**HTML (**[**http://www.poenitz-net.de/Informatik/3.HTML/3.HTML.html**](http://www.poenitz-net.de/Informatik/3.HTML/3.HTML.html)**)**

[**1. Grundbegriffe**](http://www.poenitz-net.de/Informatik/3.HTML/3.1.Grundbegriffe.html)

[**2. Grundelemente.**](http://www.poenitz-net.de/Informatik/3.HTML/3.2.Grundelemente.html)

 [**Zeichenformatierung**](http://www.poenitz-net.de/Informatik/3.HTML/3.3.Zeichenformatierung.html)

[**4. Absatzformatierung**](http://www.poenitz-net.de/Informatik/3.HTML/3.4.Absatzformatierung.html)

[**5. Links und Grafiken**](http://www.poenitz-net.de/Informatik/3.HTML/3.5.Links%20und%20Grafiken.html)

[**6. Listen**](http://www.poenitz-net.de/Informatik/3.HTML/3.6.Listen.html)

[**7. Tabellen**](http://www.poenitz-net.de/Informatik/3.HTML/3.7.Tabellen.html)

## 1. Grundbegriffe

**HTML = Hypertext Markup Language** ist ein Formatierungscode, (Hypertext Language), der einem unformatierten Text beigefügt wird, um einem geeigneten Textprogramm (Browser, s.u.) mitzuteilen, wie der Text formatiert werden soll. Da die Formatierungsanweiseungen (tags) sehr viel weniger Speicherplatz beanspruchen, als der fertig formatierte Text, wird durch diese Verschlüsselung das Verschicken des Textes über das Internet wesentlich beschleunigt. Als Besonderheit enthält der HTML-Code die Möglichkeit, Markups (Links) einzufüen, die es dem Leser ermöglichen, mittels Mausklick auf eine andere Datei beliebiger Gestalt (Bild, Text, Programm, u.s.w.) zu springen.

**CSS = cascading style sheets** ist ein Satz von Formatierungsbefehlen, der die zentrale Formatierung des Dokumentes über Formatvorlagen ähnlich wie in Microsoft Word ermöglicht.

**Browser** wie z.B. der Microsoft Internet Explorer oder Mozilla Firefox fordern mittels http die vom Client gewüschten Dateien vom Server an. Sie entschlüsseln außerdem den HTML-Code des gsendeten Dokuments und fomatieren den Text fuuml;r die Ausgabe auf dem Bildschirm oder uuml;ber den Drucker.

**Server** sind Computer mit besonders großer Speicherkapazität, die über eine Standleitung 24 h am Tag mit dem Internet verbunden sind. e-mails, Dateien und HTML-Dokumente, die auf einem Server gespeichert sind, können zu jeder Tages- und Nachtzeit von jedem Client mit Internetzugang und entsprechender Berechtigung abgerufen werden.

**Clients** nennt man die Computer zu Hause oder im Büro, die nur zeitweise mit dem Internet verbunden sind. Clients senden (upload) und empfangen (download) e-mails über SMTP, Dateien über ftp und HTML-Dokumente über http zum bzw. vom Server.

**Provider** nennt man alle Unternehmen, die Webserver betreiben. Möchte man eine Datei (Bild, Text oder Programm) über das Internet veröffentlichen, so mietet man für ein geringes Entgelt (1 - 10 Euro pro Monat) bei solch einem Provider einen bestimmtem Speicherabschnitt (z.B. 500 MB) auf einem Server. Auf diesen reservierten Speicher kann man die gewüschten Dateien mit einem ftp-Programm (z.B. filezilla) übertragen und damit f%uuml;r Anwender (Clients) auf der ganzen Welt zugälich machen.

**http = hypertext transfer protocol** ist die Sprache, mit der sich Server und Client beim Austausch von HTML- Dokumenten verständigen.Sie ermöglicht insbesondere den Aufruf mehrerer Seiten gleichzeitig beim Anklicken von Hyperlinks.

**ftp = file transfer protocol** ist die Sprache, mit der sich Server und Client beim Austausch von Dateien verständigen. ftp kann mit Hyperlinks nichts anfangen und ist daher für Aufrufe von Webseiten ungeeignet. Es ist aber deutlich schneller als http und wird daher zum Hochladen von Webseiten gern genutzt.

**smtp = standard mail transfer protocol** ist die Sprache, mit der sich Server und Client beim Austausch von e-mails verständigen.

