

CSS Grundlagen

Inhalt

Was ist CSS?.....	3
Strukturbau allgemein.....	3
Verschachtelung (Eltern- und Kindelemente).....	3
Aufbau einer CSS-Regel (Syntax).....	3
ID oder Klasse – wann nimmt man was?.....	3
Element- oder Typenselektor.....	4
ID-Selektor.....	5
Klassen-Selector	6
CSS verwenden.....	7
direkt (im Element definiert).....	7
zentral (Formatierung findet im head der aktuellen Seite statt).....	7
externe Datei.....	8
CSS Klassen	9
Globale Klassen	9
Tag-spezifische Klassen	9
Pseudo-Klassen.....	9
CSS-Regeln Gruppieren	10
mehrere Elemente gleichzeitig definieren.....	10
mehrere CSS-Klassen in einem Attribut eines Elementes verwenden.....	11
Gewichtung von CSS-Regelsätzen (sog. Spezifität)	12
CSS-Boxen & Boxmodell	13
Bestandteile Box.....	13
Funktionsweise Boxmodell.....	13
Einzelbox	14
mehrere Boxen untereinander.....	16
mehrere Boxen nebeneinander	17
Außenabstände	19
Innenabstände.....	20
Farben.....	21
Farbnamen	21
Farocode RGB Hex.....	21
Farocode RGB Dezimal	21
Beispiele	25
Textpositionierung	25

CSS Grundlagen

CSS Grundlagen

Was ist CSS?

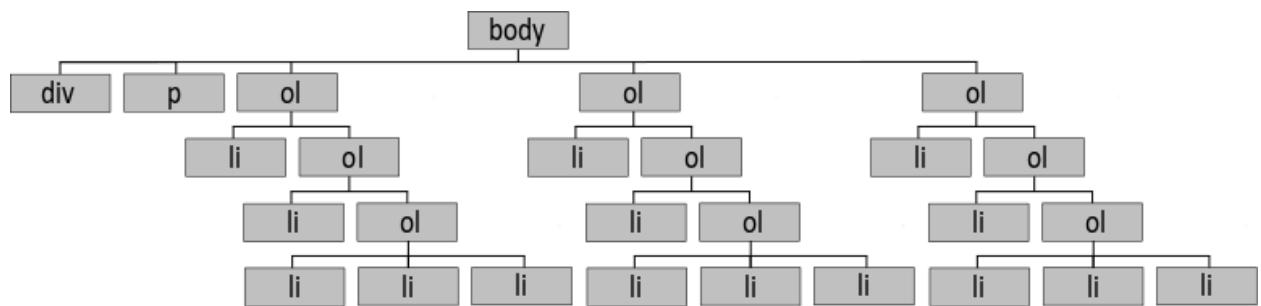
Mit Hilfe von CSS (Cascading Stylesheets) wird das Design einer HTML Website festgelegt. Dies können u.a. sein: Schrift-, Rand-, Aussehen, Farben, Positionierung

Strukturbau allgemein

Der Strukturbau zeigt grafisch das Vorhandensein und die Verschachtelung von Elementen einer HTML Datei. Ein Strukturbau erleichtert es stets, den Überblick zu behalten.

Ausgangspunkt für einen Strukturbau ist das *body*-Element, da dieses den gesamten anzuzeigenden Inhalt umschließt. Man dieses auch das Wurzel-Element.

Grundlage ist der HTML-Quellcode lt. Anlage „Beispiel Textpositionierung“



Verschachtelung (Eltern- und Kindelemente)

Das Verständnis von Eltern- und Kindelementen ist bei der Formatierung sehr wichtig. Das Generationenprinzip symbolisiert die Vererbung (Weitergabe) von Eigenschaften der Eltern an die Kinder.

Dadurch eröffnen sich verschiedene Möglichkeiten für Design. Im ersten Schritt schauen wir uns das Beispiel mit Platzieren eines Elementes am Rand unten rechts an und dann das Konzept der Eltern- und Kindelemente.

Aufbau einer CSS-Regel (Syntax)

Selector	Eigenschaft	Wert
h1 {	color:	red }

ID oder Klasse – wann nimmt man was?

Der Unterschied zwischen *class* und *id* bereitet vielen CSS-Einsteigern Probleme, ist aber im Grunde ganz einfach:

- Beide bieten die Möglichkeit, eigene Namenszusätze für HTML-Elemente zu vergeben.
- Eine *id* ist einmalig und darf auf einer Webseite nur einmal vorkommen.
- Eine *class* darf auf einer Webseite mehrfach verwendet werden.

