

7 Tabellen

- übersichtliches Gestalten von Websites
- enorme Steigerung der Layoutmöglichkeiten durch Tabellen

7.1 Erstellen einer Tabelle

Syntax:	Kommentar
<TABLE >	<!-- Tabelle wird geöffnet-->
<TR>	<!--1.Zellenreihe wird geöffnet-->
<TD>Inhalt1</TD>	<!--Zelle1,Spalte1 wird geöffnet und geschl.-->
<TD>Inhalt2</TD>	<!--Zelle2,Spalte1 wird geöffnet und geschl.-->
</TR>	<!--1.Zellenreihe wird geschl.-->
<TR>	<!--2.Zellenreihe wird geöffnet-->
<TD>Inhalt1</TD>	<!--Zelle1,Spalte2 wird geöffnet und geschl.-->
<TD>Inhalt2</TD>	<!--Zelle2,Spalte2 wird geöffnet und geschl.-->
</TR>	<!--2.Zellenreihe wird geschl.-->
</TABLE>	<!-- Tabelle wird geschlossen-->

- Attribut BORDER innerhalb des Tags <TABLE> definiert Rahmen um eine Tabelle
- Dicke des Randes in Pixeln
- Lässt man BORDER weg, dann ist Tabellenrahmen nicht sichtbar
BORDER=0 um rahmenlose Tabelle zu erzeugen, funktioniert mit vielen Browsern nicht

Beispiel:

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN//">
<HTML>
<HEAD> <TITLE> </TITLE> </HEAD>

<BODY>
<TABLE BORDER=
  <TR>
    <TD>
      Beispiel für eine Tabelle
    </TD>
    <TD>
      und zwei Spalten einer Reihe
    </TD>
  </TR>
</TABLE>
</BODY>
</HTML>

```

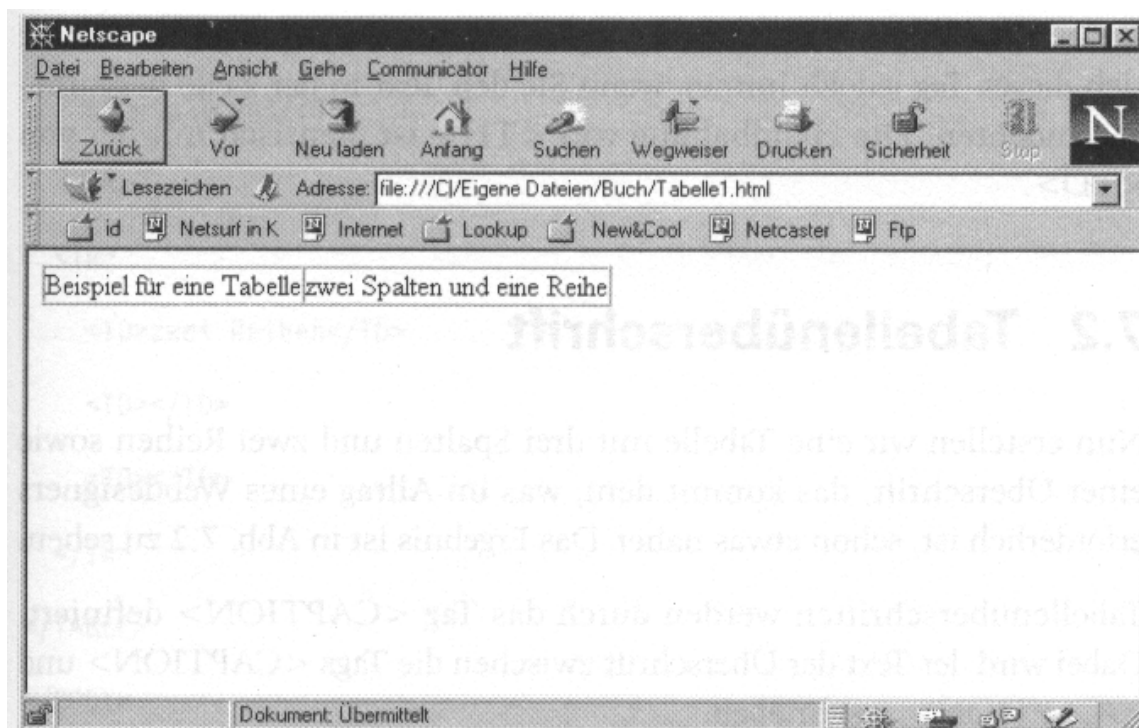


Abb. 7.1: Tabelle mit 2 Spalten

Spalten mit <TH> anstatt <TD>

- Inhalt wird fett und Zelle zentriert ausgegeben
- gedacht für Kopfzellen

7.2 Tabellenüberschrift

- Überschrift wird zwischen Tag<CAPTION> und </CAPTION> geschrieben

Tabellenunterschrift

- mit Attribut ALIGN=Bottom im Tag <CAPTION> wird aus einer Überschrift eine Unterschrift

Die Größe einer Tabellenzelle

- Feldgröße passt sich automatisch an
- Attribut WIDTH lässt sich bei allen Zellen, die mit <TD> erstellt wurden, anwenden (sowohl absolut als auch in Prozent zur verfügbaren Breite des Browserfensters)

Tabellenzellhöhe anpassen:

- Eingabe des Entity für ein Leerzeichen () und einer Absatzmarke mit Hilfe des Tagpaares <P>

Beispiel:

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN//">
<HTML>
<HEAD> <TITLE> </TITLE> </HEAD>
<BODY>
<TABLE BORDER>
<CAPTION> Tabelle mit &Uuml;berschrift </CAPTION>
  <TR>
    <TD>&nbsp;
      <P> </P>
      Beispiel f&uuml;r eine Tabelle</TD>
    <TD>und drei</TD>
    <TD>Spalten</TD>
  </TR>
  <TR>
    <TD>zwei Reihen &nbsp;
      <P> </P>
    </TD>
    <TD>
    </TD>
    <TD>
    </TD>
  </TR>
</TABLE>
</BODY>
</HTML>