**GIF = Graphics Interchange Format** ist ein Format für Grafikdateien, dessen Kompressionsalgorithmus vor allem an den Farben spart. Geradlinige Kanten werden dagegen auch bei hoher Kompression genau wiedergegeben. GIF ist daher weniger für Fotos und mehr für Zeichnungen geeignet.

**JPEG = Joint Photographic Experts Group** ist ein vor allem für Fotos sehr verbreitetes Grafikformat. Es gibt trotz Kompression Farben wirklichkeitsgetreu wieder, ist aber für Grafiken und Zeichnung mit scharfen geradlinigen Kanten (z.B. Comics) weniger geeignet.

## 2. Grundelemente

**Beispiel eines einfachen HTML-Dokumentes:**

DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

<html>

 <head>

 <title>listing 3.2</title>

 </head>

 <body>

 <p>Hello World!</p>

 </body>

</html>

**Dokumententyp-Definition**: Zu Beginn des HTML-Dokumentes muss dem Browser mitgeteilt werden, um welche Version von HTML es sich handelt. Wir verwenden die folgende Kopfzeile:

<DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">.

Wir verwenden also die öffentlich (PUBLIC) verfügbare HTML-Version des World-Wide-Web-Konsortiums (W3C). Die von und benutzte Dokumententyp-Definition (DTD) Version ist HTML 4.01 Transitional in der Sprache Englisch (EN). Achtung: Die Sprachangabe bezieht sich nicht auf den Text, sondern auf die Formatierungsangaben!

|  |  |
| --- | --- |
| **html-Element**: | Das Element <html>...</html> schließt alle html-Anweiseungen ein. Nur Anweisungen, die innerhalb dieses Elementes liegen, werden als html interpretiert.  |
| **head-Element**: | Das Element <head>...</head> schließt alle html-Anweisungen ein, die sich auf den **gesamten** folgenden Textkörper beziehen. Hier finden sich vor allem Verweise auf Dateien für Formatierungsvorlagen, Scripte und Applets, die im Text verwendet werden.  |
| **title-Element**: | Das Element <title>...</title> schließt den Speichernamen des Dokumentes ein, der in der blauen Kopfzeile des Browserfensters erscheint.  |
| **body-Element**: | Das Element <body>...</body> schließt den kompletten Textkörper ein.  |
| **p-Element**: | Das Element <p>...</p> schließt einen Absatz (Paragraphen) ein. |

### Aufgabe 2

1. Öffne das Programm Editor unter Start/Programme/Zubehör.
2. Schreibe das listing 3.2 in das Textfenster
3. Speichere den Text unter dem Namen listing3.2.html. Dazu muss im Dialogfenster die Option Textdateien durch die Option alle Dateien ersetzt werden, sonst wird automatisch die für uns unbrauchbare Endung .txt angehängt.
4. Beende Editor und öffne anschließend die gespeicherte Datei listing3.2.html wieder durch Doppelklick. listing3.2.html wird nun durch den Browser geöffnet. Dieser zeigt zunächst nur den Text Hello World! an.
5. Wähle im Browser die Option Ansicht/Quelltext. Der Browser startet selbstä:ndig den Editor und zeigt das vollständige Listing an. Ersetze im Listing den Text Hello World! durch Hallo Ihr! und wähle die Option Datei/Speichern.
6. Wechsle wieder zum Browserfenster und wähle die Option Ansicht/Aktualisieren. Der Browser zeigt jetzt den geänderten Text Hallo Ihr! an.

## 3. Zeichenformatierung

### 3.1 Sonderzeichen

**Sonderzeichen, Umlaute und die für Formatierungsanweisungen reservierten Zeichen** werden nach der folgenden Tabelle verschlüsselt:

|  |  |
| --- | --- |
| Sonderzeichen | HTML-Code |
| " | &quot; |
| & | &amp; |
| < | &lt; |
| > | &gt; |
| Leerzeichen | &nbsp; |
| ß | &szlig; |
| ä | &auml; |
| ö | &ouml; |
| ü | &uuml; |

**Beispiel:**

 <DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

 <html>

 <head>

 <title>listing 3.3.1</title>

 </head>

 <body>

 <p>Auf den h&ouml;chsten &Auml;sten sa&szlig;en gro&szlig;e Affen und

 beschimpften mich mit uns&auml;glichen Ausdr&uuml;cken wie z.B. &quot;&amp;