Die Hauptbereiche der Beispieleite wie #kopfbereich oder #textbereich bekommen eine *id*, weil sie pro Seite einmalig sind. Andere Elemente wie z. B. Infoboxen können hingegen mehrfach auftreten und bekommen deshalb eine *class*.

CSS Grundlagen

Element- oder Typenselektor

Die Definition eines Elementes wird durch **Tag-Name** eingeleitet, es folgt anschl. innerhalb einer geschweiften Klammer die Eigenschaften(n).

Wird das mit dem Tag-Name definierte Element im HTML-Code verwendet, so wird deren Definition automatisch angewandt.

Beispiel 1

CSS Element-Selector

```
<style>
h1 {
    color:gray;          /* h-Tag */
    text-shadow: black 1px 1px 1px; /* Schriftfarbe Grau */
    /* Schriftschatten Schwarz 3-Seiten*/
}
</style>
```

HTML ID-Selector

```
<h1>Beispielüberschrift</h1>
```

Anzeige

Beispielüberschrift

Achtung: Gültigkeit, Eigenschaften von Blockelementen beachten.

Beispiel 2

CSS Element-Selector

```
<style>
p {
    color:blue;          /* p-Tag */
    /* Schriftfarbe Blau */
}
</style>
```

HTML ID-Selector

```
<p>Beispielabsatz</p>
```

Anzeige

Beispielabsatz

Achtung: Gültigkeit, Eigenschaften von Blockelementen beachten.

CSS Grundlagen

ID-Selektor

Beispiel 1

CSS ID-Selector

```
<style>
#id_selector {
    color:gray;
}
</style>
```

HTML ID-Selector

```
<div id="id_selector">Beispielinhalt</div>
```

ID-Selector nur 1 mal pro HTML Seite verwenden

Anzeige

Beispielinhalt

Beispiel 2

CSS ID-Selector

```
<style>
#muster {border: red 2px solid;}           /* Randfarbe Rot, -stärke 2px, -art durchgängige Linie */
</style>
```

HTML

```
<div id="muster">Diese DIV-Box wurde mit einem ID-Selector definiert. </div>
```

Anzeige

Diese DIV-Box wurde mit einem ID-Selector definiert.

CSS Grundlagen

Klassen-Selector

CSS Klassen-Selector

```
<style>
.class_selector{
  color:blue;
}
<style>
```

HTML ID-Selector

```
<div class="class selector">Beispielinhalt</div>
```

Klassen-Selector beliebig viele Elemente pro HTML Seite einsetzbar

Anzeige

Beispielinhalt

Achtung: keine Umlaute (ä, ö, ü, ß)

Hinweis: case sensitiv (Groß- Kleinschreibung beachten)

Empfehlung: ausschließlich Kleinschreibung verwenden

CSS Grundlagen

CSS verwenden

direkt (im Element definiert)

HTML

```
<p style="text-align: center; color: green;">Dieser Absatz wird über ein style-Attribut formatiert. </p>
```

Anzeige

Dieser Absatz wird über ein style-Attribut formatiert.

zentral (Formatierung findet im head der aktuellen Seite statt)

CSS ID-Selector

```
<head>
<style>
  p {text-align: center; color: green}           /* Textausrichtung: zentriert, Textfarbe: grün */
</style>
</head>
```

HTML

```
<p>Dieser Absatz wird über ein style-Attribut formatiert. </p>
```

Anzeige

Dieser Absatz wird über ein style-Attribut formatiert.

Hinweis:

Es können mehrere Eigenschaften mit ihrem Wert gleichzeitig für einen Selector definiert werden, dann sind diese durch ein Semikolon zu trennen.

CSS Grundlagen

externe Datei

CSS Datei (z.B. "stylesheet.css")

```
<style>
  p {text-align: center; color: green}      /* Textausrichtung: zentriert, Textfarbe: grün */
</style>
```

HTML

```
<head>
<link href="stylesheet.css" rel="stylesheet" type="text/css">
</head>
<body>
...
<p>Dieser Absatz wird über ein style-Attribut formatiert, das in einer externen Datei definiert wird.</p>
...
</body>
```

Hinweis: Verweis auf externe Datei "stylesheet.css" im *head* Bereich der aktuellen Seite

Anzeige

Dieser Absatz wird über ein style-Attribut formatiert.