```

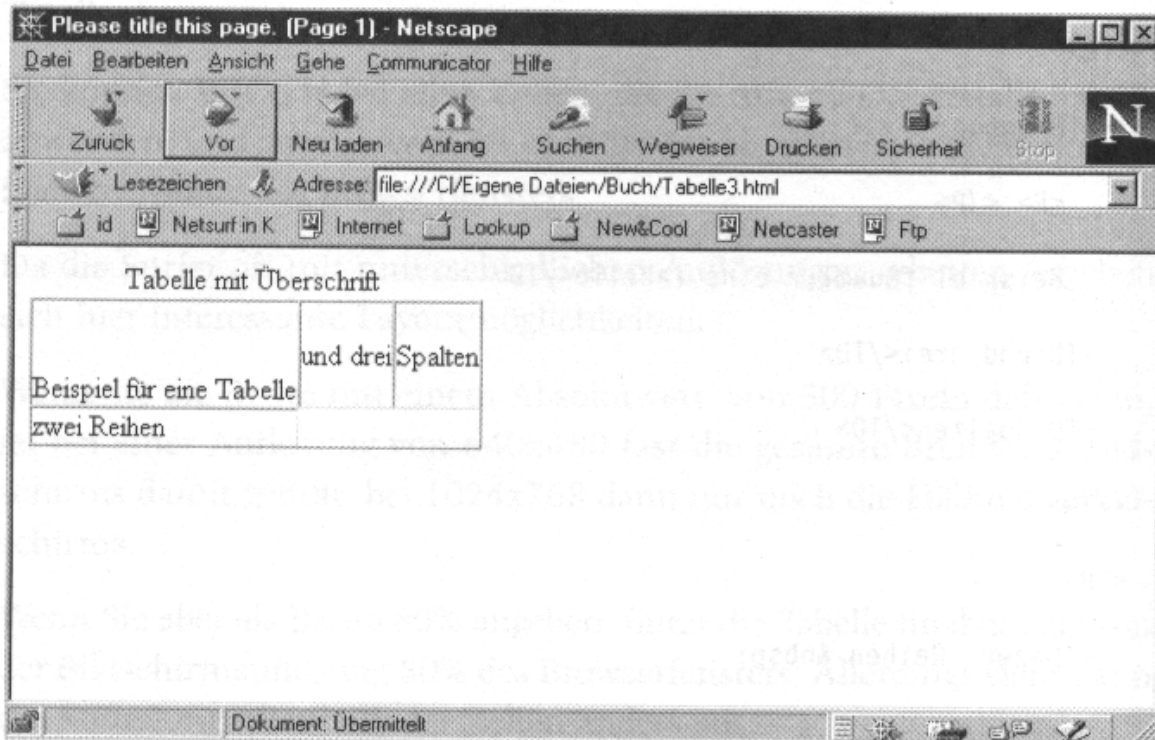


Abb. 7.4: Zellhöhe verändert

CELLPADDING und CELLSPACING

- CELLPADDING dient dazu, den Abstand der Zellen zueinander einzustellen.
- Mit CELLSPACING wird der Abstand des Zellinhalts zum Zellenrand eingestellt.

