

Zur HTML5-CSS-Kurzreferenz

Die Kurzreferenz soll beim Lernen von HTML5 und CSS unterstützen und als Spickzettel für HTML-Befehle und CSS Anweisungen dienen. Es wurden hier nur bewusst aktuelle Befehle (Doctype HTML5) aufgenommen.

Die Kurzreferenz ist Bestandteil des HTML und CSS-Tutorial des <https://www.html-seminar.de>

Wenn Sie diese Referenz für eigene Kurse verwenden wollen, sehen Sie auf der letzten Seite nach.

Aufbau HTML-Befehl

Jeder HTML-Befehl hat die Form `<html>` - der Befehl (engl. TAG) am Ende `</html>`

Die Befehle sind Abkürzungen zum englischen Begriff und werden kleingeschrieben.

Schachteln von HTML-Befehlen

Bei der Reihenfolge der HTML-Befehle muss darauf geachtet werden, dass diese in umgekehrter Reihenfolge wieder geschlossen werden. `<befehl1> <befehl2> Inhalt </befehl2> </befehl1>`

siehe: <https://www.html-seminar.de/schachteln-von-html-tags.htm>

Absätze

`<p>` `</p>` Paragraf – bildet einen Absatz

`
` Harter Umbruch – Text geht in nächster Zeile weiter (mit HTML5 auch `
` in Ordnung)

Überschriften

`<h1>` `</h1>` Überschrift, die Zahl gibt den Rang an (1 am wichtigsten).
Es gibt `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>` und `<h6>`

Hervorhebungen im Text

`` `` fett hervorgehoben (physikalisch) (HTML5: stilistisch hervorgehoben)

`` `` fett hervorgehoben (logisch) (HTML5: Text ist wichtig)

`<i>` `</i>` kursiv (schräg gestellt) (HTML5: z.B. Gedanken, veränderte Stimmlage)

`` `` betonen von Text (oft kursiv dargestellt)

`^{` `}` hochgestellter Text

`_{` `}` tiefgestellter Text

`<u>` `</u>` unterstrichen - veraltet (engl. deprecated)! Zum Test von Validierungstools.

Trennlinie

`<hr>` Trennlinie ziehen (HTML5: thematischer Umbruch)

Aufzählungen und Listen

Auflistung mit Spiegelstrichen

`` Start einer Aufzählung (durch ul mit Spiegelstrichen)

`` `` Aufzählungspunkt

`` `` nächster Aufzählungspunkt

`` Ende einer Aufzählung

Nummerierte Auflistung

<code></code>	im Gegensatz zu <code></code> Aufzählungen mit Nummerierung
<code></code> <code></code>	Aufzählungspunkt
<code></code> <code></code>	nächster Aufzählungspunkt
<code></code>	Ende einer Aufzählung

Definitionsliste

Wird gerne für Glossare verwendet. Für die Definitionsliste werden 3 HTML-TAGs benötigt. Beispiel unter <https://www.html-seminar.de/html-definitionsliste-dl-dt-dd.htm>

<code><dl></code>	
<code><dt>Apache</dt></code>	
<code><dd>ist ein weit verbreiteter Webserver, der sich ...</dd></code>	
<code><dt>Browser</dt></code>	
<code><dd>nennen sich Programme, die ...</dd></code>	
<code></dl></code>	

logische Textauszeichnungen

<code><address></code> <code></address></code>	Um eine Adresse zu kennzeichnen
<code><code></code> <code></code></code>	Quellcode darstellen
<code><samp></code> <code></samp></code>	Beispiel darstellen
<code><kbd></code> <code></kbd></code>	Tastatureingabe darstellen
<code><var></code> <code></var></code>	Variable darstellen
<code><tt></code> <code></tt></code>	dicktengleich Schrift (alle Buchstaben nutzen dieselbe Breite) in Fernschreiberstil (Teletyper) (HTML5: gibt es nicht mehr)
<code><pre></code> <code></pre></code>	präformatierter Textabschnitt (mit dicktengleich Schrift) (Block-Element)

Definitionen, Abkürzungen und Zitate

<code><dfn></code> <code></dfn></code>	Definition
<code><abbr></code> <code></abbr></code>	Abkürzung
<code><acronym></code> <code></acronym></code>	besondere Abkürzungen (z.B. Lkw) (HTML5: gibt es nicht mehr - <code><abbr></code> nutzen)
<code><q></code> <code></q></code>	Zitat
<code><cite></code> <code></cite></code>	Quelle oder Autor v. Zitat
<code><blockquote></code> <code></blockquote></code>	Zitat (Block-Element)

Änderungsmarkierungen

`` `` Inhalt ist veraltet und nicht mehr gültig

`<ins>` `</ins>` Inhalt ist auf einer Seite neu dazugekommen

physische Textauszeichnungen

`<small>` `</small>` Text wird kleiner dargestellt (HTML5: Rechtshinweise, „Kleingedrucktes“)

`<big>` `</big>` Text wird größer dargestellt (HTML5: HTML-Befehl gibt es nicht mehr! Veraltet!)