CSS Grundlagen

CSS Klassen

Globale Klassen

Globale Klassen werden über die gesamte Seite angewandt ohne extra aufgerufen werden zu müssen.

CSS

```
<style>
  .{background-color: blue}          /* Hintergrundfarbe Blau */
  .notiz{font-style: italic}         /* Schriftart: italic */
</style>
```

Tag-spezifische Klassen

Tag-spezifische Klassen werden nur dort wirksam, wo sie im entsprechenden Element (hier *p*) verknüpft werden.

CSS

```
<style>
  p.notiz{font-style: arial}        /* gilt nur für p-Elemente mit Klasse notiz */
</style>
```

HTML

```
<p class="notiz">Dieser Absatz wird über ein style-Attribut formatiert.</p>
```

Pseudo-Klassen

Pseudo Klassen werden angewandt, wenn Elemente durch ein Ereignis (Mauszeiger darüber, anklicken o.ä.) angesprochen werden.

CSS

```
<style>
  a:link{color: blue}              /* gilt nur für p-Elemente mit Klasse notiz */
  a:hover{color: yellow}           /* Mauszeiger über Link */
  a:visited{color: green}          /* Seite wurde bereits besucht */
</style>
```

CSS Grundlagen

CSS-Regeln Gruppieren

mehrere Elemente gleichzeitig definieren

Durch Gruppieren, können mehrere Elemente gleichzeitig definiert werden.
Dies spart Zeit, Platz und dadurch lassen Anpassungen schneller umsetzen.

CSS Datei (z.B. "stylesheet.css")

```
<style>
h1, h2, h3 { border-bottom: 2px solid gray; } /* Rahmen -farbe Grau */
</style>
```

HTML

```
<h1>Überschrift 1</h1>
<h2>Überschrift 2</h2>
<h3>Überschrift 3</h3>
```

Anzeige

Überschrift 1
Überschrift 2
Überschrift 3

CSS Grundlagen

mehrere CSS-Klassen in einem Attribut eines Elementes verwenden

Damit CSS-Eigenschaften nicht in vielen Regeln wiederholt werden müssen, lassen sich mehrere Klassen in ein CLASS-Attribut schreiben.

CSS Datei (z.B. "stylesheet.css")

```
<style>
.green { background-color: green; color: white; } /* Hintergrundfarbe Grün, Schriftfarbe Weiß */
.large { font-size: 1.1em; line-height: 250%; }
</style>
```

HTML

```
<h2 class="green">Beispieltext 1</h2>
<p class="green large">Beispieltext 2</p>
```

Anzeige

Beispieltext 1

Beispieltext 2

CSS Grundlagen

Gewichtung von CSS-Regelsätzen (sog. Spezifität)

Die Gewichtung legt fest, mit welcher Priorität die Definitionen angewandt oder auch überschrieben werden.

siehe auch: <https://wiki.selfhtml.org/wiki/CSS/Kaskade>

CSS Grundlagen

CSS-Boxen & Boxmodell

Als sehr nützlichen Baustein spielen Boxen eine große Rolle.

Mit ihnen ist es möglich, Texte, grafische Elemente, Bilder, Schaltflächen oder auch nur „Platzhalter“ sowie vieles mehr, nicht nur vom Aussehen her zu beeinflussen, sondern auch deren Positionierung.

Bestandteile Box

Jede Box besteht aus einem Rechteckigen Bereich.

Diese Box kann durch folgende Angaben angepasst werden:

- äußeren Abstand (*margin*)
- Rahmen (*border*)
- Inhalt: Länge (engl. *width*), Breite (engl. *height*)
- innerer Abstand (*padding*)

Funktionsweise Boxmodell

Der Rahmen (*border*) bildet die Trennung zwischen äußeren Abstand (*margin*) zum Bezugspunkt (Elternelement) und dem inneren Abstand (*padding*) zum Inhalt.

Als Bezugspunkt wird (wenn keine weitere Box vorhanden) automatisch die linke, obere Ecke des Browserfensters verwendet.