Beispiel:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN//">
<HTML>
<HEAD> <TITLE></TITLE> </HEAD>
<BODY>
<TABLE BORDER CELLSPACING=0 CELLPADDING=8>
<CAPTION> Cellpadding </CAPTION>
  <TR>
    <TD>Beispiel f&ampuumlr eine Tabelle</TD>
    <TD>und drei</TD>
    <TD>Spalten</TD>
  </TR>
  <TR>
    <TD>zwei Reihen</TD>
    <TD></TD>
    <TD></TD>
  </TR>
</TABLE>
<TABLE BORDER CELLSPACING=8 CELLPADDING=0>
<CAPTION> Cellspacing </CAPTION>
  <TR>
    <TD>Beispiel f&ampuumlr eine Tabelle</TD>
    <TD>und drei</TD>
    <TD>Spalten</TD>
  </TR>
  <TR>
    <TD>zwei Reihen</TD>
    <TD></TD>
    <TD></TD>
  </TR>
</TABLE>
</BODY>
</HTML>
```

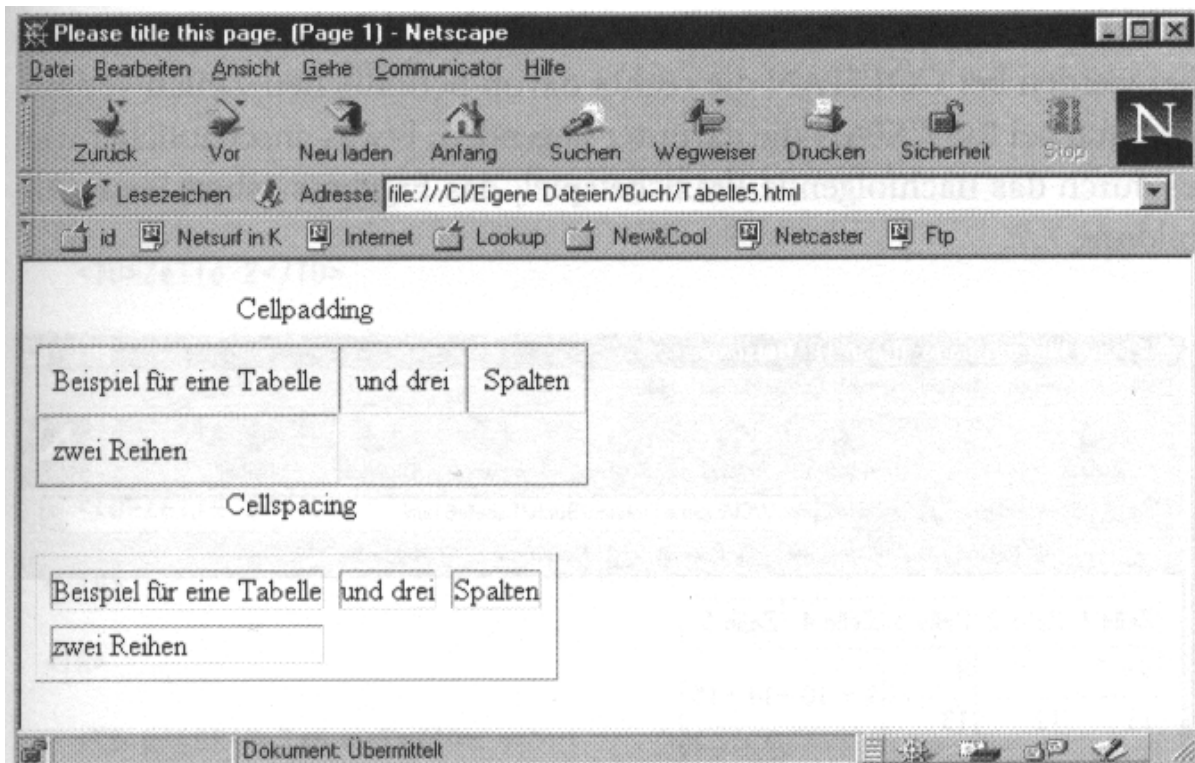


Abb. 7.6: Cellpadding und Cellspacing

7.4 Verbinden von Zellen

- innerhalb der Tags `<TD>` oder `<TH>` Attribute `ROWSPAN` und `COLSPAN`
- `<TD COLSPAN=n>` n rechts befindliche Zellen werden verbunden
- `<TD ROWSPAN=n>` n darunter liegende Zellen werden verbunden

7.5 Farbe in der Tabelle

- Attribut `BGCOLOR` sowohl für ganze Tabelle im Tag `<TABLE>` oder für einzelne Zelle im Tag `<TD>`
- `<TABLE BGCOLOR=Farbe>` wobei Farbe durch einen gültigen Farbnamen oder den entsprechenden Hexadezimalcode ersetzt wird
- Attribut `BORDERCOLOR` zur Veränderung der Farbe des Tabellenrandes
- kann auch innerhalb des Tags `<TD>`, `<TR>` oder `<TH>` verwendet werden

Beispiel:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN//">
<HTML>
<HEAD> <TITLE>Mein erstes Dokument</TITLE> </HEAD>
<BODY>
<TABLE BORDER CELLSPACING=0 CELLPADDING=0>
<CAPTION>Einzelne Zellen in einer Farbe </CAPTION>
  <TR>
    <TD BGCOLOR="#ff0000">&nbsp;
      <P> </P>
      Beispiel f&uuml;r eine Tabelle</TD>
    <TD>und drei</TD>
    <TD BGCOLOR="#a0a0a0">Spalten</TD>
  </TR>
  <TR>
    <TD>zwei Reihen</TD>
    <TD>
</TD>
    <TD>
</TD>
  </TR>
</TABLE>
</BODY>
</HTML>
```

Wie in Abbildung 7.10 zu sehen, ist nun die erste Zelle der oberen Reihe rot sowie die dritte Zelle der oberen Reihe weiß.

