

CSS-Klassen am Beispiel von List-Definitionen

Jobst-Hartmut Lüddecke

10. Oktober 2011

Zusammenfassung

Es werden verschiedene CSS-Definitionen für den Tag `` vorgestellt. Diese CSS-Definitionen sollten beim letzten Termin in Eigenarbeit unter www.selfhtml.org gefunden und mit den eigenen HTML-Seiten ausprobiert werden.

Erweitert werden diese Definitionen nun mit der Verwendung der HTML-Tags als **Klassen** durch die *Definitionen von Klassen* im *Cascading Style Sheet*. Weiterhin werden die Aufzählungszeichen der `ul`-Definitionen um die Verwendung von Bildern erweitert.

Inhaltsverzeichnis

1	Allgemeine Definition für den HTML-Befehl <code></code>	2
1.1	Geschachtelte Definition für den HTML-Befehl <code></code>	3
2	Zusätzliche Klassen-Definitionen für <code></code>	4
2.1	<code></code> mit Kreisen	4
2.2	<code></code> mit Quadraten	5
2.3	<code></code> mit Bild	6
3	Anwendung der Klassen in einer html-Seite	7
4	Möglichkeit für zusätzliche Abstände zwischen <i>list items</i>.	8
5	Aufgaben	8

Abbildungsverzeichnis

1	Allgemeine Definition von <code></code>	2
2	Geschachtelte Definition von <code></code>	3
3	Klassen-Definition von <code></code> mit <i>Kreis</i>	4
4	Klassen-Definition von <code></code> mit <i>Quadrat</i>	5
5	<code>ul</code> -Klasse mit einem Bild im <i>gif</i> -Format eines blauen Punktes als Markierungssymbol .	6
6	<code>ul</code> -Klasse mit einem Bild im <i>png</i> -Format eines grünen Punktes als Markierungssymbol	6
7	HTML-Ausschnitt: In der ersten Ebene die allgemeine Umgebung für <code></code> . In der 2. Stufe wird die Klasse <i>viereck</i> für <code></code> verwendet.	7
8	Jetzt die Klassen mit Bildern in verschiedener Schachtelung.	7
9	<code>li</code> -Definition mit 3mm oberen und unteren Rand.	8

1 Allgemeine Definition für den HTML-Befehl

```
ul {
  list-style-type:circle;
  list-style-position:outside;
  position:relative;
  font-family:"Garamond", serif;
  font-style:normal;
  font-variant:normal;
  font-weight:normal;
  font-size:14pt;
  line-height:21pt;
  letter-spacing:0pt;
  word-spacing:1pt;
}
```

Abbildung 1: Allgemeine Definition von

Dies ist eine allgemeine Definition für das Aussehen des html-Befehls . Die definierten Parameter im Einzelnen:

list-style-type: circle; das Markierungssymbol für einen Listeneintrag (*list item*) soll ein Kreis (*circle*) sein. Weitere Möglichkeiten sind ein Viereck (*square*), ein dicker Punkt (*disc*), oder gar keins (*none*).

list-style-position: outside; das Markierungssymbol ist vor der Blockformatierung des Listeneintrags. Ein *inside* würde einen Einzug des Listeneintrags bewirken und das Markierungssymbol in die Blockformatierung ziehen.

position: relative; die Position des Einrückens ist relativ zum aktuellen Wert. Nur so ist eine Schachtelung von Listen innerhalb übergeordneter Listen möglich. Bei einem absoluten Wert hat jede Liste einen genau definierten Abstand vom linken Rand. Allerdings kann man dies auch – wie in der letzten Lektion gelernt – mit entsprechenden Definitionen der Schachtelungen selbst in die Hand nehmen. Dann aber bitte vollständig für alle Kombinationsmöglichkeiten der Schachtelungen.

font... ist die Definition des Zeichensatzes und dessen Attribute innerhalb der Listenumgebung, wie bereits bei den Header-Definitionen besprochen.

letter-spacing: 0pt; zusätzlicher Abstand von 0 Punkten (also keiner) nach einzelnen Buchstaben. In diesem Fall spricht man auch von *kerning*.

word-spacing: 1pt; zusätzlicher Abstand von 1 Punkt (also ein wenig) zwischen einzelnen Worten.

1.1 Geschachtelte Definition für den HTML-Befehl

Definiert man zwei *Tags* im *Cascading Style Sheet* hintereinander, wie im Beispiel zweimal *ul*, dann gilt diese Definition für eine *unsorted list* innerhalb einer *unsorted list*, also die **2. Schachtelungstiefe**.