Hinweis:

Die Bezeichnung **Box** ist nicht der Bezeichnung **Element** gleichzusetzen.
Nicht jede Box erzeugt ein Element

CSS Grundlagen

Einzelbox

Beispiel 1

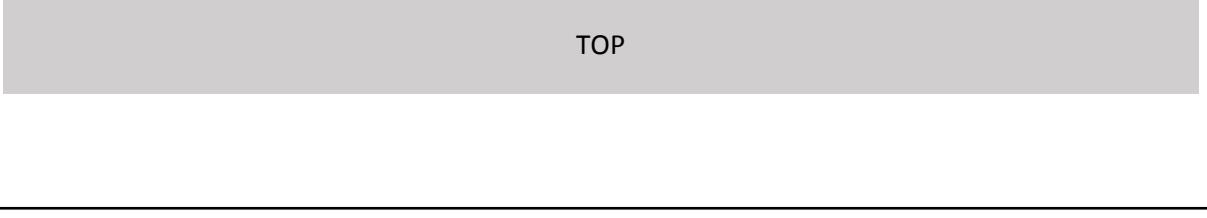
CSS (Box) Beispiel 1

```
<style>
  #top {
    height: 100px;          /* Höhe = 100 Pixel */
    text-align: center;      /* Textausrichtung: zentriert (horizontal) */
    color: white;           /* Textfarbe: Weiß */
    background-color: silver; /* Hintergrundfarbe: Silber*/
  }
</style>
```

HTML

```
<html>
<body>
<div id="top">TOP</div>
</body>
</html>
```

Anzeige



TOP

Hinweis:

Wurde keine Breite für eine Box angegeben, so erstreckt sich diese bis vom linken bis zum rechten Rand.

CSS Grundlagen

Beispiel 2

Das folgende Beispiel ist analog dem Vorangegangen, mit dem Unterschied, dass es keine Abstände zu den Rändern (margin = 0; padding = 0) aufweist.

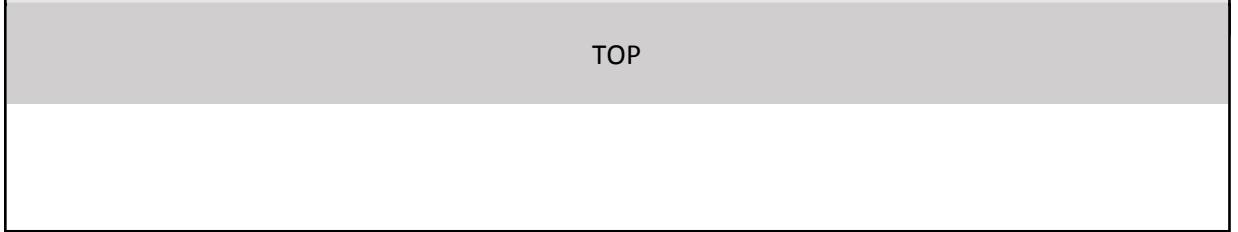
CSS (Box) Beispiel 2

```
</style>
body, html {
    margin: 0; /* Elemente */
    padding: 0; /* Abstand zum Rand = 0 -kein- */
    /* Abstand zum Inhalt = 0 -kein- */
}
#top {
    height: 100px; /* Höhe = 100 Pixel */
    text-align: center; /* Textausrichtung: zentriert (horizontal) */
    color: white; /* Textfarbe: Weiß */
    background-color: silver; /* Hintergrundfarbe: Silber*/
}
</style>
```

HTML

```
<html>
<body>
<div id="top">TOP</div>
</body>
</html>
```

Anzeige



TOP

CSS Grundlagen

mehrere Boxen untereinander

Beispiel 1

CSS (2 Boxen untereinander) Beispiel 1

```
</style>
body, html {
    margin: 0; /* Elemente */
    padding: 0; /* Abstand = 0 -kein- */
}
.box_1{
    border: red 2px solid;
}
.box_2{
    border: green 2px solid;
}
</style>
```

HTML

```
<html>
<body>
<div class="box_1">Box 1</div>
<div class="box_2">Box 2</div>
</body>
```

Anzeige

Box 1

Box 2

Hinweis: Wurde keine Breite für eine Box angegeben, so erstreckt sich diese vom linken über den Inhalt (hier im Beispiel der Text „Box ...“) hinaus bis zum rechten Rand.