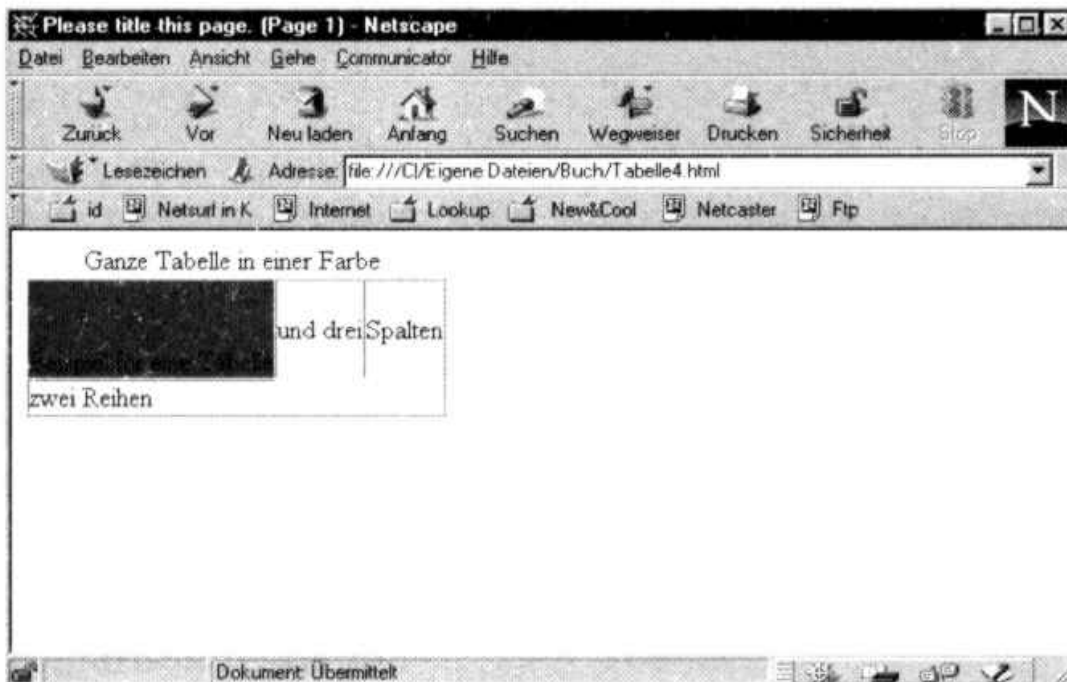


Abb. 7.10: Farbige Tabelle

7.6 Tabellen als Layouthilfe

Fast jedes gewünschte Layout kann mit Hilfe von Tabellen erreicht werden. Durch die oben kennengelernten Möglichkeiten läßt sich Spaltensatz und punktgenaues Einbinden von Grafiken verwirklichen. Hier haben Sie mit das mächtigste Werkzeug zur professionellen Seitengestaltung mit HTML zur Hand.

Der häufigste Anwendungszweck für das Tag `<TABLE>` im WWW ist mehrspaltiger Satz. Dabei erstellen Sie eine Tabelle mit einer Reihe und z. B. zwei oder drei Tabellenzellen nebeneinander, je nachdem, wie viele Spalten Sie setzen möchten.

Bedenken Sie jedoch, daß aufgrund niedriger Bildschirmauflösungen mehr als drei Spalten die Seite oft unübersichtlich machen.



Abb. 7.11: Seitenlayout durch Tabellen

8 Frames

Frames sind nichts anderes als Rahmen

Vorteile:

- verbessertes Seitenlayout
- feste Navigationsleiste oder Logo (nur einmal laden)

Nachteile:

- längere Dauer beim Bildaufbau (es müssen mehrere HTML-Dokumente geladen werden, bis das erste fertige Dokument im Browser zu sehen ist.
- Ansehnliche Darstellungen nur bei hohen Bildschirmauflösungen

Bedingungen für Frames werden immer besser.

(Einsatz abhängig von der Zielgruppe)

8.1 Das Frameset-Grundgerüst

Wenn eine Seite aus drei Frames besteht, dann lädt der Browser drei zusätzliche HTML-Dokumente, In jedes Frame wird ein Dokument geladen.

