

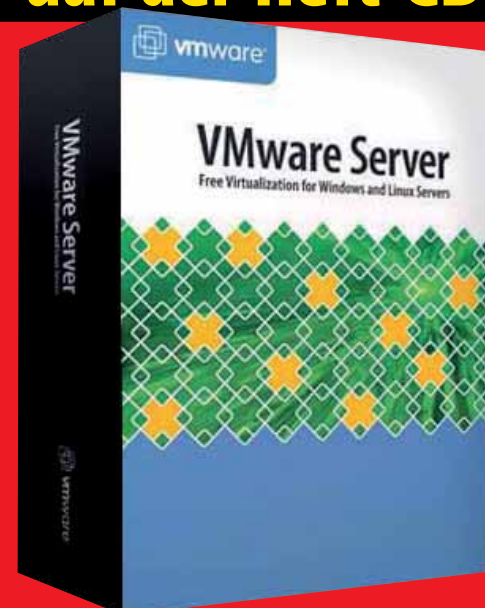
INTERNET

PROFESSIONELL

XML
& CSS

So formatieren Sie XML-Daten direkt mit CSS

>S.40

mit
CDVOLLVERSIONEN
auf der Heft-CD

VMware-Server*

Die ideale Testumgebung
für Webdesign & BrowserOpen
Clipart
LibraryÜber 6900 Cliparts
für Buttons & LogosXAMPP
1.5Neueste Versionen
für Linux & Windows

PLUS

- ▶ East-Tec Eraser 8 Lite
- ▶ Datei Commander 8.2 LE
- ▶ Zone Alarm 7 free

*kostenlose Online-Registrierung erforderlich

PHP
Version 6

>S.22

- ▶ **Vorsicht: Nicht alles ist abwärtskompatibel**
- ▶ **Abgeschafft: safe_mode & register_globals**

CSS-Tricks:
Fehlersuche

>S.58

So kommen Sie CSS-Problemen auf die Spur

Barrierefrei

So gestalten Sie Websites
benutzerfreundlich für alle >S.48Google
AdSense

>S.52

So verdienen Sie mit Ihrer
Website ein paar Euro extra

TIPPS

- ▶ **Formulare mit Joomla 1.5** S.80
- ▶ **Datenbanken jenseits von MySQL** S.66
- ▶ **Open-Source-Shops** S.32

Die Heft-CD enthält
Info- bzw. Lernprogramme
gemäß § 14 JuSchG

4 194078 704505



Neues in PHP 6

Nach der Einführung der Objekt-orientierung in PHP 5 sollen in Version 6 nun neue Operatoren aufgenommen und die Skript-sprache von Ballast bereinigt werden. [Seite 22](#)

Webshop zum Nulltarif

Open-Source-Shops sind flexibel, beliebig erweiterbar und kosten nicht die Welt. Der Artikel stellt die drei wichtigsten Vertreter gegenüber. [Seite 32](#)



■ Logon

- 6 **Netwatch**
Die neuesten Trends & Produkte
- 16 **Kalter Krieg um MP3s**
Reiner Gärtner über den russischen Musikdienst Allofmp3
- 17 **Doppelt vergoogelt**
Johannes Selbach über Ranking-Probleme durch doppelte Inhalte

■ Titelstory

- 22 **Ausblick auf PHP 6**
In Sachen Unicode, Safemode und Operatoren tut sich in der neuen Version von PHP einiges. Das Special gibt einen ausführlichen Überblick über die hinzugekommenen und weggefallenen Elemente der Skriptsprache

