

## 13.5 Nachfahren auswählen: Der Selektor mit dem Leerzeichen

## 13.6 Selektoren zum Auswählen von Kindelementen

13.6.1 Der Kindselektor: Der Selektor mit dem »>« (Größer-als-Zeichen)

13.6.2 Praktisch: Die Pseudoklassen »:first-child« und »:last-child«

13.6.3 Der Zauberstab zum Auswählen von Kindern: »:nth-child()«

## 13.7 Nachbar- und Geschwisterselektoren mit + und ~

## 13.8 Attributselektoren haben eckige Klammern: [attribut]

13.8.1 Der Selektor [attribut] prüft nur, ob es das Attribut gibt

13.8.2 Attributselektoren mit einem Gleichheitszeichen: [attribut="wert"]

13.8.3 Attributselektoren mit Tilde und Gleichheitszeichen: [attribut~="wert"]

13.8.4 Attributselektoren mit senkrechtem Strich und Gleichheitszeichen:  
[attribut|= "wert"]

13.8.5 Attributselektoren mit Hütchen und Gleichheitszeichen: [attribut^="wert"]

13.8.6 Attributselektoren mit Dollar und Gleichheitszeichen:  
element[attribut\$="zeichen"]

13.8.7 Attributselektoren mit Sternchen und Gleichheitszeichen:  
element[attribut\*="zeichen"]

## 13.9 Quellen zum Nachschlagen von weiteren Selektoren

## 13.10 Spezifität: Einige Selektoren sind wichtiger als andere

13.10.1 Einer wird gewinnen: So funktioniert Spezifität

13.10.2 Ganz sparsam benutzen: »!important« macht Anweisungen WICHTIG!

## 13.11 Auf einen Blick

# 14 Der Browser und das CSS: Kaskade, Vererbung oder Standardwert

---

## 14.1 Die Kaskade: Wichtigkeit, Spezifität und Reihenfolge

14.1.1 Die Kaskade hilft dem Browser, genau eine Anweisung zu finden

14.1.2 Die Ausgangssituation: Das Beispiel im Überblick

## 14.2 Intuitiv: Die Reihenfolge im CSS entscheidet

## 14.3 Ungewohnt: Spezifität ist wichtiger als Reihenfolge

## 14.4 Ausnahme: »!important« gewinnt immer

## 14.5 Nichts gefunden? Vererbung oder Standardwert

- 14.5.1 »Vererbung« macht ein Stylesheet übersichtlicher
  - 14.5.2 Bestimmte Eigenschaften werden nicht vererbt
  - 14.5.3 Falls er gar nichts findet, nimmt der Browser den »Standardwert«
- 14.6 Die Kaskade im Browser analysieren
- 14.7 Übersicht: Kaskade, Vererbung oder Standardwert
- 14.8 Auf einen Blick

## 15 Schrift und Text gestalten per CSS

---

- 15.1 Klassische Schriftarten mit und ohne Serifen im Web
- 15.1.1 Es gibt Schriftarten mit und ohne »Serifen«
  - 15.1.2 Sehr praktisch: Die Schriftgestaltung in Firefox analysieren
- 15.2 Die Schriftart definieren mit »font-family«
- 15.2.1 Bitte eine Schriftart ohne Serifen mit »font-family«
  - 15.2.2 Generische Schriftfamilien im Überblick
- 15.3 Die Systemschrift des Geräts: Gut lesbar und echt schnell
- 15.4 Webfonts – die Schriftart gleich mitliefern
- 15.5 Schnell und einfach: Google Fonts direkt von Google
- 15.5.1 Schritt 1: Schriftart und Schriftschnitte auswählen
  - 15.5.2 Schritt 2: Den Code für die Schriftart kopieren und einfügen
- 15.6 Auf der sicheren Seite: Google Fonts selbst ausliefern
- 15.6.1 Schritt 1: Schriftart und Zeichensatz auswählen
  - 15.6.2 Schritt 2: Die gewünschten Styles festlegen
  - 15.6.3 Schritt 3: Den Code für die Schriftarten kopieren und einfügen
  - 15.6.4 Schritt 4: Schriftdateien herunterladen und im Ordner »font« speichern
- 15.7 Gut lesbarer Text: »font-size« und »line-height«
- 15.7.1 Schriftgröße definieren mit »font-size« und einer Längeneinheit
  - 15.7.2 Die Schriftgröße für die Überschriften ändern
  - 15.7.3 Schriftgröße definieren mit »font-size« und einem Wort
  - 15.7.4 Wichtig für die Lesbarkeit: Der Zeilenabstand mit »line-height«
- 15.8 Listen: Aufzählungszeichen gestalten per CSS
- 15.8.1 Die Gestaltung von Listen in den Browser-Stylesheets
  - 15.8.2 Aufzählungszeichen gestalten mit »list-style« & Co
  - 15.8.3 Aufzählungszeichen gestalten mit dem Pseudoelement »::marker«