Das Tag <FRAMESET>

- zum Aufbau der Frames und laden der Inhalte durch den Browser, wird zuerst ein HTML-Dokument geladen, das Informationen zum Aufbau der Frames enthält.
- <FRAMESET> Definition der Frames </FRAMESET>
- Definition des Frameset benötigt ein eigenes Dokument
- Wird an Stelle des Tag <BODY> verwendet

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Frameset//EN//">
```

```
<HTML>
```

```
<HEAD> <TITLE> </TITLE> </HEAD>
```

```
<FRAMESET>
```

```
</FRAMESET>
```

```
</HTML>
```

Beachten Sie auch die geänderte *Document Type Declaration* (DTD) in der ersten Zeile des Quelltextes. Diese gibt an, daß es sich bei diesem Dokument um ein *Frameset* handelt und daß es konform mit HTML 4.0 definiert wurde.

- Attribut FRAMEBORDER=n wird von HTML 4.0 zwingend vorgeschrieben; n=0 –kein Rand n=1 Rand
- Attribut COLS – vertikale Trennung
- Attribut COLS="n,n" n - absoluter Wert in Pixeln oder
- Attribut COLS="n%,n%" n - prozentualer Wert
- Attribut ROWS= x,x x – wie bei COLS, horizontale Trennung
- Mehr als zwei Frames möglich

- Kombination von ROWS und COLS durch Verschachtelung des Tag <FRAMESET>

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Frameset//EN//">
```

```
<HTML>
```

```
<HEAD> <TITLE> </TITLE> </HEAD>
```

```
<FRAMESET FRAMEBORDER=1 COLS="20%,*%">
```

```
<FRAMESET FRAMEBORDER=1 ROWS="30%,*%">
```

```
</FRAMESET>
```

```
</FRAMESET>
```

```
</HTML>
```

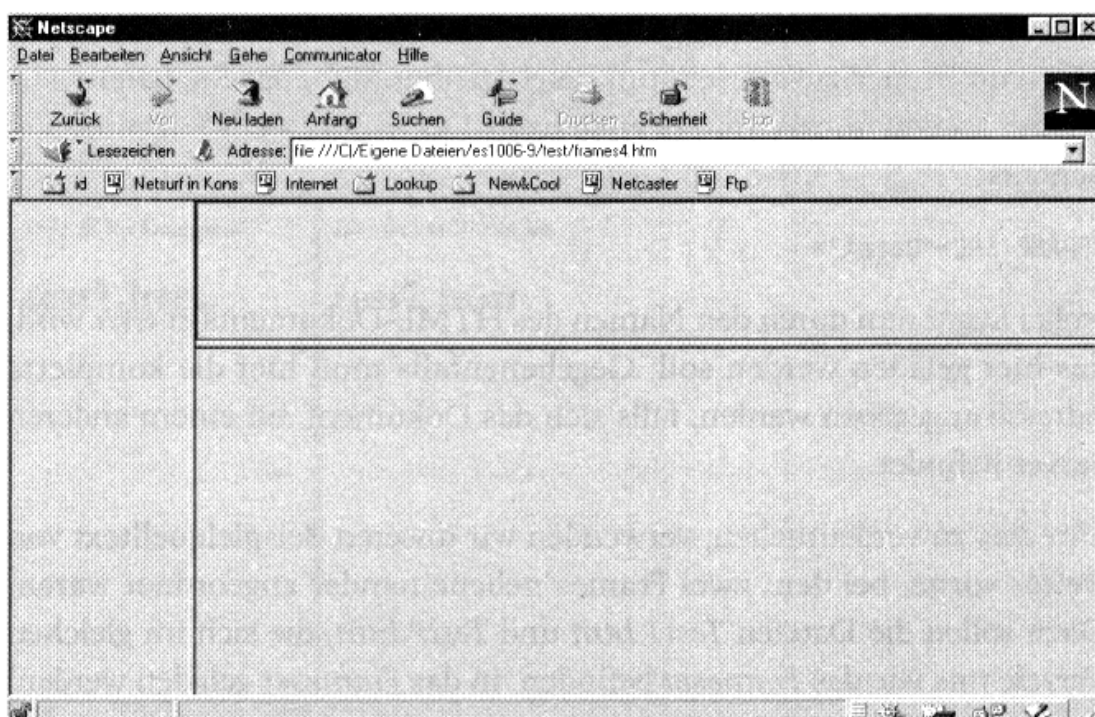


Abb. 8.4: Kombination von COLS und ROWS

8.2 Die Rahmen füllen

- bisher nur Gerüst der Seite, nun müssen wir die Rahmen noch mit Inhalten füllen
- dazu wird jedem Frame ein HTML-Dokument zugeordnet, mit Hilfe des Tag `<FRAME>` und dem Attribut `SRC`

```
<FRAME SRC="Datei">
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Frameset//EN//">

<HTML>

<HEAD> <TITLE> </TITLE> </HEAD>

<FRAMESET FRAMEBORDER=1 COLS="20%,80%">

<FRAME SRC="Test1.htm">

<FRAME SRC="Test2.htm">

</FRAMESET>

</HTML>
```

Dieser Quelltext erzeugt im Browser die folgende Abbildung 8.5. unter der Voraussetzung, die beiden zu ladenden Dateien (*Test1.htm* und *Test2.htm*) befinden sich auch im gleichen Verzeichnis wie das Dokument. Am besten erstellen Sie sich dafür zwei einfache HTML-Dokumente, die Sie dann laden können.