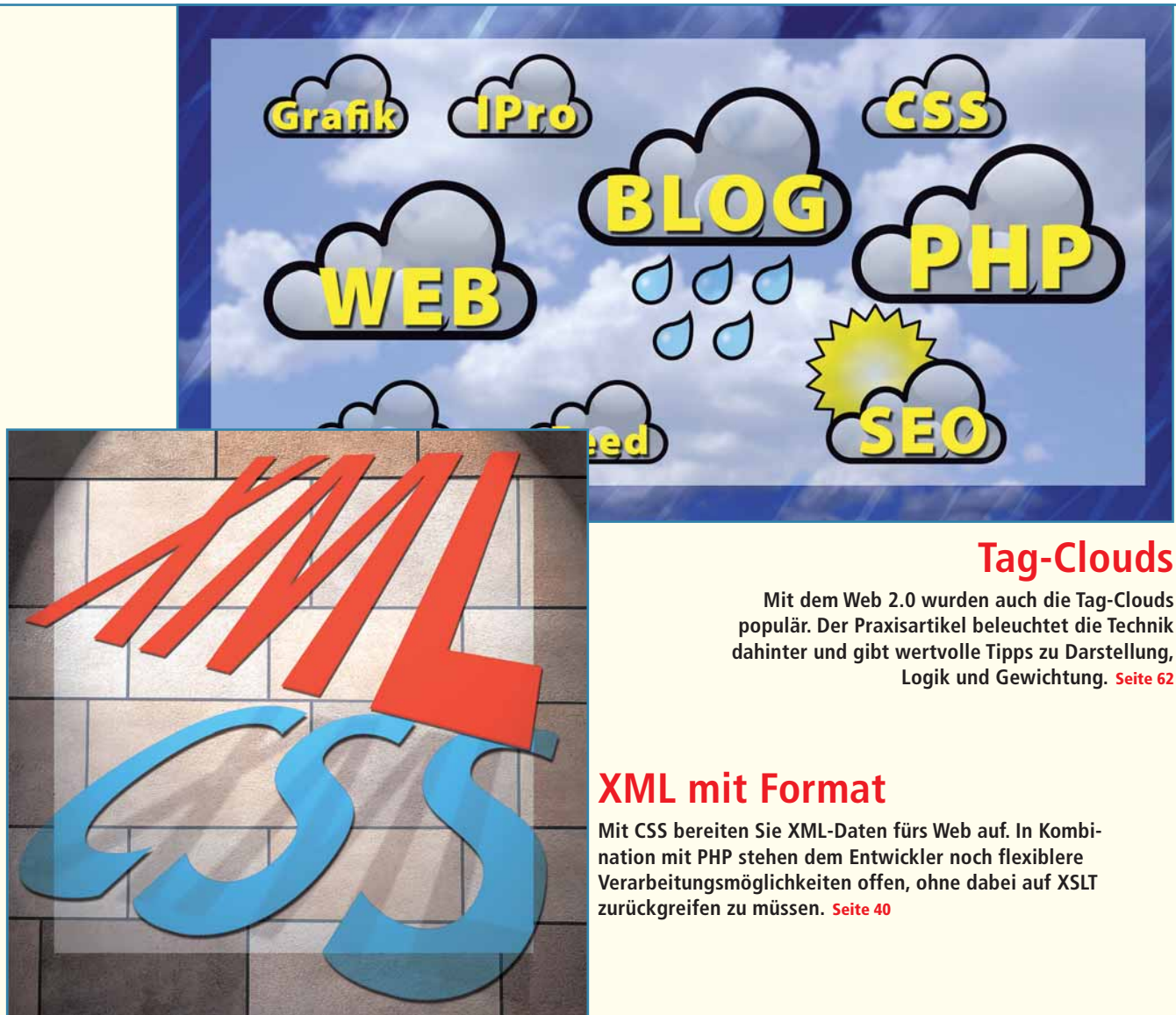
■ Technik & Sicherheit

- 28 **Einzeltests**
Adobe Photoshop CS 3 Beta • Mirabyte Feedwriter • SMC Skype Phone • PC-Kaufmann 2007 for eBay • Parallels Workstation 2.2 • D-Link Horstbox
- 32 **Shopping mit Open Source**
OS Commerce, XT-Commerce und Virtuemart sind Paradebeispiele für kostenlose Shop-Alternativen. Der Artikel stellt die Systeme im Detail vor und streicht Unterschiede zu kommerziellen Produkten heraus
- 36 **CMS-Entscheidungshilfe**
Wer sich für ein Content-Management-System entscheidet, muss sich im Vorfeld genau über die Erfordernisse klar werden und auch Alternativen im Auge behalten



■ Webdesign

- 40 **XML-Anzeige mit CSS**
Sitemaps und andere XML-Daten stellen Sie mit CSS in besucherfreundlicher Form im Web dar
- 44 **Effekte im Hintergrund**
Ein Trend auf modernen Webseiten ist die Hintergrundtapete. Der Workshop zeigt, wie Sie clevere Kacheffekte anlegen
- 48 **Barrierefrei in fünf Schritten**
Anhand eines Beispiel-Layouts verrät der Workshop, wie Sie ein barrierefreies Design von Grund auf realisieren
- 52 **Tricks zu Google AdSense**
Wer im Web Geld verdienen will, kommt um Google AdSense kaum herum. Hier erfahren Sie die besten Tricks für höhere Einkünfte
- 56 **Bilder statt Worte**
Eine Imagemap bietet dem Besucher eine hervorragende visuelle Navigationshilfe



Tag-Clouds

Mit dem Web 2.0 wurden auch die Tag-Clouds populär. Der Praxisartikel beleuchtet die Technik dahinter und gibt wertvolle Tipps zu Darstellung, Logik und Gewichtung. [Seite 62](#)

XML mit Format

Mit CSS bereiten Sie XML-Daten fürs Web auf. In Kombination mit PHP stehen dem Entwickler noch flexiblere Verarbeitungsmöglichkeiten offen, ohne dabei auf XSLT zurückgreifen zu müssen. [Seite 40](#)

- 58 CSS fehlerfrei**
Um valides CSS zu produzieren und Fehler aufzuspüren, sind einige Tricks und Tools von Nutzen

Web Dev

- 62 Tag-Clouds im Detail**
Verschiedene Tag-Cloud-Varianten werden vorgestellt und Praxistipps helfen beim Generieren eigener Wortwolken
- 66 Datenmanagement mit PHP**
Ganz ohne MySQL lässt sich in PHP eine effektive Datenhaltung realisieren
- 70 Websitebaker: Skript-Fütterung**
Das benutzerfreundliche CMS bietet eine geniale Funktion, um eigenen PHP-Code zu integrieren
- 73 Javascript-Tipps**
Optimieren Sie Ihre Website mit den besten Tricks rund um Javascript