Abschnitt für CSS-Definitionen

`<style>` `</style>` CSS-Definitionsbereich in HTML-Datei (siehe Kapitel CSS)

Bereiche festlegen ohne vordefinierte logische Struktur

`<div>` `</div>` Bereich festlegen (Blockelement)

`` `` Bereich innerhalb Zeile festlegen (Inline-Element)

Klasse/ID einfügen für Verknüpfung mit CSS

Die Attribute **class** und **id** können bei jedem HTML-Befehl genutzt werden. Sehr oft werden diese gerne bei `<div>` und `` genutzt. Die eigentliche Stärke von Klassen und IDs kommt erst durch CSS zum Vorschein.

`` Notwendig, damit man über CSS das Design für den gewählten Bereich beeinflussen kann!

`` genauso wie class, aber der ID-Name darf pro HTML-Einzelseite nur einmal vorkommen

Verlinkung

`Ausgabe auf Bildschirm für Besucher`

Interne Links: Link nach „zielseite“ – intern dann nur der Dateiname der HTML-Seite
z.B. „hobbys.htm“, „index.htm“ etc.

Bsp.: `Meine Hobbys`

Externer Link mit voller URL, z.B. `http://www.html-seminar.de/` (der letzte / gehört dazu!)

Bsp.: `HTML-Tutorial`

Externer Link in neuem Fenster mit voller URL, z.B. `http://www.html-seminar.de/`
und zusätzlich mit `target="_blank"`, damit neue Seite aufgeht

Bsp.: `HTML-Tutorial`

Link auf Bild: anstatt einer HTML-Seite wird ein Bild angegeben.

Bsp.: `Anzeige Portrait`

Link auf PDF: anstatt einer HTML-Seite wird ein PDF angezeigt bzw. zum Download angeboten
(je nach Einstellung des Browsers/PDF-Programms).

Bsp.: `HTML-CSS-Übersicht als PDF`

Beispiel für PDF als externe Datei:

`PDF HTML-R.`

Mail als Link: anstatt einer HTML-Seite wird ein Mail-Adresse mit **mailto:** angegeben.

(sollte wegen Spam vermieden werden!)

Bsp.: `Mail an Mustermann`

Sprungmarken Links: Im Dokument wird ein Anker gesetzt (wohin gesprungen wird) in der
folgenden Form: `Bereich 2`

Und den Link: `springen zu Bereich 2`

(Sprung auf Bereich kann auch auf andere Seite erfolgen, dann vor # die URL angeben.)

Grafikdatei einbinden (Foto, Grafik, etc.)

``

im Attribut **src** wird der Dateiname angegeben und wo das Bild auf dem Server liegt.

Breite und Höhe sollten angegeben werden

alt sollten den Inhalt des Bildes in Textform wiedergeben

siehe: <https://www.html-seminar.de/grafiken.htm>

Farben

`#ff39a0`

Die Angabe von Farbnamen kann direkt mit den entsprechenden englischen Begriffen
erfolgen oder Hexadezimal in der Form `#rrggbb` (rot grün blau)

siehe: <https://www.html-seminar.de/farben.htm>

Dokumentieren/Ausdokumentieren in HTML

`<!-- ... -->`

Um Kommentare im Quellcode einzubauen, werden diese zwischen diese Zeichenfolge
gesetzt. Dadurch werden diese im Browser nicht mehr angezeigt

(so können zum Testen auch Teile des Quellcodes „deaktiviert“ werden)