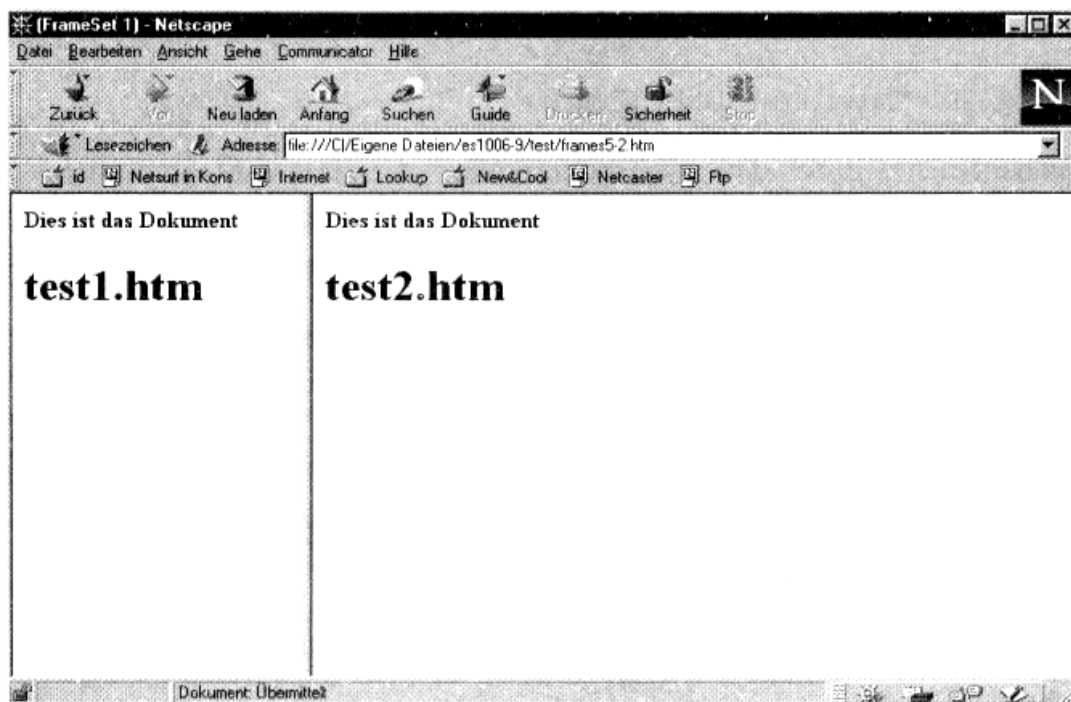


Abb. 8.5: Frameset mit Inhalt

8.3 Zuordnung der Frames

- häufige Einsatzmöglichkeit: Navigationsleiste, die beim Browsen durch die Webseite immer stehen bleibt
- Damit die dort ausgeführten Links nicht im gleichen Fenster erscheinen müssen Frames einen Namen haben (Attribut NAME)

```
<FRAME SRC="Datei" NAME="Frame1">
```

- nun können wir mit Hilfe eines Links innerhalb des einen Fensters das Zieldokument im zweiten Fenster anzeigen lassen
- Dazu Attribut TARGET

Der Einsatz erfolgt innerhalb der Definition des Links, der, wie bereits bekannt, mit Hilfe des Tags `<A>` definiert wird:

```
<A HREF="Test3.htm" TARGET="n"> Link </A>
```

wobei *n* durch den Namen des *Zielframe* ersetzt wird.

Die Integration in das Dokument verdeutlicht im nachfolgenden Quelltext noch einmal die Anwendung.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN//">
<HTML>
<HEAD> <TITLE> </TITLE> </HEAD>
<BODY>
<H2> Test1.htm </H2>
<A HREF="Test3.htm" TARGET="2"> Link zu Test3.htm </A>
```

Die Darstellung des Dokuments Test3.htm soll im rechten Fenster erfolgen.