- 74 Ajax-Serie: Dialogfenster**
Modale Dialoge fokussieren die Aufmerksamkeit des Users auf das Hinweisfenster
- 77 Mobiles Webdesign mit ASP.NET**
Am Beispiel einer Kinokartenreservierung führt Sie dieser Artikel in die Programmierung mit den flexiblen Mobile Controls ein
- 80 Profi-Formulare in Joomla**
Mit der Erweiterung Facile Forms entwickeln Sie schnell anspruchsvolle Webformulare
- 84 Pear: Schnelles PHP**
Mit Caching machen Sie Ihre PHP-Skripts schneller
- 90 Tipps & Tricks**
Tipps & Tricks rund um das Thema Skriptsprachen und Webdesign



Standards

- 3 Editorial
- 94 Internet-Dienstleister-Verzeichnis
- 96 Impressum/Inserenten-Verzeichnis
- 97 Leserbriefe
- 98 Vorschau/Logoff



18 Heft-CD

- Vollversionen: VMware Server, East-Tec Eraser Lite, Datei Commander
- Preview: PHP 6
- Specials: Web-Entwicklungs-Software, Open-Source-Shops, Ajax-Tools



DIE NEUE PC PROFESSIONELL ab März 2007



Neugierig? Bloggen Sie mit unter www.pcpro-2007.de!

Schritt für Schritt zum *div*-Layout

Grundgerüst


Barrierefreiheit im Internet ist in aller Munde. Barrierefreies Webdesign basiert auf bewusstem und zielgerichtetem Einsatz der HTML-Syntax sowie der klaren Trennung zwischen Inhalt und Layout per XHTML und CSS. **Von Stephan Heller**

Info

Auf einen Blick

»Der Artikel richtet sich an Webdesigner, die in die barrierefreie Entwicklung einsteigen möchten, und jene, die HTML-Quelltexte ohne irgendwelche Tools schreiben möchten.

Das brauchen Sie

- »Text- oder HTML/CSS-Editor
- »Listings auf Heft-CD 

» Die Grundgerüste von barrierefreien Webseiten sind alle sehr ähnlich. Der Artikel zeigt, mit wie wenig HTML Sie auskommen, um den Rahmen für eine gut zugängliche Seite zu schaffen. Alle Schritte des Workshops finden Sie auch auf der Heft-CD im Bereich *Listings*.

Gute Planung ist die halbe Arbeit: Legen Sie die Struktur Ihrer Seite fest, bevor Sie überlegen, wie Sie Ihr Layout umsetzen. Nehmen Sie sich dazu ein Beispiel an Büchern. Diese haben immer den gleichen Aufbau: ein Cover, ein Inhaltsverzeichnis, den eigentlichen Lesestoff und am Ende noch etwas Informatives. Diese Reihenfolge sollten Sie auf Ihrer Webseite ebenfalls einhalten:

- Kopf
 - Logo
 - Titel/Slogan
- Navigation
- Inhalt
 - Überschriften
 - Texte
 - Bilder
- Fuß

Es ist immer wieder zu sehen, dass die Navigation irgendwo in der Mitte oder am Ende der Seite zu finden ist. Unter SEO-Gesichtspunkten ist das natürlich richtig, denn hier will man eher den Content so weit vorn wie möglich haben, anstatt auf jeder Seite immer wieder die Navigation. Aus Accessibility-Sicht ist das allerdings, als wäre das Inhaltsverzeichnis mitten im Buch. Finden Sie hier einen für sich vertretbaren Kompromiss aus beiden Ansprüchen.

»Verwendung von XHTML 1.0«

Gute Webseiten sind in XHTML 1.0 geschrieben. Vereinfacht ausgedrückt ist das die Weiterentwicklung von HTML 4 unter Berücksichtigung der Kriterien von XML. XHTML ist zukunftssicher und bietet die größte Gewissheit, dass Ihre Website auf unterschiedlichen Ausgabegeräten, darunter auch Handys und PDAs, dargestellt werden kann.

Ein paar Grundregeln machen die Umsetzung von XHTML einfach: In HTML 4 sind *hr* und *br* gültige, so genannte leere Tags. In XHTML ist für jedes Tag ein Abschluss-Tag notwendig. Die richtigen Schreibweisen sind hier `<hr/></hr>` beziehungsweise `
</br>`. Als Kurzschreibweise sind hier auch `<hr/>` und `
` erlaubt, der Slash am Ende des Tags ersetzt das schließende Tag. Tags müssen alle klein geschrieben sein, Attribute gehören in doppelte Anführungszeichen: `<div color="red">`. Leere Attribute wie *checked* bekommen ihren eigenen Namen als Parameter: `checked="checked"`. Details finden Sie in der Referenz SelfHTML (de.selfhtml.org).