Tabellen

<code><table border="1"></code>	Beginn einer Tabelle (nur zum Testen border="1" setzen)
<code><thead></code>	Bereich für den Kopf beginnt
<code><tr></code>	hier fängt die erste Reihe für die Kopfzeile an
<code><th>Überschrift</th></code>	hier fängt die erste Zelle für die Kopfzeile (Überschriften für Spalte)
<code><th>Ü. Spalte 2</th></code>	zweite Überschrift für zweite Spalte
<code></tr></code>	erste Reihe für Überschriften beenden
<code></thead></code>	Bereich für Kopf zu Ende
<code><tbody></code>	Bereich für Inhalt beginnt
<code><tr></code>	hier fängt die erste Reihe mit Inhalt an
<code><td> Zelle 11 </td></code>	hier fängt die erste Zelle an, dann der eigentliche Inhalt und die Zelle beenden
<code><td> Zelle 12 </td></code>	hier fängt die zweite Zelle an, dann der eigentliche Inhalt und die Zelle beenden
<code></tr></code>	erste Reihe mit Inhalt beenden
<code><tr></code>	nächste Reihe mit Inhalt
<code><td> Zelle 21 </td></code>	erste Zelle, zweite Reihe
<code><td> Zelle 22 </td></code>	zweite Zelle, zweite Reihe
<code></tr></code>	zweite Reihe mit Inhalt beenden
	Hier können noch beliebig viele weiteren Reihen folgen
<code></tbody></code>	Bereich für Inhalt zu Ende
<code><tfoot></code>	Bereich für die Fußzeile (kommt in HTML5 nach dem Inhalt (war bei HTML4 anders))
<code><tr></code>	Beginn der Reihe für Fußzeile
<code><td> Fuß Z1 </td></code>	Zell-Anfang, Inhalt Fußzeile 1, Zell-Ende
<code><td> Fuß Z2 </td></code>	Zell-Anfang, Inhalt Fußzeile 2, Zell-Ende
<code></tr></code>	Ender der Reihe für die Fußzeile
<code></tfoot></code>	Bereich für Fuß zu Ende
<code></table></code>	Tabelle beenden

Wichtig ist: unbedingt auf die korrekte Schachtelung achten!

siehe: <https://www.html-seminar.de/html-tabellen.htm>

Obiger HTML-Code ausgeführt:

Überschrift	Ü. Spalte 2
Zelle 11	Zelle 12
Zelle 21	Zelle 22
Fuß Z1	Fuß Z2

Struktur Seite

DTD

DTD-Doctype-Definition – Definition für Browser

Seitenaufbau

siehe: <https://www.html-seminar.de/html-grundlagen.htm>

Grundgerüst: Grundsätzlicher Aufbau einer HTML5-Seite

```
<!DOCTYPE html>
<html lang="de">

<head>
  <meta charset="UTF-8">
  <title>Titel der Seite</title>
  <meta name="description" content="Beschreibung des Inhalts (160 Zeichen)">
  <link href="design.css" rel="stylesheet">
</head>

<body>

<h1>Eigentlicher Inhalt</h1>

</body>
</html>
```

Websites für mobile Geräte (Handy, Tablet & Co.)

Angabe Viewport <head>-Bereich: <https://www.html-seminar.de/viewport-richtig-setzen.htm>

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
```

Weitere Informationen dazu unter

Media Queries




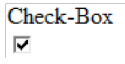
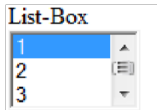
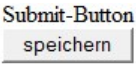
<https://www.html-seminar.de/media-queries-responsive-design.htm>

```
@media screen {
  /* CSS-Anweisungen für die Ausgabe auf Bildschirmen */
}
@media print {
  /* CSS-Anweisungen für die Druckerausgabe */
}
```

für unterschiedliches Design je nach Bildschirmbreite:

```
@media only screen and ( min-width: 950px ) {
  /* breites Browserfenster */
  body { background-color: yellow;}
}
@media only screen and ( max-width: 440px ) {
  /* mobile Geräte */
  body { background-color: green;}
}
@media only screen and ( min-width: 440px ) and ( max-width: 950px ) {
  /* Tablets und mittlere Bildschirme */
  body { background-color: red;}
}
```

Formulare

<pre><form action="" method=""></pre> <p><!-- Hier kommen nun die gewünschten Formularelemente --></p> <pre></form></pre>		<p>Zwischen diesen TAGs werden die einzelnen Formularfelder eingeschlossen</p> <p>action="" -> entweder Seite, die aufgerufen werden soll und deren der Inhalt der Felder übergeben wird, oder Mailadresse in Form "mailto:donald@duck.ent" (nicht ratsam wegen SPAM!) method="post get" – entweder post oder get</p>
<pre><input type="text" name="" value="" size="" maxlength="" ></pre>		<p>Text-Box name="" -> Name des Feldes (für Variablenübergabe) value="" -> Vorgabewert (normalerweise leer) size="" -> Größe der Anzeige maxlength="" -> maximale Länge des Feldes</p>
<pre><input type="hidden" name="" value="" size="" maxlength="" ></pre>		<p>verstecktes Feld</p>
<pre><textarea name="" cols="10" rows="60"></textarea></pre>		<p>Text-Area für Eingabe von viel Text cols="" -> wie viele Spalten angezeigt werden rows="" -> Anzeige der Anzahl Reihen</p>
<pre><input type="radio" name="" value="v" checked="checked" ></pre>		<p>Radio-Button Wird gerne verwendet, wenn es um entweder-oder-auswählen geht (z. B. Herr oder Frau) name="" -> Name des Feldes (für Variablenübergabe) pro Auswahl eine input- Zeile mit demselben Namen! value="v" -> Vorgabewert ist hier wichtig, da der Nutzer nur anklicken kann (Bsp: value="Herr")</p>
<pre><input type="checkbox" name="" checked="checked" ></pre>		<p>Check-Box Anklickbox – über checked="checked" kann die Box bereits angeklickt sein</p>
<pre><select name="" size="3" multiple="multiple"> <option value="1" selected="selected">1</option> <option value="2">2</option> <option value="3">3</option> </select></pre>		<p>List-Box Pull-Down Auswahl size="3" -> Größe der Anzeige multiple="multiple" -> ob mehrere Elemente angewählt werden können <option ... -> die einzelnen Elemente selected="selected" -> vorselektiertes</p>
<pre><input type="Submit" name="" value="speichern" ></pre>		<p>Button zum Anklicken und Absenden des Formulars value="speichern" -> Der value wird als Beschriftung der Schaltfläche angezeigt</p>

Nützliche Links

https://www.html-seminar.de	zum HTML5 und CSS lernen
https://www.webmaster-crashkurs.de	weiteres HTML-Tutorial
http://www.javascript-kurs.de	JavaScript-Kurs
http://wiki.selfhtml.org	SELFHTML (sehr ausführlich auch bei alten Befehlen)
https://validator.w3.org	Kontrolle, ob Fehler im HTML sind
https://jigsaw.w3.org/css-validator/	Validieren von CSS
http://brackets.io	HTML-Editor (Freeware)
https://atom.io	
https://winscp.net	FTP-Programm (Freeware) – unterstützt sichere Verbindung
https://www.php-kurs.com	zum PHP lernen
https://www.studium-und-pc.de	Office (Word, Excel, PowerPoint), Suchmaschinen nutzen wissenschaftliches Arbeiten
https://www.foto-kurs.com	Fotografieren lernen (schnell zu den eigenen Fotos für die Website)

Domainnamen

www.nic.de	Kontrolle, ob ein DE-Domainnamen frei ist, bzw. wer Besitzer davon ist
www.nic.com	Kontrolle, ob ein COM oder NET-Domainnamen frei ist

Provider für Domains mit Webspace und E-Mail-Adressen

1und1.de	bekannter großer Anbieter
goneo.de	günstiger Anbieter
united-domains AG	noch ein Anbieter
	aktuelle Angebote mit Vergünstigungen siehe: https://www.html-seminar.de/provider.htm

Website Online stellen per FTP

<https://www.html-seminar.de/online-stellen.htm>

FTP-Programm WinSCP mit Anleitung:

<https://www.php-kurs.com/winscp-ftp-konfigurieren-und-nutzen.htm>

CSS – Design für Websites

Integration in die HTML-Datei entweder durch Bereich im Kopf (gut zum Lernen)

```
<!DOCTYPE html>
<head>
<style>
/*  HIER KOMMEN NUN DIE CSS-ANWEISUNGEN  */
</style>
</head>
<body>
```

Oder als externe Datei (ratsam), die von allen Seiten einer Website genutzt wird – Link wird im Kopfbereich wie folgt integriert:

```
<!DOCTYPE html>
<head>
<link href="design.css" rel="stylesheet">
</head>
<body>
```

In der externen Datei (hier im Beispiel design.css genannt) werden dann alle CSS-Definitionen gebracht Beispiel: Beispiel Inhalt der Datei design.css