CSS Grundlagen

mehrere Boxen nebeneinander

Beispiel 1

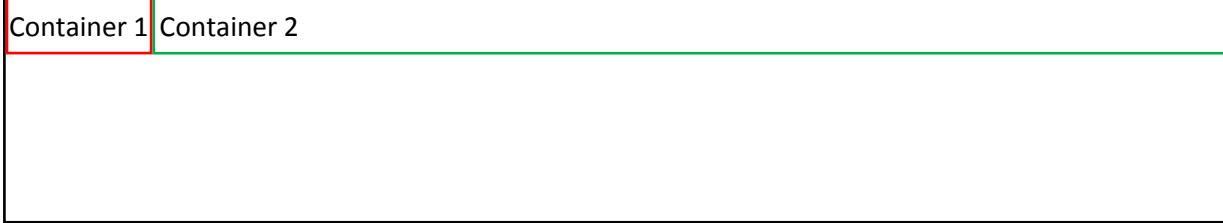
CSS (2 Boxen nebeneinander) Beispiel 1

```
</style>
body, html {
    margin: 0; /* Elemente */
    padding: 0; /* Abstand = 0 -kein- */
}
#hauptcontainer {
    margin: 100px 0 0 10px; /* äußere Abstände */
    width: 100%; /* gesamte Breite der Anzeige verwenden */
}
#container1 {
    float: left; /* Start- Ausrichtungsposition der Box */
    margin-left: 0px;
    margin-top: 0px;
    border: red 2px solid; /* Box-Rahmenfarbe: rot, -breite: 2px; durchgezogene Linie */
}
#container2 {
    margin-left: 0px;
    margin-top: 0px;
    border: green 2px solid;
}
</style>
```

HTML

```
<html>
<body>
<div id="hauptcontainer">
    <div id="container1">Container 1</div>
    <div id="container2">Container 2</div>
</div></body>
</html>
```

Anzeige



Container 1 Container 2

Achtung: Um mehrere Div-Boxen gezielt zu positionieren, ist es wichtig, jeweils den Bezugspunkt (äußerer Abstand / Rahmen) zu definieren.

Alternativ ist es möglich jede Box mit absoluten Positionen zu versehen. Je nach Displaygröße kann sich das jedoch störend auf die Anzeige auswirken.

Empfehlung: Zusätzliche Div-Box verwenden, die die zu positionierenden Boxen umschließt.
siehe Div-Box „hauptcontainer“

Hinweis: Wurde keine Breite für eine Box angegeben, so erstreckt sich diese vom linken über den Inhalt (hier im Beispiel der Text „Container 1“ hinaus bis zum rechten Rand).

Ausnahme: Es existiert eine „Nachbar-Box“, dann endet die Box am Ende des Inhaltes.

CSS Grundlagen

CSS Grundlagen

Außenabstände (*margin* = engl. Rand)

Die äußersten Abstände eines Elementes können mit *margin* angepasst werden. Für jede Seite können unterschiedliche Werte vergeben werden.

margin-top (Rand oben), *margin-right* (Rand rechts), *margin-bottom* (Rand unten), *margin-left* (Rand links)

Es können unterschiedliche Schreibweisen verwendet werden, um die Abstände zu beeinflussen.

CSS (gleiche Abstände an allen 4 Seiten) Beispiel 1

```
<style>
  body, html {
    margin: 0;           /* Elemente */
    /* Abstand = 0 -kein- */
  }
</style>
```

CSS (gleiche Abstände an allen 4 Seiten) Beispiel 2

```
<style>
  body, html {
    margin-top: 0;        /* Elemente */
    /* Abstand oben */
    margin-right: 0;      /* Abstand rechts */
    margin-bottom: 0;     /* Abstand unten */
    margin-left: 0;       /* Abstand links */
  }
</style>
```

CSS (gleiche Abstände an allen 4 Seiten) Beispiel 3

```
<style>
  body, html {
    margin: 0 0 0 0;      /* Elemente */
    /* Abstand oben, rechts, unten, links = 0 -kein- */
  }
</style>
```

Das Beispiel 3 zeigt die verkürzte Schreibweise gegenüber der im Beispiel 2.

Die verkürzte Schreibweise bietet sich an, wenn unterschiedliche Abstände zum Einsatz kommen sollen. Folgende Tabelle zeigt Kombinationsmöglichkeiten zur Angabe von Werten.