 &amp;&amp;&amp;&quot; .</p>

 </body>

 </html>

### 3.2 Überschriften

**Überschriften** werden nach der folgenden Tabelle verschlüsselt:

|  |  |
| --- | --- |
| Eingabe | Darstellung |
| <h1>...</h1> | **Überschrift 1** |
| <h2>...</h2> | **Überschrift 2** |
| <h3>...</h3> | **Überschrift 3** |
| <h4>...</h4> | **Überschrift 4** |

**Beispiel:**

<DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

 <html>

 <head>

 <title>listing 3.3.2</title>

 </head>

 <body>

 <h1>&Uuml;berschrift&nbsp;1</h1>

 <h2>&Uuml;berschrift&nbsp;2</h2>

 <h3>&Uuml;berschrift&nbsp;3</h3>

 <h4>&Uuml;berschrift&nbsp;4</h4>

 </body>

 </html>

### Aufgabe 3.1

Formuliere den Quelltext für das folgende HTML-Dokument und speichere es unter dem Titel Aufgabe 3.3.1.



### 3.3 Schriftart und -grösse

**Schriftart und -grösse** einzelner Buchstaben und Wörter lassen sich am einfachsten mit Hilfe des span -Elementes und der font-Eigenschaft verändern. Der allgemeine Aufbau ist folgendermaßen:

<span style="font:font-style font-weight font-size font-family;">...</span>

Die möglichen Werte dieser Eigenschaften sind in der folgenden Tabelle aufgelistet. Die Werte werden mit einem Doppelpunkt an die entsprechende Eigenschaft gehängt und durch Leerzeichen voneinander getrennt. Den Abschluss bildet immer ein Semikolon: Eigenschaft:Wert Wert Wert; Die Eigenschaften können die folgenden Werte annehmen:

|  |  |
| --- | --- |
| Eigenschaft | mögliche Werte |
| font-style | italic, normal |
| font-weight | bold, normal |
| font-size | 8pt, 10pt, 12pt, usw. |
| font-family | Times New Roman, Arial, Courier, usw. |

**Beispiel:**

 <DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

 <html>

 <head>

 <title>listing 3.3.3</title>

 </head>

 <body>

 <p>

 Dieser Text enthät <span style="font:italic 12pt Times New Roman;">

 kursive</span>, <span style="font:bold 12pt Times New Roman;">

 fettgedruckte</span>und <span style="font:normal 14pt Times New Roman;

 ">vergrößerte</span> Wörter. <span style="font:normal 12pt Arial;

 ">Dieser Satz ist in Arial geschrieben.</span> Alles lässt sich kombinieren

 wie z.B. <span style="font:bold 12pt Courier;">fettgedruckter Courier-Zeichensatz.

 </span>

 </p>

 </body>

 </html>

### 3.4 Farben

**Farben** werden mit den Eigenschaften color und background-color definiert. Als Werte kann man die RGB (Rot/Grün/Blau)-Anteile in Prozent einsetzen. color:rgb(50%,0%,0%) bedeutet 50% rot, 0% grün und 0% blau. Grautöne erhält man durch gleiche Gewichtung aller Farbanteile. color:rgb(0%,0%,0%) bedeutet schwarz, color:rgb(50%,50%,50%) bedeutet mittelgrau und color:rgb(100%,100%,100%) bedeutet weiß. color und background-color müssen mit Hilfe des style-Elementes in das span-Element integriert werden:

<span style="color:rgb(x%,y%,z%); background-color:rgb(u%,v%,w%);">...</span>

**Beispiel:**

 <DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

 <html>

 <head>

 <title>listing 3.3.4</title>

 </head>

 <body>

 <p>

 Dieser Text enthät einen <span style="color:rgb(0%,0%,100%);

 background-color:rgb(100%,100%,0%);">blauen Teil auf gelbem Grund</span>

 und einen <span style="color:rgb(100%,100%,0%); background-color:rgb(0%,0%,100%);

 ">gelben Teil auf blauen Grund</span>.