## 15.9 Hyperlinks: Unterstreichungen gestalten mit »text-decoration«

15.9.1 Zusätzliche Eigenschaften zur Unterstreichung von Links

15.9.2 Die Unterstreichungen der Links gestalten

## 15.10 Hyperlinks: Linkzustände gestalten mit Pseudoklassen

15.10.1 Besuchte und nicht besuchte Hyperlinks mit »:link« und »:visited«

15.10.2 Benutzeraktionen gestalten mit »:hover«, »:focus« und »:active«

## 15.11 Externe Hyperlinks kennzeichnen mit »::after«

15.11.1 Schritt 1: Externe Hyperlinks auswählen mit einem Attributselektor

15.11.2 Schritt 2: Das Pseudoelement »::after« und die Eigenschaft »content«

15.11.3 Schritt 3: Links kennzeichnen mit einem Unicode-Symbol

15.11.4 Alternative: Externe Links mit »target="\_blank"« selektieren

## 15.12 Weitere Eigenschaften zur Gestaltung von Schrift und Text

15.12.1 Die wichtigsten Eigenschaften zur Schrift- und Textgestaltung

15.12.2 Schrift gestalten: fett, kursiv, Kapitälchen und Zeichenabstand

15.12.3 Die Kurzschreibweise »font«

15.12.4 Text ausrichten und die erste Zeile einrücken

15.12.5 Schatten im Text: »text-shadow«

## 15.13 Auf einen Blick

# 16 Abstände gestalten mit dem Box-Modell

---

## 16.1 Das klassische Box-Modell für Blockboxen

## 16.2 Die Breite begrenzen: »min-width« und »max-width«

## 16.3 Der Abstand zum Rand: »padding«

16.3.1 Das »padding« für den Kopfbereich der Seite

16.3.2 Das »padding« für die Navigation und den Fußbereich

16.3.3 Das »padding« für den Inhaltsbereich

## 16.4 Rahmenlinien gestalten mit »border« und »border-radius«

16.4.1 Die Eigenschaften zum Gestalten von Rahmenlinien

16.4.2 Abgerundete Ecken mit »border-radius«

16.5 Blockboxen horizontal zentrieren mit »margin: auto«

16.6 Abstände zwischen den Elementen mit »margin«

16.7 »Collapsing Margins«: Vertikale Außenabstände kollabieren

16.7.1 Praktisch: Vertikale Außenabstände aufeinanderfolgender Elemente kollabieren

16.7.2 Problematisch: Außenabstände kollabieren nicht nur bei aufeinanderfolgenden Elementen

16.7.3 Beispiel: Ein Kopfbereich mit Überschrift und »Collapsing Margins«

16.7.4 »padding-top« für den Kopfbereich verhindert das Kollabieren der Außenabstände