Anzahl an Werten Auswirkung

1 Wert	alle vier Ränder erhalten den gleichen festgelegten Wert
2 Werte	erster Wert gilt für oben und unten, der zweite Wert für rechts und links
3 Werte	erster Wert gilt für oben, der zweite für rechts und links, der dritte für unten
4 Werte	erster Wert gilt für oben, der zweite für rechts, der dritte für unten, der vierte für rechts

Hinweis: Wird **kein** Wert für den Abstand angegeben, werden Werte aus den Browser-Einstellungen übernommen.
(was zu ungewollten Darstellungsänderungen führen kann!)

Empfehlung: Geben Sie für *body* und *html* den Wert 0 an (sowohl für *margin* als auch für *padding*).

CSS Grundlagen

Innenabstände (*padding* = engl. Polsterung)

Die innenren Abstände eines Elementes können mit *padding* angepasst werden. Für jede Seite können unterschiedliche Werte vergeben werden.

padding-top (Polsterung oben), *padding-right* (Polsterung rechts), *padding-bottom* (Polsterung unten), *padding-left* (Polsterung links)

Es können unterschiedliche Schreibweisen verwendet werden, um die Abstände zu beeinflussen.

CSS (gleiche Abstände an allen 4 Seiten) **Beispiel 1**

```
<style>
  body, html {
    padding: 0;           /* Elemente */
    /* Abstand = 0 -kein- */
  }
</style>
```

CSS (gleiche Abstände an allen 4 Seiten) **Beispiel 2**

```
<style>
  body, html {
    padding-top: 0;        /* Elemente */
    /* Abstand oben */
    padding-right: 0;      /* Abstand rechts */
    padding-bottom: 0;     /* Abstand unten */
    padding-left: 0;       /* Abstand links */
  }
</style>
```

CSS (gleiche Abstände an allen 4 Seiten) **Beispiel 3**

```
<style>
  body, html {
    padding: 0 0 0 0;      /* Elemente */
    /* Abstand oben, rechts, unten, links = 0 -kein- */
  }
</style>
```

Das Beispiel 3 zeigt die verkürzte Schreibweise gegenüber der im Beispiel 2.

Die verkürzte Schreibweise bietet sich an, wenn unterschiedliche Abstände zum Einsatz kommen sollen. Folgende Tabelle zeigt Kombinationsmöglichkeiten zur Angabe von Werten.

Anzahl an Werten Auswirkung

1 Wert	alle vier Ränder erhalten den gleichen festgelegten Wert
2 Werte	erster Wert gilt für oben und unten, der zweite Wert für rechts und links
3 Werte	erster Wert gilt für oben, der zweite für rechts und links, der dritte für unten
4 Werte	erster Wert gilt für oben, der zweite für rechts, der dritte für unten, der vierte für rechts

Hinweis: Wird **kein** Wert für den Abstand angegeben, werden Werte aus den Browser-Einstellungen übernommen.
(was zu ungewollten Darstellungsänderungen führen kann!)

Empfehlung: Geben Sie für *body* und *html* den Wert 0 an (sowohl für *margin* als auch für *padding*).

CSS Grundlagen

Farben

Die Farbenlehre ist **nicht** Bestandteil dieses CSS-Grundlagen Kurses.
Die Angabe der Farbe kann über folgende Möglichkeiten erfolgen.

Farbnamen Möglichkeit 1

Farbcode RGB Hex Möglichkeit 2

Farbcode RGB Dezimal Möglichkeit 3

Quelle: <https://www.w3.org/TR/css-color-3/#svg-color>

.	Farbname	Hex	RGB	Dezimal
	aliceblue	#f0f8ff	240,248,255	
	antiquewhite	#faebd7	250,235,215	
	aqua	#00ffff	0,255,255	
	aquamarine	#7ffffd	127,255,212	
	azure	#f0ffff	240,255,255	
	beige	#f5f5dc	245,245,220	
	bisque	#ffe4c4	255,228,196	
	black	#000000	0,0,0	
	blanchedalmond	#ffebcd	255,235,205	
	blue	#0000ff	0,0,255	
	blueviolet	#8a2be2	138,43,226	
	brown	#a52a2a	165,42,42	
	burlywood	#deb887	222,184,135	
	cadetblue	#5f9ea0	95,158,160	
	chartreuse	#7fff00	127,255,0	
	chocolate	#d2691e	210,105,30	
	coral	#ff7f50	255,127,80	
	cornflowerblue	#6495ed	100,149,237	
	cornsilk	#fff8dc	255,248,220	
	crimson	#dc143c	220,20,60	
	cyan	#00ffff	0,255,255	
	darkblue	#00008b	0,0,139	
	darkcyan	#008b8b	0,139,139	
	darkgoldenrod	#b8860b	184,134,11	
	darkgray	#a9a9a9	169,169,169	
	darkgreen	#006400	0,100,0	
	darkgrey	#a9a9a9	169,169,169	
	darkkhaki	#bdb76b	189,183,107	
	darkmagenta	#8b008b	139,0,139	
	darkolivegreen	#556b2f	85,107,47	
	darkorange	#ff8c00	255,140,0	
	darkorchid	#9932cc	153,50,204	
	darkred	#8b0000	139,0,0	