 </p>

 </body>

 </html>

### 3.5 Rahmen

**Rahmen** werden mit der Eigenschaft border:border-width border-style border-color; definiert:

<span style="border:border-width border-style border-color;">...</span>

Die möglichen Werte für border-width, border-style und border-color sind in der folgenden Tabelle aufgelistet.

|  |  |
| --- | --- |
| Eigenschaft | mögliche Werte |
| border-width | 1px, 2px, 3px, usw. |
| border-style | double, solid, dashed, dotted |
| border-color | rgb(x%,y%,z%) |

**Beispiel:**

 <DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

 <html>

 <head>

 <title>listing 3.3.5</title>

 </head>

 <body>

 <p>

 Dieser Text enthät einen <span style="color:rgb(0%,0%,100%); background-color:rgb(100%,100%,0%); border:2px double rgb(0%, 100%, 0%)">

 blauen Teil auf gelbem Grund mit grünem doppelten Rahmen (Würg)</span> und einen <span style="color:rgb(100%,100%,0%);

 background-color:rgb(0%,0%,100%);border:2px dashed rgb(100%, 0%, 0%)">gelben Teil auf blauen Grund mit rotem gestrichelten Rahmen

 (auch nicht besser)</span>.

 </p>

 </body>

 </html>

### Aufgabe 3.2

Formuliere den Quelltext für das fogende HTML-Dokument und speichere es unter dem Titel Aufgabe 1.3.2.

## 3.4. Absatzformatierung

### 3.4.1 Zeichenformatierung

**Zeichen** in Absätzen und Überschriften lassen sich mit den schon bekannten Eigenschaften font, color, background-color und border formatieren.

**Beispiel:**

 <DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

 <html>

 <head>

 <title>listing 3.4.1</title>

 </head>

 <body>

 <p style="font:italic 14pt Arial; color:rgb(0%,100%,100%); background-color:rgb(50%,0%,20%);

 border:solid 3px rgb(100%,0%,0%);">

 Dieser Absatz ist in kursiv 14pt Arial türkis auf pinkfarbenem

 Hintergrund mit rotem Rahmen dargestellt.

 </p>

 </body>

 </html>

### 3.4.2 Ausrichtung

Die **Ausrichtung** wird durch die Eigenschaft text-align mit den möglichen Werten left, center, right und justify (Blocksatz) gesteuert. Ein Anwendungsbeispiel erscheint im folgenden Abschnitt.

### 3.4.3 Zentrale Formatierung

Der eigentliche Sinn der Absatzformatierung besteht in der Einrichtung von **Formatvorlagen** für Absätze, Überschriften und Tabellen im head-Element.

**Beispiel:**

 <DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

 <html>

 <head>

 <title>listing 3.4.2</title>

 <style type="text/css">

 p { font:normal 14pt Arial;

 color:rgb(100%,100%,100%);

 background-color:rgb(0%,0%,100%); text-align:justify; }

 h1 { font:bold 16pt Arial;

 color:rgb(100%,100%,0%);

 background-color:rgb(0%,0%,100%);

 text-align:left; }

 </style>

 </head>

 <body>

 <h1>

 Alle Überschriften in diesem Text sind nun fettgedruckt 16 pt Arial

 gelb auf blauem Grund links ausgerichtet.

 </h1>

 <p>

 Alle Absätze in diesem Text sind nun normal 14 pt Arial

 weiß auf blauem Grund im Blocksatz.

 Die Hintergrundfarbe für das gesamte Dokument wird man aber

 normalerweise im BODY-Element festlegen.

 </p>

 </body>

 </html>

**Erklärung:** Das style-Element <style type="text/css">...</style> schließt die Formatvorlagen ein. Eine Formatvorlage besteht aus dem jeweiligen Textelement mit seinen Eigenschaften in geschweiften Klammern.

### Aufgabe 3.4.1

Formuliere den Quelltext für das folgende HTML-Dokument und speichere es unter dem Titel Aufgabe 3.4.1.



## 3.5. Links und Grafiken

**Links** werden mit Hilfe des a-Elementes (für anchor = Anker) und der href-Eigenschaft (für hyper reference = übergeordneter Bezug) eingefügt. Als Wert wird die vollständige Adresse des Zieldokumentes eingetragen. **Achtung!** Leerzeichen in der Adresse müssen durch die Zeichenfolge %20 ersetzt werden!

**Beispiel:**

 <DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

 <html>

 <head>

 <title>listing 3.5.1</title>

 </head>

 <body>

 <p>

 Dieser Absatz enthält einen Link auf die

 <a href="http://www.poenitz-net.de/Mathematik/Mathematik.htm">

 Startseite Mathematik<a> von poenitz-net.de.

 </p>

 </body>

 </html>

**e-mails** werden durch einen Link gestartet, in dem die entsprechende e-mail-Adresse als Wert der mailto-Eigenschaft eingetragen wird. Der Bezug der e-mail kann über die subject-Eigenschaft vorgegeben werden.