16.7.5 Nützlich: Eine CSS-Regel zur Vermeidung von »Collapsing Margins«

16.8 Das intuitivere Box-Modell: »box-sizing: border-box«

16.8.1 Das Border-Box-Modell in der Übersicht

16.8.2 Das Border-Box-Modell aktivieren mit »box-sizing: border-box«

16.9 Das Box-Modell für Inline-Boxen

16.10 Inline-Block: Blockboxen, aber nebeneinander

16.11 Auf einen Blick

## 17 Boxen gestalten per CSS

---

17.1 Hintergrundgrafiken per CSS einbinden und gestalten

17.1.1 Hintergrundgrafiken einbinden: »background-image«

17.1.2 Hintergrundgrafiken wiederholen: »background-repeat«

17.1.3 Hintergrundgrafiken positionieren: »background-position«

17.1.4 Hintergrundgrafiken fixieren: »background-attachment«

17.1.5 Die Größe der Hintergrundgrafik definieren: »background-size«

17.1.6 Die Kurzschreibweise: »background«

17.1.7 Das Box-Modell und die dritte Dimension

17.2 Lineare Farbverläufe: »background-image« und »linear-gradient()«

17.3 Schattenboxen mit »box-shadow«

17.4 Gestalten mit dem Box-Modell: Zitate als Kundenstimmen

17.4.1 Das HTML: »section« und »blockquote«

- 17.4.2 Zitate gestalten mit den Box-Modell-Eigenschaften
- 17.5 »Call to Action«: Hyperlinks in Buttons verwandeln**
  - 17.5.1 Die Ausgangssituation: Zwei ganz normale Hyperlinks
  - 17.5.2 Schritt 1: Die grundlegende Gestaltung für beide Links
  - 17.5.3 Schritt 2: Die Unterschiede – primäre und sekundäre Buttons
  - 17.5.4 Schritt 3: Die Linkzustände der Buttons gestalten
  - 17.5.5 Schritt 4: Einen sanften Übergang mit »transition« hinzufügen
- 17.6 Boxen am Bildschirm ausblenden: »visually-hidden«**
  - 17.6.1 Schritt 1: Die Klasse »visually-hidden« erstellen
  - 17.6.2 Schritt 2: Den Elementen die Klasse »visually-hidden« zuweisen
- 17.7 Auf einen Blick**

## 18 Ordnung halten: Stylesheets organisieren

---

- 18.1 Stylesheets strukturieren mit Kommentaren**
  - 18.1.1 Der Kommentar am Anfang des Stylesheets
  - 18.1.2 Ein Stylesheet mit Kommentaren in Abschnitte unterteilen
- 18.2 Verschiedene Schreibweisen für CSS-Regeln**
  - 18.2.1 Übersichtlich und weit verbreitet: Auf jeder Zeile eine Deklaration
  - 18.2.2 Kurze Regeln: Alles in einer Zeile
  - 18.2.3 Übersichtlich: Mehrere Selektoren auf Zeilen verteilen
  - 18.2.4 Reihenfolge der Deklarationen: 1. Am Box-Modell orientieren
  - 18.2.5 Reihenfolge der Deklarationen: 2. Am Alphabet orientieren
- 18.3 CSS überprüfen mit dem CSS-Validator**
- 18.4 Modulbauweise: Ein zentrales Stylesheet erleichtert die Entwicklung**
  - 18.4.1 Während der Entwicklung: Modulbauweise mit mehreren Stylesheets
  - 18.4.2 Für die Live-Site: Alles wieder in einem Stylesheet vereinen
- 18.5 Die Stylesheets für die einzelnen Module erstellen**
  - 18.5.1 Schritt 1: Die einzelnen Stylesheets erstellen
  - 18.5.2 Schritt 2: Stylesheets mit @import in »style.css« einbinden
- 18.6 Das Modul »basis.css« ist das Fundament**
  - 18.6.1 Der Abschnitt »Globale Einstellungen für die gesamte Website«
  - 18.6.2 Der Abschnitt »Grundlegende Gestaltung von Schrift und Text«