CSS Grundlagen

	<i>darksalmon</i>	#e9967a	233,150,122
	<i>darkseagreen</i>	#8fbcc8f	143,188,143
	<i>darkslateblue</i>	#483d8b	72,61,139
	<i>darkslategray</i>	#2f4f4f	47,79,79
	<i>darkslategrey</i>	#2f4f4f	47,79,79
	<i>darkturquoise</i>	#00ced1	0,206,209
	<i>darkviolet</i>	#9400d3	148,0,211
	<i>deeppink</i>	#ff1493	255,20,147
	<i>deepskyblue</i>	#00bfff	0,191,255
	<i>dimgray</i>	#696969	105,105,105
	<i>dimgrey</i>	#696969	105,105,105
	<i>dodgerblue</i>	#1e90ff	30,144,255
	<i>firebrick</i>	#b22222	178,34,34
	<i>floralwhite</i>	#ffffaf0	255,250,240
	<i>forestgreen</i>	#228b22	34,139,34
	<i>fuchsia</i>	#ff00ff	255,0,255
	<i>gainsboro</i>	#dcdcdc	220,220,220
	<i>ghostwhite</i>	#f8f8ff	248,248,255
	<i>gold</i>	#ffd700	255,215,0
	<i>goldenrod</i>	#daa520	218,165,32
	<i>gray</i>	#808080	128,128,128
	<i>green</i>	#008000	0,128,0
	<i>greenyellow</i>	#adff2f	173,255,47
	<i>grey</i>	#808080	128,128,128
	<i>honeydew</i>	#f0fff0	240,255,240
	<i>hotpink</i>	#ff69b4	255,105,180
	<i>indianred</i>	#cd5c5c	205,92,92
	<i>indigo</i>	#4b0082	75,0,130
	<i>ivory</i>	#fffff0	255,255,240
	<i>khaki</i>	#f0e68c	240,230,140
	<i>lavender</i>	#e6e6fa	230,230,250
	<i>lavenderblush</i>	#ffff0f5	255,240,245
	<i>lawngreen</i>	#7fcfc00	124,252,0
	<i>lemonchiffon</i>	#ffffacd	255,250,205
	<i>lightblue</i>	#add8e6	173,216,230
	<i>lightcoral</i>	#f08080	240,128,128
	<i>lightcyan</i>	#e0ffff	224,255,255
	<i>lightgoldenrodyellow</i>	#fafad2	250,250,210
	<i>lightgray</i>	#d3d3d3	211,211,211
	<i>lightgreen</i>	#90ee90	144,238,144
	<i>lightgrey</i>	#d3d3d3	211,211,211
	<i>lightpink</i>	#ffb6c1	255,182,193
	<i>lightsalmon</i>	#ffa07a	255,160,122
	<i>lightseagreen</i>	#20b2aa	32,178,170
	<i>lightskyblue</i>	#87cefa	135,206,250
	<i>lightslategray</i>	#778899	119,136,153
	<i>lightslategrey</i>	#778899	119,136,153