**Beispiel:**

 <DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

 <html>

 <head>

 <title>listing 3.5.2</title>

 </head>

 <body>

 <p>

 Klickt man auf den Link

 <a href="mailto:Arne.Poenitz@gmx.de?subject=HTML">

 E-mail an mich<a>, so startet das e-mail-Programm und trägt

 automatisch die Adresse Arne.Poenitz@gmx.de

 sowie den Betreff HTML ein.

 </p>

 </body>

 </html>

**Grafiken** werden mit Hilfe des img-Elementes (image = Bild) eingebunden. Das img-Element kann zu Beginn eines Absatzes (der folgenden Text umfließt dann die Grafik) oder in einen extra Absatz (die Grafik steht dann zwischen den Absätzen) eingefügt werden. Die folgenden Eigenschaften müssen bzw. können definiert werden:

|  |  |
| --- | --- |
| Eigenschaft | Bedeutung und mögliche Werte |
| src="..." | vollständige Quelladresse (source = Quelle) |
| alt="..." | Alternativtext, falls die Grafik nicht angezeigt werden kann. |
| align="..." | Ausrichtung der allein stehenden Grafik left, center oder right |
| style="float:..." | Ausrichtung der Grafik im umfließ:enden Text left und right |
| height | Dehnung oder Stauchung der Grafik auf die vorgegebenen Höhe in px oder % (optional) |
| width | Dehnung oder Stauchung der Grafik auf die vorgegebenen Breite in px oder % (optional) |

**Beispiel:**

 <DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

 <html>

 <head>

 <title>listing 3.5.3</title>

 </head>

 <body>

 <p align="justify">

 <img src="stavanger.jpg" alt="Stavanger"

 width="30%" heigth="30%" style="float:right" >

 Dieser Absatz umfließt im Blocksatz ein rechts ausgerichtetes

 Bild der norwegischen Hafenstadt Stavanger , das auf 30 % der

 Bildschirmgrösse gestaucht wurde. Wenn nur der Dateiname des

 Bildes ohne Adresse angegeben wurde, geht der Browser davon aus,

 dass sich das Bild im gleichen Ordner befindet wie das Hauptdokument

 selbst. Dies entspricht auch der Regel, dass man einzubindende

 Bilder nicht auf irgendeiner weit entfernten Webseite, sondern

 im gleichen Ordner speichert. In diesem Fall wird abweichend von der

 Regel tatsächlich auf ein Bild zugegriffen, das auf der Seite

 <tt> http://www.poenitz-net.de</tt> liegt.

 Der Alternativtext sollte die eigentlich darzustellende Grafik

 möglichst unmissverständlich beschreiben.

 </p>

 <p>

 Das gleiche Bild noch einmal am Ende des Absatzes. Diesmal ist das

 Bild selbst horizontal zentriert ausgerichtet. Beachte, dass die Anweiseung

 <tt>align="center"</tt> nicht im <tt>

 img</tt>-Element, sondern im übergeordneten <tt>p</tt>-Element steht!

 </p>

 <p align="center">

 <img src="stavanger.jpg" alt="Stavanger"

 width="30%" heigth="30%" >

 </body>

 </html>

**Hinweis**: Das im listing 3.5.1 verwendetete <tt>...</tt>-Element (tt = true type) setzt den Inhalt in der für listings üblichen Schriftart Courier

### Aufgabe 3.5.1

Formuliere den Quelltext für das folgende HTML-Dokument und speichere es unter dem Titel Aufgabe 3.5.1 Das Bild findet sich unter der Adresse http://www.poenitz-net.de/Informatik/3.HTML/sailor.jpg.



## 3.6. Listen

**Geordnete Listen** werden durch das Element <ol>...</ol> (ordered list) eingeschlossen. Jedes Element der Liste wird wiederum durch <li>...</li> (list item) begrenzt

**Beispiel:**

 <DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

 <html>

 <head>

 <title>listing 3.4</title>

 </head>

 <body>

 <ol>

 <li>Januar</li>

 <li>Februar</li>

 <li>:März</li>

 </ol>

 </body>

 </html>

**Ungeordnete Listen** (Aufzählungszeichen) werden durch das Element <ul>...</ul> (unordered list) eingeschlossen. Jedes Element der Liste wird wie oben durch <li>...</li> begrenzt

**Beispiel:**

 <DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

 <html>

 <head>

 <title>listing 3.5</title>

 </head>

 <body>

 <ul>

 <li>Jacke</li>

 <li>Hose</li>

 <li>Schuhe</li>

 </ul>

 </body>

 </html>

### Aufgabe 3.6.1

Formuliere den Quelltext für das folgende HTML-Dokument und speichere es unter dem Titel Aufgabe 3.6.1

.

