

## Übung 3 – Lösungshinweise

(Zu den benötigten HTML- und CSS-Anweisungen im Einzelnen siehe beispielsweise <http://selfhtml.teamone.de>)

Aufgabe 4:

- Mittels HTML-Attribut, z. B. `<table border="1">`
- Wenn keine expliziten Angaben zur Darstellung gemacht werden, können die Seiten in verschiedenen Browsern unterschiedlich angezeigt werden (so z. B. auch die Überschrift).

Aufgabe 5:

- Mittels HTML-Attribut nicht möglich. Mittels CSS:  
`<table style="border-style:solid; border-width:1">`  
An dieser Stelle bietet CSS mehr Möglichkeiten als HTML. Wichtigere Gründe jedoch für die Verwendung von CSS sind:
  - die Trennung von Inhalt und Darstellung und
  - die Möglichkeit, in einem einzigen Stylesheet zentral Formatierungen für beliebig viele Webseiten vornehmen zu können.
- Obige Einbindung von CSS bewirkt jedoch, dass Formatierung nur für die Tabelle selbst (d. h. deren Rahmen) nicht jedoch für die darin enthaltenen Zellen gilt. Falls letzteres gewünscht wird, könnte man jeder einzelnen Zelle analog mittels ‚style‘ den Rahmenstil zuweisen. Dies wäre sehr umständlich, weshalb es zweckmäßiger ist, stattdessen ein separates Stylesheet zu erstellen, in dem einmalig und zentral für alle Tabellen und Zellen der Rahmen definiert wird.

Aufgabe 6:

Für die zweite Tabelle könnte man eine selbstdefinierte Stilklasse verwenden, z. B. im

Stylesheet: `.2ndTableStyle {border-style:dotted}`

und entsprechend in der HTML-Datei: `<table class="2ndTableStyle">`.

Um die Zellen innerhalb dieser Tabelle zu formatieren, verwendet man kontextuelle

Selektoren; z. B. im Stylesheet: `.2ndTableStyle td {border-style:dotted}`

gilt für alle Tabellenzellen (td) die innerhalb eines Elementes vorkommen, das mit der selbstdefinierten Klasse ‚2ndTableStyle‘ formatiert ist (in diesem Fall also die Tabelle).

Analog ist dies für die Zellen der Tabellenüberschrift (th) vorzunehmen.

Aufgabe 8: Analog zu Aufgabe 6 definiert man mittels kontextueller Selektoren das Format für Listen, die sich innerhalb einer Tabelle befinden, z. B.:

```
table ul { ... }
```

Aufgabe 10: In diesem Fall benötigt man (zumindest für manche Browser) ein geschütztes Leezeichen (`&nbsp;`), damit die Tabellenzeile nicht als überflüssig erachtet und weggelassen wird.

Aufgabe 12: Siehe Folie 3-80 der Vorlesung.

Aufgabe 13: Zu jeder Webseite kann auch der zugehörige Quelltext betrachtet werden. Dazu ruft man entweder im Browser einen entsprechenden Menüpunkt auf (z. B. ‚Quelltext

anzeigen' im Menü ‚Ansicht‘), oder man speichert die Webseite und öffnet sie anschließend mit einem Texteditor.

=> man sollte sich bewusst sein, dass man die Möglichkeit hat, den Quellcode fremder guter Seiten zu betrachten, jedoch auch, dass jeder den Quellcode der eigenen Seiten betrachten kann.