```
h1 {
    color: red;
    letter-spacing: 20px;
}
```

Auflistung CSS und Attribute Schrift (CSS)

font-size:	Mögliche Werte: pt, px, em
font-family:	Schriftart, wichtig ist als letzte Angabe entweder sans-serif (Schrift ohne Serifen) oder serif (Schrift mit Serifen) Beispiel: font-family: Arial, Helvetica, sans-serif;
font-weight:	normal bold bolder lighter
font-style:	italic = kursiv (physikalische Auszeichnung) oblique = schräggestellt (logische Auszeichnung) normal = normal
text-indent:	Einrückung der ersten Zeile eines Textblockes um den vorgegebenen Abstand
line-height:	Zeilenhöhe bestimmen, wenn der Text mehrzeilig angezeigt wird. Empfehlenswert sind relative Angaben über die Einheit „em“
letter-spacing:	Abstände zwischen den einzelnen Buchstaben
word-spacing:	Abstände zwischen den einzelnen Wörtern
font-variant:	Schriftvariante: normal , small-caps
text-decoration:	Textgestaltung: underline , overline , line-through , blink
text-transform:	Textart capitalize , uppercase , lowercase , none

Farbe (CSS)

color:	Farbe für Schrift bzw. Vordergrund – Angabe über englische Bezeichnungen oder Hexadezimal #000000
background-color:	Farbe für den Hintergrundbereich

Rahmen (CSS) alle 3 Angaben sind für einen Rahmen notwendig!

border-color:	Rahmenfarbe - soll nur ein Teil des Rahmens gefärbt werden, dann: border-top-color, border-right-color, border-bottomcolor und border-left-color
border-width:	Rahmenstärke in Pixel soll nur ein Bereich geändert werden: border-top-width, border-right-width, border-bottom-width und border-left-width
border-style:	Rahmenart bestimmen folgende Möglichkeiten gibt es: solid = durchgezogen double = doppelt none = kein Rahmen (unsichtbar) hidden = kein Rahmen (unsichtbar) dotted = gepunktet dashed = gestrichelt groove = 3D-Effekt ridge = 3D-Effekt inset = 3D-Effekt outset = 3D-Effekt Soll nur ein Teil des Rahmens definiert werden, dann: border-top-style, border-right-style, border-bottom-style und border-left-style

Ausrichtung von Elementen (CSS)

text-align:	Horizontale Ausrichtung, möglich Werte: left center right justify
vertical-align:	Vertikale Ausrichtung mögliche Werte: top = oben bündig middle = mittig bottom = unten bündig baseline = ausrichten an Basislinie falls vorhanden, ansonsten unten bündig sub = tieferstellen sup = höherstellen text-top = am oberen Schriftrand text-bottom = am unteren Schriftrand

Abstände (CSS)

margin:	Abstand zum nächsten Element: kann auch für jede Seite einzeln angegeben werden: margin-left, margin-right, margin-bottom, margin-top
padding:	Abstand zum Rand: kann auch für jede Seite einzeln angegeben werden: padding-left, padding-right, padding-bottom, padding-top

Weiten, Höhen-Angaben (CSS)

width:	Breite in: px, em, %, auto
height:	Höhe in: px, em, %, auto

Grafik und Bilder (CSS)

background-image: url("hintergrund.jpg");	Hintergrundbild, anstatt hintergrund.jpg den Pfad und den Namen des gewünschten Bildes angeben
background-repeat:	Verhaltensweise des Bilds - ob es nur einmal angezeigt oder wiederholt wird. Mögliche Werte: repeat = wiederholen (Standard) repeat-y = wiederholen nur in Y-Richtung (senkrecht) repeat-x = wiederholen nur in X-Richtung (waagrecht) no-repeat = nicht wiederholen (nur einmal anzeigen)
background-position:	1-Wert für Horizontale, 2-Wert für Vertikale erster Wert: left center right zweiter Wert: top center bottom Der erste Wert bestimmt die horizontale und der zweite Wert die vertikale Position. Es können auch Pixelangaben erfolgen, z. B. background-position: 100px 20px; Die Grafik wird 100px von links und 20px von oben platziert. Das geht auch mit Prozentangaben (auch in gemischter Form): background-position: 55px 50%;
background-attachment:	fixed Hintergrundbild fixieren. Die Bilder bleiben dann an Ort und Stellen, wenn gescrollt wird. Standard ist die Angabe „scroll“

Dokumentieren/Ausdokumentieren in CSS

/* */ Um Kommentare in CSS einzubauen, werden diese zwischen diese Zeichenfolge gesetzt. Dadurch werden diese nicht mehr ausgeführt (so können zum Testen auch Teile der CSS-Anweisungen „deaktiviert“ werden)

Kurzschreibweisen

Für einige CSS-Formate gibt es die Möglichkeit, diese in einer Kurzschreibweise zusammenzufassen. Dadurch wird es übersichtlicher und weniger Code.

Kurzschreibweise in Form von Beispiel

<code>border:1px solid red;</code>	Rahmen: <code>border-width border-style border-color;</code>
<code>background: #ff00ee url(bild.jpg) fixed no-repeat right top;</code>	<code>background-color: #ff00ee;</code> <code>background-image: url(bild.jpg);</code> <code>background-attachment: fixed;</code> <code>background-repeat: no-repeat;</code> <code>background-position: right top;</code>
<code>padding: 35px 30px 20px 15px;</code>	Leserichtung wie bei Uhr (oben, rechts, unten, links) Kurzschreibweise für: <code>padding-top: 35px;</code> <code>padding-right: 30px;</code> <code>padding-bottom: 20px;</code> <code>padding-left: 15px;</code>
<code>padding: 35px 18px;</code>	Wenn oben und unten gleich sind (wie auch rechts und links) Kurzschreibweise für: <code>padding-top: 35px;</code> <code>padding-right: 18px;</code> <code>padding-bottom: 35px;</code> <code>padding-left: 18px;</code>
<code>padding: 8px;</code>	Wenn alle 4 Seiten gleich sind Kurzschreibweise für: <code>padding-top: 8px;</code> <code>padding-right: 8px;</code> <code>padding-bottom: 8px;</code> <code>padding-left: 8px;</code>
<code>margin:</code>	Dasselbe wie bei padding
<code>font:</code>	<code>font-size</code> und <code>font-family</code> sind Pflichtangaben, wobei die <code>font-family</code> immer die letzte Angabe ist. <code>font: 12pt Arial, sans-serif;</code> wenn auch die <code>line-height</code> mitgegeben wird, dann wird diese mit / von der Schriftgröße getrennt: <code>font:12pt/1.6em serif;</code>

CSS Validieren

<https://jigsaw.w3.org/css-validator/>

Pseudoklassen

Folgende Formate werden verwendet, damit ein Feedback bei Links erreicht werden kann.

Beispiel mit CSS-Code:

Sehr wichtig ist die Reihenfolge (Faustformel dazu: LoVe For HAtE – dazu sollte man den Film „Rosenkrieg“ gesehen haben)