## 3.7.Tabellen

**Tabellen** werden durch das Element <table>...</table> eingeschlossen.
Jede Zeile wird durch <tr>...</tr> (table row) eingeschlossen.
Jede Zelle wird durch <td>...</td> (table data) eingeschlossen.

Gibt man keine Eigenschaften für die Formatierung an, so wird die Tabelle **ohne Rahmen (!)** mit Ausrichtung links unten, Hintergrundfarbe weiß und minimaler an den Inhalt angepassten Größe gezeichnet.

**Beispiel einer Tabelle ohne Formatierung in Standardausgabe:**

 <DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

 <html>

 <head>

 <title>listing 3.7.1</title>

 </head>

 <body>

 <table>

 <tr>

 <td>Deutsch</td>

 <td>Italienisch</td>

 </tr>

 <tr>

 <td>Januar</td>

 <td>Gennaio</td>

 </tr>

 <tr>

 <td>Februar</td>

 <td>Febbraio</td>

 </tr>

 </table>

 </body>

 </html>

Die Formatierung der einzelnen Zellen steuert man am einfachsten zentral durch Formatvorlagen im head-Element. Alle Formatvorlagen werden durch das style-Element <style type="text/css">...</style> eingeschlossen. Eine Formatvorlage besteht aus dem jeweiligen Tabellenelement mit seinen Eigenschaften in geschweiften Klammern.

<style type="text/css">
table {align; }
td { border; border-collapse; background-color; font; text-align; vertical-align; padding }

Die möglichen Werte dieser Eigenschaften sind in den folgenden Tabellen aufgelistet. Die Werte werden mit einem Doppelpunkt an die entsprechende Eigenschaft gehängt und durch Leerzeichen voneinander getrennt. Den Abschluss bildet immer ein Semikolon: Eigenschaft:Wert Wert Wert;. Wichtig und relativ neu ist die Eigenschaft border-collapse. Der wert collapse unterdrückt die in HTML früher unvermeidbare doppelte Zellenbegrenzung und erzeugt eine Tabelle im gewohnten WORD-Stil. Mit separate erhält jede Zelle einen extra Rahmen.

|  |  |
| --- | --- |
| Eigenschaft | mögliche Werte |
| align | left, center, right |
| border | border-style border-width border-color |
| border-collapse | separate, collapse |
| background-color | rgb(x%,y%,z%) |
| font | font-style font-weight font-size font-family |
| text-align | left, center, right, justify |
| vertical-align | top, middle, bottom |
| padding | 2px, 3px, 4px, usw. |

Die möglichen Werte der Untereigenschaften von border sind:

|  |  |
| --- | --- |
| Eigenschaft | mögliche Werte |
| border-style | double, solid, dashed, dotted |
| border-width | 2px, 3px, 4px, usw. |
| border-color | rgb(x%,y%,z%) |

Die möglichen Werte der Untereigenschaften von font sind:

|  |  |
| --- | --- |
| Eigenschaft | mögliche Werte |
| font-style | italic, normal |
| font-weight | bold, normal |
| font-size | 8pt, 10pt, 12pt, usw. |
| font-family | Times New Roman, Arial, Courier, usw. |

**Beispiel einer formatierten Tabelle:**

 <DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

 <html>

 <head>

 <title>listing 3.7.2</title>

 <style type="text/css">

 table {border-collapse:collapse;

 background-color:rgb(100%,100%,0%);}

 td { border:solid 2px rgb(0%,0%,0%);

 text-align:center;

 vertical-align:middle }

 </style>

 </head>

 <body>

 <table>

 <tr>

 <td>Deutsch</td>

 <td>Italienisch</td>

 </tr>

 <tr>

 <td>Januar</td>

 <td>Gennaio</td>

 </tr>

 <tr>

 <td>Februar</td>

 <td>Febbraio</td>

 </tr>

 </table>

 </body>

 </html>

### Aufgabe 3.7.1

Formuliere den Quelltext für das folgende HTML-Dokument und speichere es unter dem Titel Aufgabe 3.7.1

.