In der ersten Ebene wird hier also als Markierungssymbol ein *Kreis* verwendet und in einer 14pt-Schrift gesetzt.¹

Die 2. Ebene verwendet ein *Quadrat* und eine 12pt-Schrift.² Entsprechend kann man es für eine 3. und 4. Ebene weiter entwickeln. Dies sollten Sie auch tun.

```
ul ul {
  list-style-type: square;
  list-style-position: outside;
  position: relative;
  font-family: "Garamond", serif;
  font-style: normal;
  font-variant: normal;
  font-weight: normal;
  font-size: 12pt;
  line-height: 18pt;
  letter-spacing: 0pt;
  word-spacing: 0pt;
}
```

Abbildung 2: Geschachtelte Definition von

¹siehe Abbildung 1 auf Seite 2

²siehe Abbildung 2 auf Seite 3

2 Zusätzliche Klassen-Definitionen für

Möchte man unterschiedliche Typen eines bestimmten *tags* verwenden, kann man sich selbst neue *tags* definieren, oder bestimmte *Klassen* von einem bekannten und in HTML vorgesehenen *tag* erstellen.

Der Vorteil eigene *Klassen* eines in HTML normierten *tags* zu definieren, liegt darin, dass bei einem defekten *Cascading Style sheet* immerhin noch der *tag* funktioniert.

Selbst definierte *tags* führen zu einer **eigenen HTML-Sprache**, die nicht mehr kompatibel ist, für die es keine Lehrbücher gibt, die von anderen Personen schwer zu warten ist und die im Fehlerfall *völlig in die Hose geht*.

Bei der **Definition einer Klasse** wird der bekannte *tag* mit einer *Extension* – anders gesagt einem Punkt und einem weiteren Begriff, **dem Schlüssel oder Namen der Klasse** – versehen.

Welche Version des *tags* nun Verwendung finden soll, wird dann in der HTML-Seite durch die Option *class* im *tag* näher bestimmt. Im Fehlerfall funktioniert immerhin noch der *normale tag*.

2.1 mit Kreisen

```
ul.kreis {
    list-style-type:circle;
    list-style-position:outside;
    position:relative;
    font-family:"Garamond", serif;
    font-style:normal;
    font-variant:normal;
    font-weight:normal;
    font-size:14pt;
    line-height:21pt;
    letter-spacing:0pt;
    word-spacing:1pt;
}
```

Abbildung 3: Klassen-Definition von mit *Kreis*

Diese Definition entspricht vom Inhalt der o.a. allgemeinen Definition von , in diesem Fall gilt sie aber nur für **der Klasse** *kreis*.

2.2 mit Quadraten

```
ul.viereck {
    list-style-type:square;
    list-style-position:outside;
    position:relative;
    font-family:"Garamond", serif;
    font-style:normal;
    font-variant:normal;
    font-weight:normal;
    font-size:14pt;
    line-height:21pt;
    letter-spacing:0pt;
    word-spacing:1pt;
}
```

Abbildung 4: Klassen-Definition von mit *Quadrat*

Dies ist die Definition einer weiteren Klasse, die gleichberechtigt neben der o.a. Klasse *kreis* stehen kann. In diesem Fall der Klasse *viereck* ist als Markierungssymbol ein kleines Quadrat definiert. Der Rest der Definition ist gleich. Damit kann man durch die Wahl der *Klasse* die Markierungssymbole gezielt einsetzen.

2.3 mit Bild

```
ul.blau {
    list-style-image:url(blau.gif);
    list-style-position:outside;
    font-family:"Garamond", serif;
    font-style:normal;
    font-variant:normal;
    font-weight:normal;
    font-size:14pt;
    line-height:21pt;
    letter-spacing:0pt;
    word-spacing:1pt;
    position:relative;
}
```

Abbildung 5: ul-Klasse mit einem Bild im *gif*-Format eines blauen Punktes als Markierungssymbol

In diesem Fall wird statt **list-style-type** mit Hilfe von **list-style-image** ein Bild (in diesem Fall im GIF-Format) als Markierungssymbol verwendet. Hier handelt es sich um ein *blaues Knöpfchen zu finden unter* `/public/lueddecke/IuK/icons/` bzw. `K:\lueddecke\IuK\icons` .