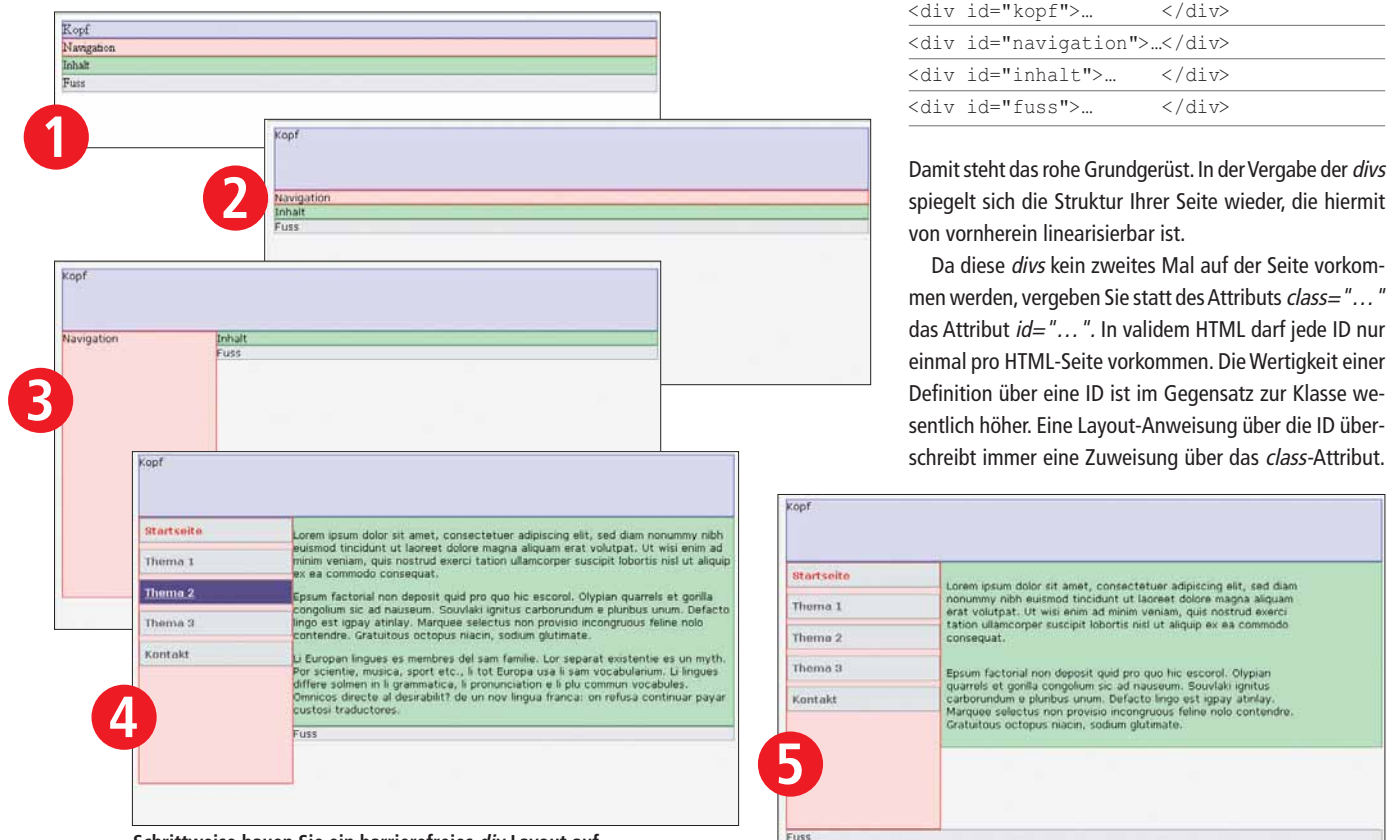
»Nur vier *divs* sind notwendig«

Für jeden Bereich der Webseite vergeben Sie ein *div*:

```
<div id="kopf">... </div>
<div id="navigation">...</div>
<div id="inhalt">... </div>
<div id="fuss">... </div>
```

Damit steht das rohe Grundgerüst. In der Vergabe der *divs* spiegelt sich die Struktur Ihrer Seite wieder, die hiermit von vornherein linearisierbar ist.

Da diese *divs* kein zweites Mal auf der Seite vorkommen werden, vergeben Sie statt des Attributs *class*= "... " das Attribut *id*= "... ". In validem HTML darf jede ID nur einmal pro HTML-Seite vorkommen. Die Wertigkeit einer Definition über eine ID ist im Gegensatz zur Klasse wesentlich höher. Eine Layout-Anweisung über die ID überschreibt immer eine Zuweisung über das *class*-Attribut.



Schrittweise bauen Sie ein barrierefreies *div*-Layout auf.

```

</head>
<body>

  <div id="kopf">Kopf</div>
  <div id="navigation">Navigation</div>
  <div id="inhalt">Inhalt</div>
  <div id="fuss">Fuss</div>

</body>
</html>
    
```

Das Grundgerüst benötigt lediglich vier *divs*.

»Einbinden des CSS«

HTML bietet drei Möglichkeiten, um Cascading Stylesheets einzubinden: in den HTML-Tags selbst über das Attribut *style="..."*, im Kopf der HTML-Datei in einem definierten CSS-Block oder ausgelagert in eine eigenständige CSS-Datei. Alle Wege sind erlaubt. Als beste Variante ist die Letztgenannte anzusehen, denn:

- Sie schaffen von Anfang an eine ganz klare Trennung zwischen Layout und Inhalt.
- Sie benötigen nur eine CSS-Datei und können damit beliebig viele HTML-Seiten gestalten, ob drei, zehn oder 5000.
- Immer gleich bleibendes Design definieren Sie nur einmal – und ändern es auch nur an einer Stelle, wenn Sie Ihr komplettes Layout überarbeiten.
- Die CSS-Datei wird einmal in den Browser-Cache geladen, der zweite Seitenaufruf spart sich das erneute Laden. Das spart Zeit. Für Besucher der Webseite, die sich über ein Modem einwählen, ist das spürbar.

