign="bottom">rechts unten</td>
  </tr>
  <tr align="center" valign="middle">
    <td height="75">alles ...</td>
    <td>... zentriert ...</td>
    <td>... mittig</td>
  </tr>
</table>

```

```

<h2>Auch am Dezimalzeichen?</h2>

<table border="1">
  <colgroup>
    <col>
      <col align="char" char=",">
    </colgroup>
  <tr>
    <th>Element</th>
    <th>Anteil in mg</th>
  </tr>
  <tr>
    <td>Nitrat</td>
    <td>0,117126</td>
  </tr>
  <tr>
    <td>Hydrogencarbonat</td>
    <td>330,0</td>
  </tr>
</table>

</body>
</html>

```

### Erläuterung:

Mit `align="left"` können Sie eine Kopfzelle in ihrem einleitenden `<th>`-Tag linksbündig ausrichten, mit `align="right"` rechtsbündig. Datenzellen können Sie in ihrem einleitenden `<td>`-Tag mit `align="center"` zentriert ausrichten und durch `align="right"` rechtsbündig. Die Angaben `align="center"` bei Kopfzellen und `align="left"` bei Datenzellen sind natürlich auch erlaubt, bewirken aber lediglich die Voreinstellung und sind deshalb nicht nötig (*align = Ausrichtung, left = links, center = zentriert, right = rechts*). Diese Angaben können Sie auch in einleitenden `<tr>`-Tags, einleitenden `<thead>`-Tags, einleitenden `<tbody>`-Tags und `<tfoot>`-Tags notieren. Dann werden alle Zellen im entsprechenden Bereich so ausgerichtet wie angegeben. Widersprechende Angaben in einzelnen Zellen haben allerdings Vorrang vor bereichsweiten Angaben. Ab HTML 4.0 ist auch die Angabe `align="justify"` erlaubt. Dadurch erzwingen Sie einen Blocksatz innerhalb der Zelle.

Mit `valign="top"` können Sie eine Zelle in ihrem einleitenden Tag obenbündig ausrichten, durch die Angabe `valign="bottom"` untenbündig. Auch die Angabe `valign="middle"` (nicht "center"!) ist möglich. Dies ist jedoch die Voreinstellung und deshalb nicht unbedingt nötig (*valign = vertical align = vertikale Ausrichtung, top = oben, bottom = unten, middle = mittig*). Auch das `valign`-Attribut können Sie in einleitenden `<tr>`-Tags, einleitenden `<thead>`-Tags, einleitenden `<tbody>`-Tags und `<tfoot>`-Tags notieren. Ab HTML 4.0 ist auch die Angabe `valign="baseline"` erlaubt. Dann werden alle Zellen innerhalb einer Zeile, die diese Angaben erhalten, an einer gemeinsamen Basislinie ausgerichtet. Und zwar so, dass die erste Textzelle aller dieser Zellen immer auf gleicher Höhe beginnt.

Ferner ist seit HTML 4.0 die Angabe `align="char"` in Verbindung mit den Attributen `char` und `charoff` erlaubt. Mit `align="char"` können Sie erzwingen, dass die Spalteninhalte an einem Dezimalzeichen ausgerichtet werden. Außerdem benötigen Sie noch das Attribut `char` (*char = character = Zeichen*). Diesem weisen Sie als Wert ein Zeichen zu, das Sie als Dezimalzeichen verwenden. Die Ausrichtung erfolgt dann so, dass das Dezimalzeichen immer an der gleichen Stelle steht. Neben dem Attribut `char` zur Bestimmung des Dezimalzeichens können Sie zusätzlich noch das Attribut `charoff` angeben (*charoff = character offset = Zeichenposition*). Damit können Sie angeben, an welcher Position das Zeichen frühestens vorkommen kann. Im obigen Beispiel wird ein Komma als Dezimalzeichen definiert. Die Werte der betreffenden - im Beispiel [☰ vordefinierten Spalten](#) - werden dann so ausgerichtet, dass das Dezimalzeichen immer an der gleichen Stelle steht. Die HTML-Spezifikation gibt jedoch an, dass die Unterstützung der Dezimalzeichenausrichtung durch Browser freiwillig ist. Und tatsächlich interpretiert auch keiner der weit verbreiteten Browser diese Angaben.

↑ ↓



## Hintergrundfarben und Hintergrundbilder

Sie können sowohl für eine gesamte Tabelle als auch für einzelne Zeilen oder Zellen eine Hintergrundfarbe definieren. Die verbreiteten Web-Browser unterstützen auch die Angabe von Hintergrundbildern mit Wallpaper-Effekt, doch diese Angaben gehörten nie zum HTML-Standard. Auch die Angaben zur Hintergrundfarbe sind als *deprecated* gekennzeichnet und sollen künftig aus dem HTML-Standard entfallen. Sowohl das Definieren von Hintergrundfarben als auch das von Hintergrundbildern ist mit Stylesheets möglich.

### Beispiel:

**Anzeigebeispiel: So sieht's aus**