```
#navi a:link      { color:blue;    text-decoration:none; }
#navi a:visited   { color:black;   text-decoration:line-through; }
#navi a:focus     { color:green;   text-decoration:underline; }
#navi a:hover    { color:red;     text-decoration:underline; }
#navi a:active    { color:orange;  text-decoration:underline; }
```

Im HTML-Code dann

```
<div id="navi">
  <ul>
    <li><a href="index.htm">Startseite</a></li>
    <li><a href="ueber-mich.htm">Über mich</a></li>
    <li><a href="termine.htm">Termine</a></li>
    <li><a href="impressum.htm">Impressum</a></li>
  </ul>
</div>
```

link	Der normale Anzeige für einen Link
visited	Einen bereits vom Nutzer besuchter Link
focus	Den Zustand von focus erreicht man, wenn man über die TAB-Taste (öfters drücken) zu einem Link wandert. Dieser kann dann mit der Return gewählt werden.
hover	Mit der Maus über einen Link fahren
active	Gerade aktiver Link

absolute/relative Positionierung

position	Art der Positionierung: relative static fixed absolute;
position: absolute;	weiter notwendige Angaben left right top bottom width height
z-index	Beispiel: z-index: 1; Reihenfolge der Elemente (wird oft bei absolut positionierten Elementen benötigt) Je höher die Nummer, desto weiter im Vordergrund

CSS3

Verschiedene CSS3-Anweisungen funktionieren nur in aktuellen Browserversionen!

Farbangaben CSS3

color: #00ff00;	alte Form der Farbangabe
color: rgb(0, 255, 0);	Farbangabe anhand RGB-Zahlen (dezimal)
color: rgb(0%, 100%, 0%);	Farbangabe in RGB anhand Prozenten
color: hsl(300, 100%, 60%);	Farbangabe im HSL-System H = hue = Farbton S = saturation = Sättigung L = lightness = Helligkeit H als Winkel von 0 – 360 S in Prozent von 0 – 100% L in Prozent von 0 – 100%



CSS3: Transparenz

opacity:	0 ... 1 Grad der Durchsichtigkeit (vererbt die Eigenschaft!)
color: rgba(..., ..., ..., 0.5)	0 ... 1 als 4 Wert zur Farbe als Transparenz
color: hsla(..., ..., ..., 0.5)	0 ... 1 als 4 Wert zur Farbe als Transparenz

CSS3: Schatten

box-shadow: Xpx Ypx Schattenverlauf Farbe;	Nun kann ein Schatten einem Element mitgegeben werden
box-shadow: Xpx Ypx Schattenverlauf Farbe , Xpx Ypx Schattenverlauf Farbe;	mehrere Schatten (Trennung erfolgt über Komma)
-webkit-box-shadow: 10px 10px 100px gray; -moz-box-shadow: 10px 10px 100px gray; box-shadow: 10px 10px 100px gray;	für verschiedene Browserhersteller! Allgemeiner Befehl kommt immer als letzter!

CSS3: abgerundete Ecken





border-radius: Xpx , Ypx;	Zum Abrunden von Ecken – ist nur 1 Wert angegeben, gilt dieser für X und Y
border-top-left-radius: border-top-right-radius: border-bottom-left-radius: border-bottom-right-radius:	Für jede Ecke einzeln definierbar
-moz-border-radius: 40px; -webkit-border-radius: 40px; border-radius: 40px; Besonderheiten Firefox: -moz-border-radius-topleft: 40px; -moz-border-radius-topright: 40px; -moz-border-radius-bottomleft: 40px; -moz-border-radius-bottomright: 40px;	für verschiedene Browser Besonderheiten Webkit-Browser: -webkit-border-radius: 40px 30px; /*Leerzeichen!*/



CSS3: Schatten bei Text

text-shadow: Xpx Ypx Schattenverlauf Farbe;	Beispiel: text-shadow: 5px 10px 8px orange;
text-shadow: -5px -5px 6px green, 5px 10px 6px red;	Mehrfarbiger Schatten

CSS3: Transformationen

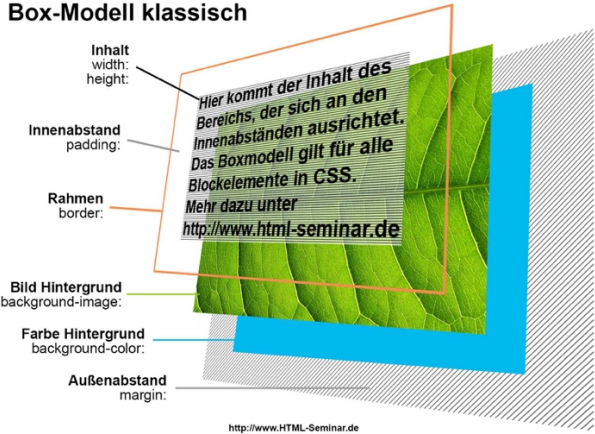
transform: rotate (Xdeg);	rotieren (drehen) Beispiel: transform: rotate(Xdeg); WICHTIG: kein Leerzeichen nach rotate!! positiver Wert = Drehung im Uhrzeigersinn negativer Wert = Drehung gegen Uhrzeigersinn 