Im Head-Bereich Ihrer HTML-Datei binden Sie das CSS ein:

```

<link media="screen, projection"
href="css/layout.css" type="text/css"
rel="stylesheet"/>
    
```

Mit diesem Link binden Sie das CSS ein, das für die Darstellung im Browser, für Beamer und so weiter zuständig ist. In der weiteren Entwicklung können Sie spezielle CSS-Dateien beispielsweise für den Drucker einbinden. Dazu passen Sie das *media*-Attribut im Link an: *media="print"*.

»Praxistipp«

Um genau zu sehen, was Sie im CSS eingeben und was davon im HTML ankommt, definieren Sie im CSS für jedes *div* einen Rahmen und eine Hintergrundfarbe. Damit be-

- Startseite
- Thema 1
- Thema 2
- Thema 3
- Kontakt

Per CSS bauen Sie aus der Navigationsliste eine schaltflächenähnliche Navigation auf.

| |
|------------|
| Startseite |
| Thema 1 |
| Thema 2 |
| Thema 3 |
| Kontakt |

| |
|------------|
| Startseite |
| Thema 1 |
| Thema 2 |
| Thema 3 |
| Kontakt |

kommen Sie schnell ein Gefühl dafür, wie sich die *divs* verhalten. Sobald Sie beginnen, die Positionierungen zu definieren, ist dies eine sehr einfache und effektive Technik. Am Ende der Entwicklung kommentieren Sie diese Hilfslinien einfach wieder aus.

Falls Sie noch wenig Übung mit CSS haben, sollten Sie nach jeder Änderung im CSS die Datei speichern und sich die Auswirkungen im Browser ansehen. Viele CSS-Eigenschaften beeinflussen sich gegenseitig, so dass man nach mehreren Änderungen nicht mehr eindeutig sagen kann, welche CSS-Definition welchen Effekt hatte. Machen Sie zum Beginn kleine Schritte.

Über den Shortcut [Strg S] ist das Speichern der Seite und über die Funktionstaste [F5] ist der Reload der Seite schnell durchgeführt.

»Generelle Angaben im CSS«

Da Sie nun beginnen, die Positionierung des Layouts umzusetzen, sollten Sie zunächst ein paar Einträge im CSS vornehmen. Ein wichtiges Kriterium der Barrierefreiheit ist die Skalierbarkeit der Schrift und der ganzen Webseite. Das bedeutet, dass der Nutzer die Schriftgröße nach eigenen Bedürfnissen im Browser einstellen kann. Diese Möglichkeit besteht allerdings nur, wenn im CSS relative Schriftgrößenangaben gemacht wurden. Das sind *%*- oder *em*-Angaben. Vergessen Sie *pt* und *px* zur Angabe von Schriftgrößen, aber auch für Tabellenbreiten, Formularfelder und Ähnliches. Auch hier sollten Sie relative Angaben verwenden.

Definieren Sie also zuerst die Standard-Schriftgröße für den Body. Auf diese Angabe beziehen sich später alle anderen Größenangaben. Deswegen ist es sinnvoll, diese Angabe als Erstes vorzunehmen.

```

font-size : 80%;
font-family : verdana,arial;
    
```

Falls Sie die Standardeinstellung des Browsers übernehmen möchten, setzen Sie diese auf *100.01%*. Damit weichen Sie Browserbugs von Internet Explorer, Opera 6 und Safari aus.

Viele empfinden diese Schriftgröße jedoch als zu groß, deswegen kann diese angepasst werden. Wichtig bei Ihrer Einstellung ist, dass alle Texte auch dann noch zu lesen sind, wenn Sie die Ansicht im Browser um zwei Stufen nach unten umstellen. Eine Schriftgröße von 80 Prozent hat sich in der Praxis etabliert.

BITV

Gesetzliche Grundlage

Die Kriterien der Barrierefreiheit sind in der BITV (Barrierefreie Informationstechnik-Verordnung) festgelegt. In 14 Punkten und 66 Unterpunkten werden alle Anforderungen an barrierefreie Webseiten beschrieben. Diese sind in zwei unterschiedliche Prioritäten unterteilt.

Öffentliche Anbieter wie zum Beispiel Städte, Ämter oder Verwaltungen sind seit Ende 2005 gesetzlich verpflichtet, ihre Web-Auftritte barrierefrei zu gestalten.