```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>Hintergrundfarben und Hintergrundbilder</title>
</head>
<body text="#000099">

<h1>Gro&szlig;e Worte mit HTML</h1>

<table border="1" bgcolor="#FFFF00">
  <tr>
    <td width="200" height="100"><h2>Traum</h2></td>
    <td width="200" bgcolor="#00FFFF"><h2>Zeit</h2></td>
    <td width="200"><h2>Luft</h2></td>
  </tr>
  <tr bgcolor="#FF00FF">
    <td height="100"><h2>Licht</h2></td>
    <td><h2>Himmel</h2></td>
    <td><h2>Leben</h2></td>
  </tr>
</table>

<h1>Gro&szlig;e Worte mit CSS</h1>

<table border="1" style="background-color:#FFFF00">
  <tr>
    <td width="200" height="100"><h2>Traum</h2></td>
    <td width="200" style="background-color:#00FFFF"><h2>Zeit</h2></td>
    <td width="200"><h2>Luft</h2></td>
  </tr>
  <tr style="background-color:#FF00FF">
    <td height="100"><h2>Licht</h2></td>
    <td><h2>Himmel</h2></td>
    <td style="background-image:url(background.jpg)"><h2>Leben</h2></td>
  </tr>
</table>

</body>
</html>

```

## Erläuterung:

Durch das Attribut `bgcolor` im einleitenden Tag der Tabelle können Sie eine Hintergrundfarbe für die gesamte Tabelle bestimmen (`bgcolor = background-color = Hintergrundfarbe`). Beim Angeben der Farbe gelten die Regeln zum [Definieren von Farben in HTML](#). Ebenso können Sie das Attribut `bgcolor` im einleitenden Tag einer Tabellenzeile (`<tr>`) notieren, um die Hintergrundfarbe für alle Zellen in dieser Zeile zu bestimmen. Wenn Sie `bgcolor` im einleitenden Tag einer Datenzeile (`<td>`) oder einer Kopfzeile (`<th>`) angeben, gilt die Hintergrundfarbe für diese eine Zelle. Im Konfliktfall hat die Farbangabe in einzelnen Zellen Vorrang vor der Angabe für ganze Zeilen oder Tabellen. Die Angabe für eine Zeile hat im Konfliktfall Vorrang vor der Angabe für die ganze Tabelle.

Das Gleiche ist mit dem Attribut `background` möglich, um eine Hintergrundgrafik für die Tabelle oder einzelne Zellen einzubinden. Dieses Attribut ist jedoch im Zusammenhang mit Tabellen nicht HTML-Standard-konform und kommt im obigen Beispiel auch nicht vor. Die zweite im Beispiel notierte Tabelle zeigt, wie es HTML-Standard-konform gemacht wird. Dagegen spricht eigentlich nur noch die Existenz von Netscape 4.x, der keine CSS-Angaben bei Tabellen interpretiert.

Im obigen Beispiel wird angenommen, dass sich die Grafik `background.jpg` im gleichen Verzeichnis befindet. Sie können jedoch auch Grafiken aus anderen Verzeichnissen (z.B. `/daten/grafik/hintergrund/background.gif` oder `../gif/back.gif`) oder von entfernten URIs (z.B. `http://www.mein-anderer-server.de/grafik/back.jpg`) angeben.

## Beachten Sie:

Wenn Sie eine Hintergrundfarbe definieren oder eine Hintergrundgrafik einbinden, sollten Sie für den Text eine geeignete Kontrastfarbe definieren.



## Farben für Rand und Gitternetzlinien (Microsoft)

Sie können auch für Rahmen und Gitternetzlinien Farbangaben machen. Diese Angaben sind jedoch Microsoft-Syntax und gehören