CSS Grundlagen

<i>lightsteelblue</i>	#b0c4de	176,196,222
<i>lightyellow</i>	#ffffe0	255,255,224
<i>lime</i>	#00ff00	0,255,0
<i>limegreen</i>	#32cd32	50,205,50
<i>linen</i>	#faf0e6	250,240,230
<i>magenta</i>	#ff00ff	255,0,255
<i>maroon</i>	#800000	128,0,0
<i>mediumaquamarine</i>	#66cdaa	102,205,170
<i>mediumblue</i>	#0000cd	0,0,205
<i>mediumorchid</i>	#ba55d3	186,85,211
<i>mediumpurple</i>	#9370db	147,112,219
<i>mediumseagreen</i>	#3cb371	60,179,113
<i>mediumslateblue</i>	#7b68ee	123,104,238
<i>mediumspringgreen</i>	#00fa9a	0,250,154
<i>mediumturquoise</i>	#48d1cc	72,209,204
<i>mediumvioletred</i>	#c71585	199,21,133
<i>midnightblue</i>	#191970	25,25,112
<i>mintcream</i>	#f5ffff	245,255,250
<i>mistyrose</i>	#ffe4e1	255,228,225
<i>moccasin</i>	#ffe4b5	255,228,181
<i>navajowhite</i>	#ffdead	255,222,173
<i>navy</i>	#000080	0,0,128
<i>oldlace</i>	#fdf5e6	253,245,230
<i>olive</i>	#808000	128,128,0
<i>olivedrab</i>	#6b8e23	107,142,35
<i>orange</i>	#ffa500	255,165,0
<i>orangered</i>	#ff4500	255,69,0
<i>orchid</i>	#da70d6	218,112,214
<i>palegoldenrod</i>	#eee8aa	238,232,170
<i>palegreen</i>	#98fb98	152,251,152
<i>paleturquoise</i>	#afeeee	175,238,238
<i>palevioletred</i>	#db7093	219,112,147
<i>papayawhip</i>	#ffefc5	255,239,213
<i>peachpuff</i>	#ffdab9	255,218,185
<i>peru</i>	#cd853f	205,133,63
<i>pink</i>	#ffc0cb	255,192,203
<i>plum</i>	#dda0dd	221,160,221
<i>powderblue</i>	#b0e0e6	176,224,230
<i>purple</i>	#800080	128,0,128
<i>red</i>	#ff0000	255,0,0
<i>rosybrown</i>	#bc8f8f	188,143,143
<i>royalblue</i>	#4169e1	65,105,225
<i>saddlebrown</i>	#8b4513	139,69,19
<i>salmon</i>	#fa8072	250,128,114
<i>sandybrown</i>	#f4a460	244,164,96
<i>seagreen</i>	#2e8b57	46,139,87
<i>seashell</i>	#ffff5ee	255,245,238

CSS Grundlagen

sienna	#a0522d	160,82,45
silver	#c0c0c0	192,192,192
skyblue	#87ceeb	135,206,235
slateblue	#6a5acd	106,90,205
slategray	#708090	112,128,144
slategrey	#708090	112,128,144
snow	#fffffa	255,250,250
springgreen	#00ff7f	0,255,127
steelblue	#4682b4	70,130,180
tan	#d2b48c	210,180,140
teal	#008080	0,128,128
thistle	#d8bfd8	216,191,216
tomato	#ff6347	255,99,71
turquoise	#40e0d0	64,224,208
violet	#ee82ee	238,130,238
wheat	#f5deb3	245,222,179
white	#ffffff	255,255,255
whitesmoke	#f5f5f5	245,245,245
yellow	#ffff00	255,255,0
yellowgreen	#9acd32	154,205,50

CSS Grundlagen

Beispiele

Textpositionierung

Anzeige

	Inhaltsverzeichnis
Überschrift 1	Überschrift 1
Absatz 1	Überschrift 2
Text 1	Überschrift 3
Text 2	
Text 3	
Überschrift 2	
Absatz 1	
Text 1	
Text 2	
Text 3	
Überschrift 3	
Absatz 1	
Text 1	
Text 2	
Text 3	

HTML

```
<body>
<div align="center">Inhaltsverzeichnis</div>
<br />
<p align="center">Überschrift 1<br />Überschrift 2<br />Überschrift 3</p>
<ol>
    <li>Überschrift 1</li>
    <ol>
        <li>Absatz 1</li>
        <ol>
            <li>Text 1</li>
            <li>Text 2</li>
            <li>Text 3</li>
        </ol>
    </ol>
</ol>
<ol>
    <li>Überschrift 2</li>
    <ol>
        <li>Absatz 1</li>
        <ol>
            <li>Text 1</li>
            <li>Text 2</li>
            <li>Text 3</li>
        </ol>
    </ol>
</ol>
<ol>
    <li>Überschrift 3</li>
    <ol>
        <li>Absatz 1</li>
        <ol>
            <li>Text 1</li>
            <li>Text 2</li>
            <li>Text 3</li>
        </ol>
    </ol>
</ol>
<ol>
    <li>Überschrift 4</li>
</ol>
</body>
```