Der Pfad des Bildes ist nicht näher spezifiziert, damit muss sich dass Bild im **gleichen Verzeichnis**, wie die html-Seite und die css-Seite befinden. Andere Adressierungsarten werden in einer späteren Lektion folgen.

```
ul.gruen {
    list-style-image:url(gruen.png);
    list-style-position:outside;
    font-family:"Garamond", serif;
    font-style:normal;
    font-variant:normal;
    font-weight:normal;
    font-size:14pt;
    line-height:21pt;
    letter-spacing:0pt;
    word-spacing:1pt;
    position:relative;
}
```

Abbildung 6: ul-Klasse mit einem Bild im *png*-Format eines grünen Punktes als Markierungssymbol

Es muss nicht immer ein *GIF* sein, im *png*-Format geht es genauso gut.

Achtung! Es hat sich gezeigt, dass die Verwendung von *list-style-image* bei einer normalen *ul-Definition* diese auch bei den untergeordneten Ebenen als höherwertig betrachtet wird, als eine Definition mit *list-style-type*. Es wird also das Bild der *ul-Definition* einem Symbol in einer *ul-ul-Definition* der Vorrang gegeben. Man kann also nur mit einem anderen Bild in der *ul-ul-Definition* entgegenwirken. Entsprechendes gilt für weitere tiefere Ebenen.

3 Anwendung der Klassen in einer html-Seite

Wie bereits mehrfach erwähnt, wird dem *tag* die Option *class* und der *Name der Klasse* in amerikanischen Anführungsstrichen³ hinzugefügt. All dies geschieht zwischen den spitzen Klammern. Im unteren Beispiel wird einmal der *tag* `` und einmal der *tag mit Option* `<ul class="viereck" >` verwendet. Bitte achten Sie auf die Feinheiten.

```
<ul>
<li>The quick brown fox jumps over the lazy dog</li>
<li>The quick brown fox jumps over the lazy dog</li>
<li>The quick brown fox jumps over the lazy dog</li>
  <ul class="viereck">
    <li>Hier sind nun kleine Quadrate als Markierer</li>
    <li>Noch ein Eintrag mit Quadrat</li>
  </ul>
<li>The quick brown fox jumps over the lazy dog</li>
<li>The quick brown fox jumps over the lazy dog</li>
</ul>
```

Abbildung 7: HTML-Ausschnitt: In der ersten Ebene die allgemeine Umgebung für ``. In der 2. Stufe wird die **Klasse** *viereck* für `` verwendet.

```
<ul class="blau">
<li>Jetzt wird es bunt, wenn alles gut gegangen ist, steht hier ein
  blaues Kn&ouml;pfchen am Anfang</li>
  <ul class="gruen">
    <li>Wenn alles gut gegangen ist, steht hier ein Eintrag eine Ebene
      tiefer mit einem gr&uuml;nen Kn&ouml;pfchen am Anfang</li>
    <li>Noch einmal mit gr&uuml;nen Kn&ouml;pfchen.</li>
  </ul>
  <ul class="gelb">
    <li>Nun ein Eintrag mit gelbem Kn&ouml;pfchen.</li>
  </ul>
</ul>
```

Abbildung 8: Jetzt die **Klassen** mit Bildern in verschiedener Schachtelung.

³„Deutsche“ und «französische» Anführungszeichen sehen etwas anders aus.

4 Möglichkeit für zusätzliche Abstände zwischen *list items*.

Immer wieder tritt der Wunsch auf zwischen den einzelnen Listeneinträgen einen zusätzlichen Abstand zu definieren. Für Zeilenabstände habe wir dies schon mit einer Zeilenhöhe gesehen. In diesem Fall bietet sich eine *margin*-Definition wie bei der *body-Definition* im *Cascading Style Sheet* an. **Unterlassen** Sie bitte unbedingt irgendwelche **Break**-Konstruktionen in der *html*-Seite! In fast allen Fällen erreichen Sie das gewünschte Erscheinungsbild sehr viel besser mit neuen Paragraphen, Tabellen, oder Rändern.

Selbstverständlich kann man für seine *List-Items* auch noch weitere Definitionen für den *Font*, der *Size*, der *Color* usw. hinzufügen und so weiter seinem individuellen Erscheinungsbild der Seite ein *Fein-tuning* angedeihen lassen.

```
li {
  margin-top:3mm;
  margin-bottom:3mm;
}
```

Abbildung 9: li-Definition mit 3mm oberen und unteren Rand.

5 Aufgaben

- Definieren Sie sich verschiedene Klassen (mit Symbolen und Bildern) für die Listenumgebung ** und probieren Sie diese aus.
- Erarbeiten Sie sich die Möglichkeiten für die Listenumgebung *DL*.