```
</BODY>
</HTML>
```

Wenn Sie nun diesen Quelltext unter dem Namen *Test1.htm* abspeichern, dann wird Ihnen der Browser nachfolgende Abbildungen zeigen. Abbildung 8.6 zeigt Ihnen das Erscheinungsbild vor Ausführung des Links, Abbildung 8.7 nach Ausführung des Links.

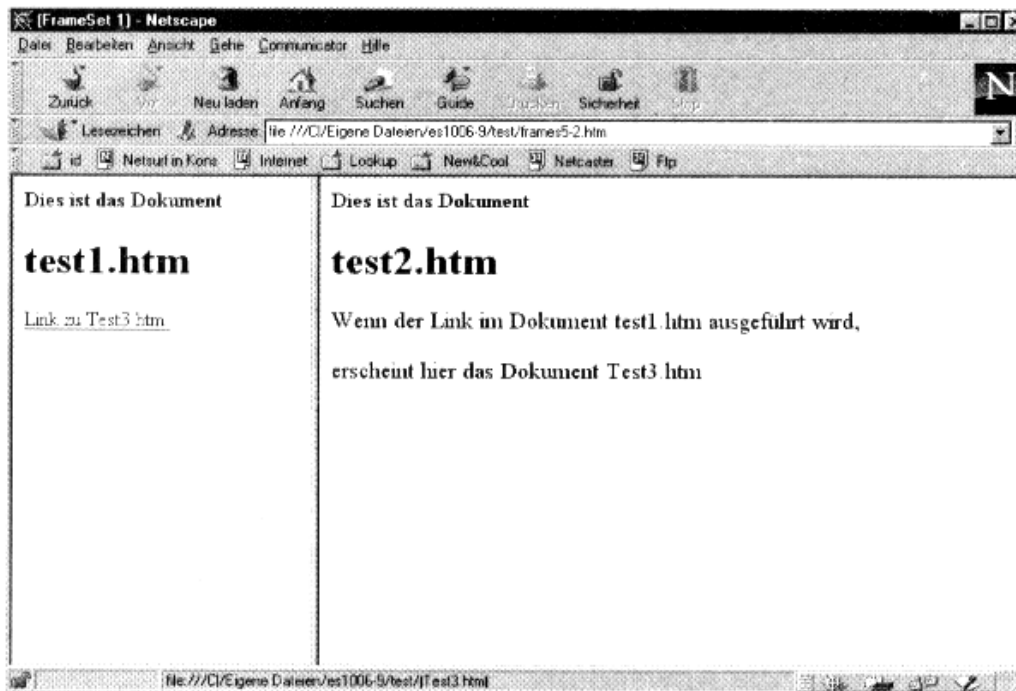


Abb. 8.6: Vor Ausführung des Links

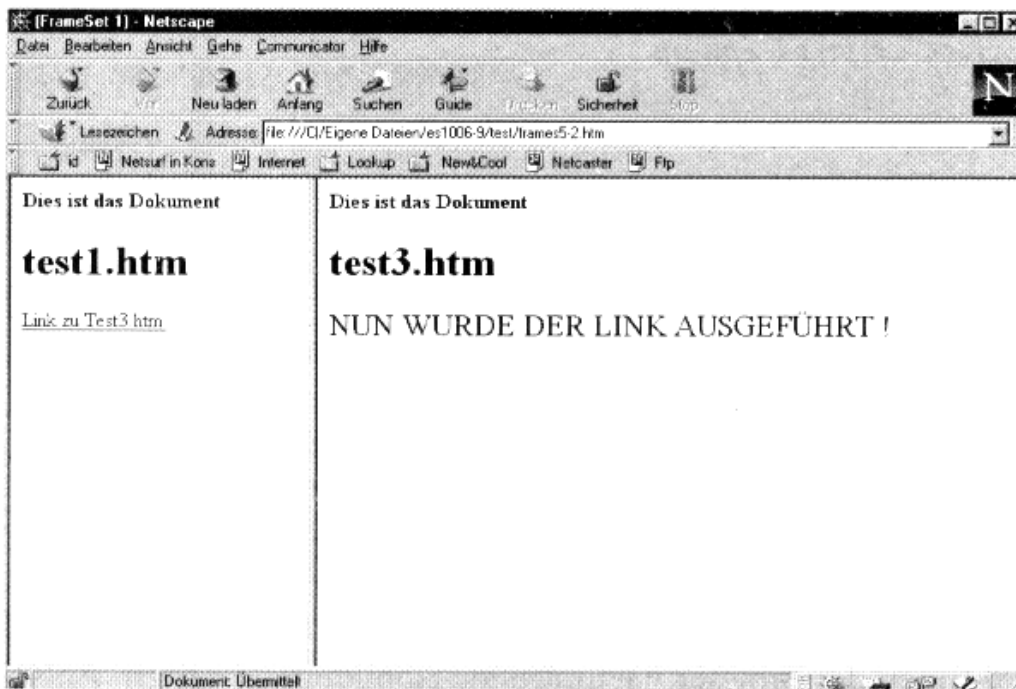


Abb. 8.7: Nach Ausführung des Links

8.4 Darstellung der Frames

Scrolleisten

- enthält diese immer dann, wenn Fläche nicht ausreicht
 - mit Attribut SCROLLING kann darauf Einfluß genommen werden
- <FRAME SCROLLING=n> n - Yes ⇒ Scrolleisten immer
 n - No ⇒ Scrolleisten nie

Abstände:

<FRAME MARGINHEIGHT=n> n-Pixelabstand des Inhaltes zum oberen und unteren Rand des Rahmens

<FRAME MARGINWIDTH=n> n-Pixelabstand des Inhaltes zum linken und rechten Rand des Rahmens

Größe eines Frames:

Attribut NORESIZE – feste Größe des Rahmens

8.5 Alternative zu Frames

- wenn Browser keine Frames darstellen kann bleibt Fenster leer
- Tag <NOFRAMES> dient oft nur dazu eine Information auf den Bildschirm zu bringen, dass ein Browser benötigt wird der Frames darstellen kann.
- Man kann aber zwischen den Tags <NOFRAMES> und </NOFRAMES> alternative Inhalte bzw. Verweise angeben, bei denen keine Frames verwendet werden
- Man kann das selbe schreiben wie zwischen <BODY> und </BODY>

Layoutgestaltung mit *Frames*

Frames ohne Rahmen

