

Deux

Auszeichnungssprache zur Erstellung von Textdateien

Vorbilder: HTML, TeX und Pascal-Sprachen wie Component Pascal;

Das Ziel ist eine pascal-typische Sprache, welche mit einem geringsten Aufwand eine schriftbildliche (typographisch) vollkommene Abbildung von Geschriften am Bildschirm ermöglicht. Die Sprache soll möglichst klein und einfach aufgebaut sein. Dabei steht insbesondere eine klare Trennung zwischen Struktur, Inhalt sowie Gestaltung im Mittelpunkt, um eine leichte Anpassung der Darstellung für verschiedenen Zwecke und Ausgabegeräte zu gewährleisten, sodass ohne Änderung des eigentlichen Quelltextes beispielsweise eine für den Buchdruck vorgesehene Textdatei als Inhalt in eine Webseite eingebunden werden kann.

Damit der Übersetzer Befehle vom eigentlichen darzustellenden Text unterscheiden kann, werden diese mit einem `\` gekennzeichnet (wie bei TeX).

Bei nicht darzustellenden Quelltexten ist dies nicht erforderlich (z. B. In einer Pascal-Datei). Um dennoch einen nach links geneigten Schrägstrich im eigentlichen Text schreiben zu können, wird dieser einfach verdoppelt: `\\`

Sollten mehrere Schrägstriche im Text dargestellt werden, reicht ein erster als Markierungszeichen aus; nach einem Leerzeichen muss jedoch erneut markiert werden.

Bemerkungen, um Quelltextbefehle oder sonstiges zu beschreiben, welche nichts mit dem eigentlichen Inhalt oder der Programmabfolge zu tun haben, also vom Übersetzer unbeachtet bleiben sollen, sind mit zwei nach rechts geneigten Schrägstrichen einzuleiten: `//`

Allg. Befehlsaufbau: `\Befehlsname[Kennzeichen](Angaben){Eigenschaften};`

Über die Kennzeichen können Struktureinheiten an einer anderen Stelle angewählt und Eigenschaften zugeordnet werden, sodass die Gestaltungsvorlage leicht austauschbar ist, ohne im eigentlichen (Inhaltlichen) Quelltext Änderungen vornehmen zu müssen. Dieses Vorgehen entspricht einer klaren Trennung zwischen strukturiertem Inhalt und Gestaltung zur Plattformunabhängigkeit.

In geschweiften Klammern erfolgen stets Stilangaben, die ausschließlich die Gestaltung vorgeben (z. B. Hintergrundfarbe).

Umgebungen, also untergeordnete Abschnitte, werden wie bei Pascal-Sprachen unmissverständlich mit „Beginn“ und „Ende“ dargestellt, denn Geschweifte Klammern umfassen ausschließlich Gestaltungsvorgaben; runde Klammern beinhalten stets Übergabewerte; und eckige Klammern stehen nur für Kennzeichen (bildhaft: Eigennamen), um woanders auf diese gesondert zugreifen zu können:

`\beginn`

`...`

`\ende`

Eine Verschachtelung wie bei HTML ist zwar eindeutig, jedoch viel zu umfangreich. Die Struktur kann dennoch ähnlich wie bei TeX genau vorgegeben werden:

```
\tabelle[einnahmen](5; 2{grün} 4{1,5cm; schriftfarbe: blau}){mittig};
```

```
\beginn
```

```
Jan | Feb | März | Apr | Mai ||  
12 | 732 | 2938 | 3838 | 383 ||  
847 | 338 | 3948 | 384 | 343 ||  
|| 394 | 345  
\ende
```

Mit | werden die einzelnen Zellen voneinander abgegrenzt. Ein || dient zur Markierung des Zeilenendes.

Der Tabellenbefehl erwartet als erste Angabe (in den runden Klammern) die Spaltenanzahl; jede weitere Abgabe ist zusätzlich: die Spalten werden von links nach rechts durchnummeriert, sodass mit der Angabe 2 die zweite Spalte ausgewählt wird und in nachfolgenden geschweiften Klammern Stilangaben erfolgen können, welche nur für diese zweite Spalte gelten.

Die nachfolgende Umgebung, eingeleitet mit \beginn und \ende beinhaltet die eigentliche Tabelle. Der jeweilige Inhalt wird bis || als Tabellenzeile aufgefasst und von oben nach unten durchnummeriert, sodass wie bei den Spalten, auf diese an anderer Stelle Bezug genommen werden kann:

```
tabelle[einnahmen], zeile 1 {schriftgröße: 16pt; hintergrundfarbe: 453525; ...}
```

Möchte man, dass alle Tabellen einen gleich gestalteten Tabellenkopf aufweisen, muss nur die Eingrenzung entfernt werden:

```
tabelle, zeile 1 {schriftgröße: 16pt; hintergrundfarbe: 453525; ...}
```

Damit wird grundsätzlich jede Zeile einer Tabelle nach den entsprechenden Angaben „gestylt“ und auf diese Weise ein recht gleichmäßiges Aussehen innerhalb eines Dokuments ermöglicht.

Manche Stilangaben werden häufig benötigt, sodass umgebungsabhängige Kurzschreibweisen vorteilhaft wären; z. B. wäre es logisch, dass eine bloße Zahlenangabe bei Spalten in erster Linie bezogen ist auf die Spaltenbreite. Dies würde einen längeren Befehle wie: 3{breite: 2cm}

zu: 3{2cm}
vereinfachen

Mitten im Quelltext sollten gestalterische Vorgaben äußerst vorsichtig und sparsam gesetzt werden, anstatt: 3{hintergrundfarbe: sommergrün}

wäre ein Zugriff an zentraler Stelle (im Dokumentenkopf) über:

```
tabelle[...] spalte 3{hintergrundfarbe: sommergrün}
```

passender. Es spräche nichts dagegen, grundsätzlich allen in sich geschlossenen Struktureinheiten, also auch bei Spalten sowie Zeilen Kennzeichnungen mit [...]

zuzulassen: 2[neben]{...}

Soll jede zweite Zeile mit der selben Hintergrundfarbe versehen werden, so ist im Gestaltungsbereich folgendes anzugeben:

tabelle[...], 2. zeile {hintergrundfarbe: hellgrau; schriftstil: fett}

Möchte man hingegen nur ganz bestimmte Zeilen gleichartig gestalten:

tabelle[...], zeile 2, 3, 6 {hintergrundfarbe: hellgrau; schriftstil: fett}