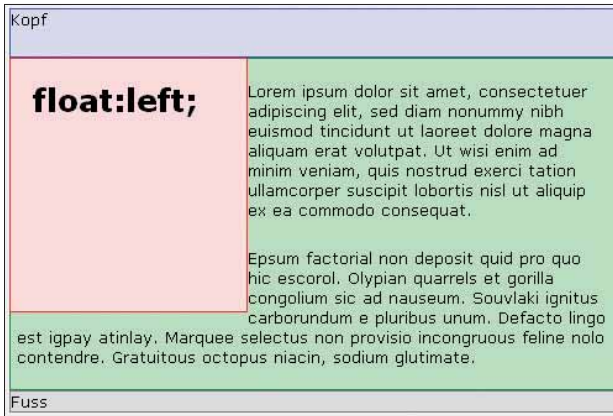
► bundesrecht.juris.de/bitv

Verlosung

Mitmachen & gewinnen



Internet Professionell verlost drei Exemplare des Buches »Barrierefreie Webseiten« aus dem Verlag Addison-Wesley im Wert von je 39,95 Euro. Dieses Buch beweist: Barrierefreiheit und attraktives Webdesign schließen sich nicht aus. Nach einer Beschreibung der Grundlagen – Behinderungen, mit denen Webdesigner bei ihren Anwendern rechnen müssen, technische Lösungen wie Braille-Zeile, rechtliche Vorgaben et cetera – demonstrieren die Autoren an einem konkreten Praxisprojekt, wie eine Website barrierefrei gemacht wird. Hierbei beschreiben sie ausführlich die Analyse- und Konzeptionsphase, Design-Kriterien, die technische Umsetzung, Nutzer-Tests sowie die vor und nach dem Relaunch involvierten Prozesse. Ein Kapitel zur barrierefreien Gestaltung einer CMS-basierten Website (Joomla, Typo 3) rundet das Buch ab. Um an der Verlosung teilzunehmen, füllen Sie bitte den kurzen Fragebogen unter www.ipro-leser.de aus.



Die CSS-Eigenschaft float lässt ein *div* in die angegebene Richtung rutschen.

Über die Firefox-Tidy-Erweiterung haben Sie die Gültigkeit des XHTMLs immer im Auge. Rechts unten wird das Ergebnis als Icon angezeigt.



Da Sie gerade zentrale Einstellungen vornehmen, ist es sinnvoll, auch den Schrifttyp zu definieren. Verdana ist dabei der Schrifttyp, der am besten zu lesen ist.

Die zweite Angabe Arial wird von allen Browsern interpretiert, denen die Verdana-Schrift nicht zur Verfügung steht. Von verschnörkelten Schriften sollten Sie keinen Gebrauch machen. Diese reduzieren die Lesbarkeit deutlich.

»Kopfarbeit«

Im nächsten Schritt definieren Sie die Höhe des Kopf-Bereichs. Falls Sie keine dynamischen Elemente im Kopf haben und definitiv wissen, dass die Höhe nicht variiert, können Sie diese exakt festlegen:

```
#kopf {
height : 6em;
}
```

Falls der Inhalt des Kopfs mehr Platz als Ihre definierte Höhe hat, sollten Sie statt *height* die CSS-Eigenschaft *min-height* verwenden. Hiermit nimmt sich der Kopf automatisch den Platz, den er braucht. Allerdings unterstützt der Internet Explorer erst ab der Version 7 diese Eigenschaft. Aus diesem Grund

Linear

Neue Sichtweisen einnehmen

Vergessen Sie oben, unten, links und rechts. Es gibt nur noch Anfang und Ende und eine durchdachte Struktur auf dieser Strecke. Stellen Sie sich vor, alle Informationen Ihrer Webseite sind an einem Faden hintereinander aufgefädelt. Das bedeutet, dass Sie Informationen nicht mehr zeitgleich sehen, sondern nacheinander bekommen. Das ist die Beschreibung von Linearisierbarkeit und ist eine zentrale Anforderung und die Basis für eine gut zugängliche Webseite.

müssen Sie für den IE dann eine spezielle Definition einbinden, damit dieser ebenfalls die gewünschte Höhe richtig darstellt. Bis zur Version 6 des IE interpretiert der Browser die Eigenschaft *height* wie *min-height*.

```
#kopf {
min-height : 6em;
}
* html #kopf {
height : 6em;
}
```

Mit der Schreibweise ** html #kopf*, sprich dem Sternchen und dem HTML vor der ID, sprechen Sie konkret nur den IE 6 an. Spezielle Definitionen nur für den IE 7 schreiben Sie zusätzlich mit einem *+* hinter dem Sternchen: **+ html #kopf*.

»Rückgrat: Navigation«

Mit der Definition der Navigation nimmt das Grundgerüst Formen an. Sie definieren die Breite und vor allem die Fließeigenschaften des *divs*. Die CSS-Eigenschaft *float* bewirkt in der Kombination mit der Breitenangabe, dass das *div* nur noch die definierte Breite einnimmt. Außerdem bewegt sich das *div* in die definierte Richtung, das weitere HTML umfließt dieses *div*.

Diese Fließeigenschaften gehören wahrscheinlich zu den größten Schwierigkeiten von CSS. Deswegen hier noch einmal: Ein *div* mit der Eigenschaft *float:left* rutscht nach links, das weitere HTML umfließt das *div* rechts.

```
#navigation {
width : 15em;
float : left;
}
```

Gleichzeitig ist im Beispiel eine Mindesthöhe für die Navigation definiert – für den IE 6 brauchen Sie ebenfalls den oben beschriebenen Browser-Hack. *float* ist mitunter ein Problem, da sich das

Verhalten auf den kompletten nachfolgenden HTML-Code auswirkt. Die Lösung des Problems lesen Sie am Fuß des Grundgerüsts.