-moz-transform: rotate(30deg); -ms-transform: rotate(30deg); -o-transform: rotate(30deg); -webkit-transform: rotate(30deg); transform: rotate(30deg);	rotate für verschiedene alte Browserversionen
transform-origin: X Y;	Mittelpunkt für die Drehung festlegen transform-origin: 0 0; /* obere linke Ecke */ transform-origin: 100% 0; /* obere rechte Ecke */ transform-origin: 100% 100%; /* untere rechte Ecke */ transform-origin: 50% 0; /* mitte oben */ 
transform: scale(wert); transform: scale(x, y);	skalieren bei Angabe eines Wertes wird x wie y vergrößert bei Unterschiedlicher Angabe von x wie y findet eine "Verzerrung" statt 
-moz-transform: scale(1.4); -ms-transform: scale(1.4); -o-transform: scale(1.4); -webkit-transform: scale(1.4); transform: scale(1.4);	scale für verschiedene alte Browserversionen
transform: skew(wert); transform: skew(x, y);	schräg, (wind)schief, verdrehen Beispiel: transform: skew(20deg); transform: skew(20deg, 30deg); 
-moz-transform: skew(0, 20deg); -ms-transform: skew(0, 20deg); -o-transform: skew(0, 20deg); -webkit-transform: skew(0, 20deg); transform: skew(0, 20deg);	skew für verschiedene alte Browserversionen

<pre>transform: translate(wert); transform: translate(...px, ...px);</pre>	<p>CSS3 translate: umsetzen, verrücken, versetzen</p> <pre>transform: translate(80px); transform: translate(80px, 50px);</pre> <p>bei Angabe eines Wertes wird nur auf x versetzt angezeigt. Bei Angabe von x wie y findet in beiden Richtungen eine entsprechende versetzte Anzeige statt.</p>
--	--

CSS3: mehrere Bilder im Hintergrund

<pre>background-image: url(css3-bilder-beispiel-saeulen.jpg), url(css3-bilder-beispiel-himmel.jpg);</pre>	<p>Es können in CSS3 nun mehrere Bilder angegeben werden. Das erste liegt im vor dem Zweiten etc.</p>
<pre>background-repeat: no-repeat, repeat-x;</pre>	<p>Alle weiteren Angaben bei Bilder können nun auch für die entsprechende Anzahl der Bilder angegeben werden – immer mit Komma getrennt.</p>

Box-Modell bei CSS

<pre>#beispielbox { padding: 30px; border: 1px solid blue; background-color:#141321; margin: 30px; background-image: url(blaetter.jpg); }</pre>	<p>Box-Modell klassisch</p>  <p>Das Diagramm zeigt ein 3D-Boxmodell mit folgenden Beschriftungen:</p> <ul style="list-style-type: none"> Inhalt (width: height:) Innenabstand (padding:) Rahmen (border:) Bild Hintergrund (background-image:) Farbe Hintergrund (background-color:) Außenabstand (margin:) <p>Quelle: http://www.HTML-Seminar.de</p>
---	---

HTML5-Befehle

<code><header> ... </header></code>	Bereich für Kopf (nicht verwechseln mit <code><head></code> !)
<code><nav> ... </nav></code>	Bereich Navigation (Steuerung)
<code><section> ... </section></code>	Gruppiert Elemente
<code><article> ... </article></code>	Bereich für Inhalt
<code><aside> ... </aside></code>	Bereich für Zusatzinformationen zum Inhalt
<code><footer> ... </footer></code>	Bereich für Fuß

Selektoren

	Typ-Selektor	ID-Selektor	Klassen-Selektor
Funktion	alle von dem ausgewählten HTML-TAG	nur 1 Element	alle in dieser Gruppe/Klasse
Bindung	gebunden an HTML-TAG	gebunden an ID (Attribut von HTML-TAG)	gebunden an CLASS (Attribut von HTML-TAG)
Aussehen in HTML	<code><h1></code>	<code><div id="steuerung"></code>	<code></code>
Aussehen in CSS	<code>h1 { ... }</code>	<code>#steuerung { ... }</code>	<code>.farbe1 { ... }</code>

Universalselektor

Für Reset von Abständen