»Navigation als Liste«

Die BITV fordert nicht ausdrücklich die Umsetzung von Navigationsblöcken als Listen, sondern formuliert sehr allgemein. Sie fordert die Gruppierung von zusammengehörigen Links.

Der Einsatz einer ungeordneten Liste *ul* ist aber semantisch richtig, da eine Navigation in der Tat eine Auflistung ist. In der Praxis hat sich diese Technik bewährt.

Die aktuell ausgewählte Seite sollte nicht verlinkt sein. Ein Link zur Seite, auf der man sich befindet, ist überflüssig und kann den Benutzer irritieren. Sobald Sie sich auf einer zweiten Navigationsebene befinden, muss jedoch der Weg zurück in die erste Ebene wieder verlinkt sein.

```
<ul>
<li><span class="auswahl">
Startseite</span></li>
<li><a href="#">Thema 1</a></li>
<li><a href="#">Thema 2</a></li>
</ul>
```

Hinter jedem Link der Navigation fügen Sie einen Punkt (oder ein anderes druckbares Zeichen) ein. Dieser dient einer klaren Trennung der Links beim Gebrauch von Screenreadern. Die Punkte blenden Sie per CSS wieder aus, da sie für die Darstellung im Browser nicht notwendig sind.

Zur Verbesserung der Usability können Sie für den Screenreader-Benutzer in das *li* der ausgewählten Seite den versteckten Text »Aktuelle Seite:« hinzufügen.

```
<li><span class="unsichtbar">Aktuelle Seite: </span><span class="auswahl"> Startseite</span></li>
<li><a href="#">Thema 1</a><span class="unsichtbar"> . </span></li>
```

»CSS für die Navigationsliste«

Über die CSS-Eigenschaft `list-style-type:none` blenden Sie die Listenpunkte aus, die per Default vom HTML angezeigt werden. Über verschiedene Hintergrund- und Rahmenfarben erzielen Sie einen Button-ähnlichen Effekt. Über den Außenabstand nach unten per `margin-bottom:0.6em`; schaffen Sie Abstände in der neu gestalteten Navigation.

```
#navigation ul li{
list-style-type : none;
border : 1px solid gray;
background-color : #DEDEDE;
margin-bottom : 0.6em;
}
```

Wenn Sie sich die Navigation nun in Firefox und im Internet Explorer ansehen, werden Sie vielleicht erschrecken, da die Positionierung wie Kraut und Rüben aussieht. Browser stellen Abstände und Linien anhand des so genannten Boxmodells dar. Hier sehen Sie, dass verschiedene Browser das Boxmodell unterschiedlich darstellen – zum Leid der CSS-Designer. Dies ist mit ein paar Kniffen zu lösen: Per `margin` und `padding` setzen Sie die Innen- und Außenabstände auf 0 und sorgen dafür, dass die Liste das komplette `div` für die Navigation ausfüllt.

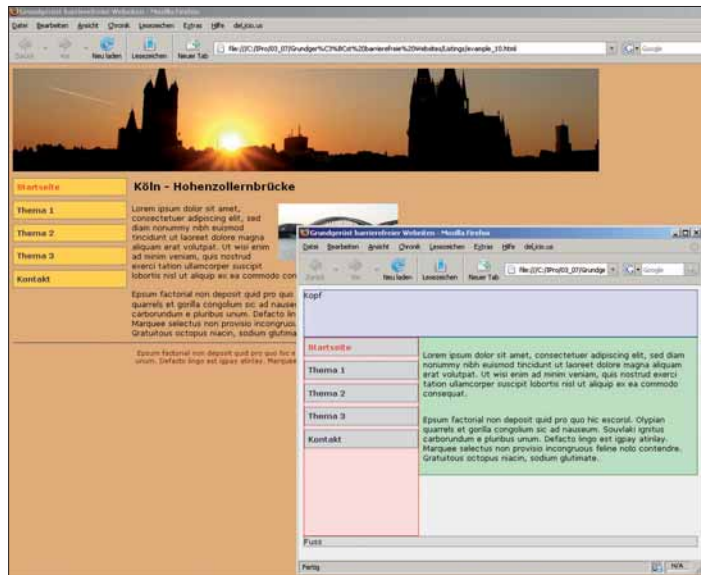
```
#navigation ul{
margin : 0;
padding : 0;
}
```

»CSS für die Navigations-Links

Links sind so zu gestalten, dass sie sich vom anderen Text abheben und sich beim Überfahren mit der Maus beziehungsweise beim Tabben durch die Seite verändern – zum Beispiel durch einen Wechsel der Schrift- oder Hintergrundfarbe oder das Ein- oder Ausblenden des Unterstrichs. Wichtig ist, dass Sie eine Mehrfachauszeichnung wählen, also zum Beispiel Farbwechsel und Wechsel des Unterstrichs. Auf diese Weise erreichen Sie, dass auch bei abgestellten Farben oder für Menschen, die keine Farben wahrnehmen können, Links deutlich zu erfassen sind.

Um das Aussehen zu verändern, sollten Sie nicht nur `hover`, sondern ebenfalls `focus` setzen. Damit erreichen Sie die Veränderung sowohl für das Mouseover als auch für das Tabben durch die Seite. Das dritte Attribut `active` sollten Sie ebenfalls definieren, da der IE dieses zur Darstellung beim Tabben benötigt. Die Reihenfolge `hover`, `focus`, `active` hat sich in der Praxis als brauchbar erwiesen.

```
#navigation ul li a {
padding : 0.5em;
display : block;
color : #383742;
```



Sobald das grundsätzliche Layout steht, können Sie Farben anpassen und Bilder einfügen.

```
text-decoration : none;
}
#navigation ul li a:hover,
#navigation ul li a:focus,
#navigation ul li a:active {
Color : #fff;
font-weight : 700;
text-decoration : underline;
background-color : #302C68;
}
```

Über `padding` erreichen Sie den Effekt einer Schaltfläche, über `display:block` schaffen Sie es zusätzlich, dass nicht nur der eigentliche Text verlinkt ist, sondern die Klickfläche auf den kompletten Listenpunkt ausgedehnt wird. Das erhöht die Usability, da Sie den Bereich zum Klicken vergrößern. Außerdem hat dies dann Einfluss auf das Mouseover-Verhalten des Links. Das Verhalten bezieht sich dann nicht nur auf den Link-Text, sondern auch auf den ganzen Listenpunkt.

»Unkomplizierter Inhalt«

Für den Inhalt gibt es beim Grundgerüst nur wenige Aspekte zu beachten. Es ist empfehlenswert, die Textbreite zu begrenzen. Texte, die über die komplette Monitorbreite laufen, sind mitunter schwerer zu lesen. Damit der Text die Navigation nicht umfließt, setzen Sie den Abstand des Inhalts nach links um die Breite der Navigation: `margin-left:15em`.

Für den Inhaltsbereich sind weitere Anforderungen der Barrierefreiheit zu berücksichtigen, zum Beispiel die richtige Beschreibung der `alt`-Attribute der `image`-Tags, Sprachwechsel (gleichzeitiger Gebrauch von Deutsch, Englisch und anderen Sprachen auf einer Seite), Auszeichnung von Abkürzungen und Akronymen, Farbwahl, Kontraste et cetera. Die Umsetzung umfasst sowohl technische als auch redaktionelle Anforderungen. Diese sind hier innerhalb dieses Artikels mit der Zielrichtung Grundgerüst nur angedeutet.

»Fuß«

Aufgrund des Floatens der Navigation werden Sie beobachtet haben, dass nicht nur der Inhalt, sondern auch der Fuß verrutscht ist. Hierfür hat CSS die Möglichkeit, das `float` aufzulösen. Per `clear:both` beenden Sie alle gesetzten Fließeigenschaften. Das erste HTML-Element, das folgt, beginnt im Browser in einer neuen Zeile, alle weiteren HTML-Elemente verhalten sich in ihren Fließeigenschaften so, wie die Default-Einstellungen der Browser es vorgeben. Prinzipiell sind auch die Variationen `clear:left` oder `clear:right` möglich, um ein bestimmtes Verhalten aufzulösen, jedoch ist dies in der Praxis eine kaum verwendete Anweisung.

```
#fuss {
clear : both;
}
```

Sie können nun den Rahmen und die Hintergrundfarben auskommentieren. Im Beispiel `example_10.html` sind zusätzlich Farben angepasst (`layout_10.css`) und Bilder eingefügt. Dort können Sie sehen, wie das Grundgerüst für Ihre barrierefreie Seite aussieht. [jp]

Links

Barrieren abbauen

HTML-Referenz zum Nachschlagen

► de.selfhtml.org

BITV – die gesetzliche Grundlage

► bundesrecht.juris.de/bitv

Umfangreiche Informationen zu barrierefreiem Webdesign

► www.barrierefreies-webdesign.de

Validator des W3C zur Überprüfung für gültiges HTML

► validator.w3.org

Erweiterung für Firefox, validiert Ihre Seite bei jedem Aufruf

► html-validator.de

INTERNET IM ABO

PROFESSIONELL

AKTUELLE NEWS ZU
INTERNET UND WEBDESIGN

Jeden Monat in Ihrer
Internet Professionell:

Heiße Tipps zu
CSS, PHP, Javascript,
Ajax und Design-Trends



Gratis:



Als Dankeschön*
erhalten Sie zusätzlich den
Flexi-Drive 128 MB USB 2.0
Hi-Speed-Stick

* Solange Vorrat reicht

Jetzt
einsteigen lohnt sich:

- Sie erhalten die nächsten drei Ausgaben der Internet Professionell zum Vorzugspreis von nur 2,90 € pro Heft
- Sie erhalten einen 128 MB USB-Stick gratis dazu

Sie sparen 35% und
einen USB-Stick gibt es
als Geschenk dazu!

Gleich anfordern und Geschenk sichern:

www.internet-pro.de/abo/11