```
* {
  padding: 0;
  margin: 0;
}
```

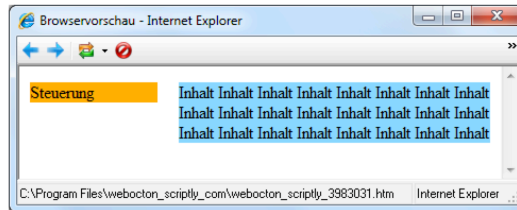
Nur 1 Element umwandeln:

```
display: block;
display: inline-block;
```

2-Spalten-Layout über float

Inhalt mit gleichem Abstand

```
nav {  
  background-color: orange;  
  width: 120px;  
  float: left;  
}
```



```
article {  
  margin-left: 140px;  
  background-color: blue;  
}
```

Tip: gleicher Abstand vom linken Seitenbereich über margin-left: 140px;

Wichtig bei der CSS-Anweisung float ist auch das Aufheben davon über: `clear: both;`

CSS Flexbox Layout

```
<div id="wrapper">
  <div id="bereichA"> Bereich 1 </div>
  <div id="bereichB"> Bereich 2 </div>
  <div id="bereichC"> Bereich 3 </div>
</div>
```

```
#bereichA { background-color: red; }
#bereichB { background-color: orange; }
#bereichC { background-color: limegreen; }
```

Aktivierung Flex-Box-Layout

```
#wrapper { display: flex; }
```

Zuweisung von Größen

```
#bereichA { flex: 1; }
#bereichB { flex: 4; }
#bereichC { flex: 2; }
```

Ändern der Anzeige-Reihenfolge

```
#wrapper {
  display: flex;
  flex-direction: row-reverse;
}
```

Reihenfolge selber festlegen der Flex-Items

```
#bereichA {
  order: 3;
}
```

Vertikale Ausrichtung der Spalten

```
#wrapper {
  display: flex;
  flex-direction: column;
}
```

Sticky Footer mit Flexbox

```
<div id="wrapper">
  <header>Kopfzeile </header>
  <article> Inhalt </article>
  <footer> Fußzeile </footer>
</div>
```

```
html, body {
  height: 100%;
  width: 100%;
  margin: 0;
  padding: 0;
}
```

```
#wrapper {
  display: flex;
  flex-direction: column;
  /* height: 100%; */
  min-height: 100%;
}
```

```
header {
  background-color: deepskyblue;
}
```

```
article {
  background-color: orange;
  flex: 1;
}
```

```
footer {
  background-color: limegreen;
}
```



diese Referenz für eigene Kurse verwenden ist kostenlos möglich

Wenn Sie diese Referenz für Ihre eigenen Kurse in Schule, Berufe oder wo auch immer verwenden wollen, können Sie dies in gedruckter Form bzw. als Folien gerne machen. Allerdings dürfen die Unterlagen nicht verändert werden und es sollte von Ihrer Webseite ein Link auf <https://www.html-seminar.de> gesetzt werden.

Referenz zum Download auf ihrer Website

Wenn Sie dieses PDF-Dokument als Download auf Ihrer Website anbieten wollen, ist Voraussetzung, dass Sie einen Link auf <https://www.html-seminar.de> setzen. Es ist nur gestattet, diese Unterlagen unverändert als PDF auf der eigenen Website zu publizieren (nicht in HTML gesetzt – das bedeutet unverändert ☺).

Verbesserungsvorschläge

Verbesserungsvorschläge sind willkommen (Kontakt siehe <https://www.html-seminar.de/kontakt.php>).

Copyright der Unterlagen liegt bei Axel Pratzner