Wie schon bei den Tabellen eignen sich auch *Frames* durch den gleichen Trick, dem Weglassen des sichtbaren Randes (FRAMEBORDER=0), hervorragend als Mittel zur optischen Aufwertung.

Besonders gut lassen sich mit Hilfe von *Frames* einheitliche Seitenlayouts innerhalb einer Website festlegen. So wird dann z. B. das Logo nur einmal geladen, und lediglich der Textinhalt wird in einem gesonderten *Frame* immer neu geladen.

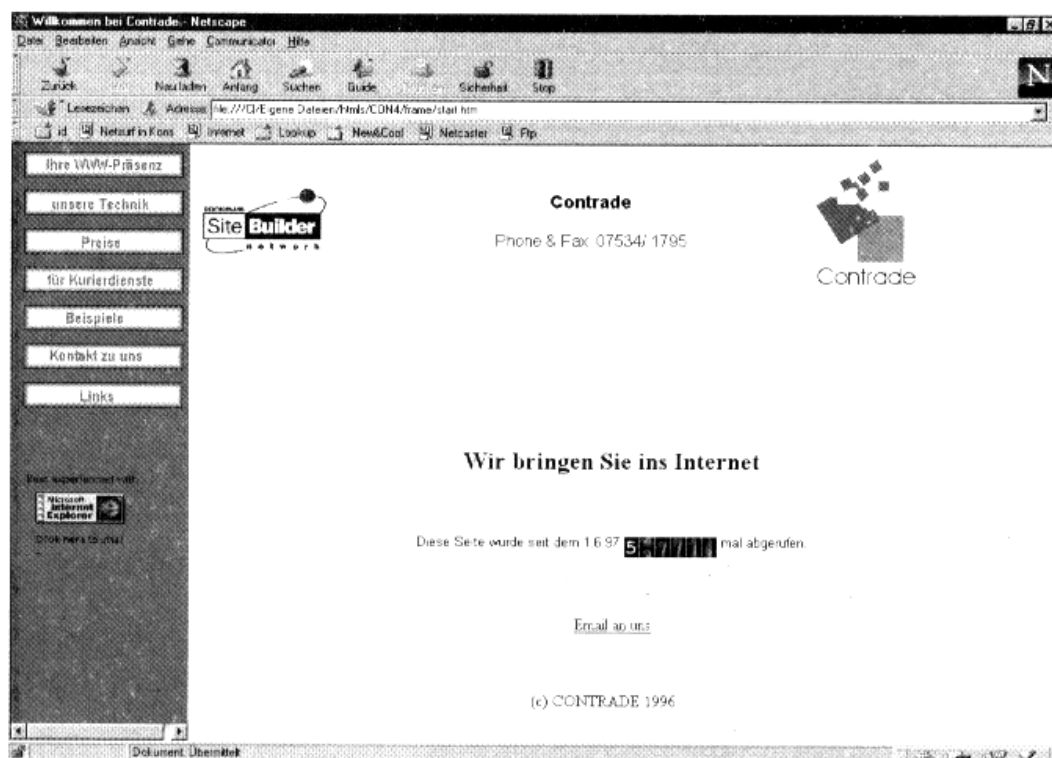


Abb. 8.11: Frames ohne sichtbaren Rand

Testen Sie hier einfach mal ein wenig herum. Besonders effektvolle HTML-Dokumente lassen sich durch die Kombination von *Frames* und Tabellen zur Layoutgestaltung erreichen.

Das A und O des effektvollen Einsatzes von *Frames* ist die Planung der *Frameaufteilung*. Bedenken Sie dabei stets, daß jedes *Frame* den Computer des Betrachters verlangsamt. Deshalb sollten Sie nie zu viele *Frames* in ein *Frameset* integrieren, drei bis maximal vier *Frames* sind meist sowieso ausreichend.

Formulare (Grundlagen)

9.1 Die Syntax

Die Definition eines kompletten Formulars wird zwischen den Tags `<FORM>` und `</FORM>` geschrieben.

Zwei Attribute müssen dem Formular zugewiesen werden, damit es auch etwas bewirkt, wenn es beendet wird, `ACTION` und `METHOD`.

```
<FORM METHOD=n ACTION="URL"> Definition </FORM>
```

METHOD

`n=get` ⇒ Formular wird durch ein Programm ausgewertet

`n=post` ⇒ Inhalt wird an eine E-Mail-Adresse versandt

ACTION

`URL` ⇒ wird durch gewünschte Adresse ersetzt.

9.2 Textfelder

```
<INPUT NAME="n">
```

`n` ⇒ gewünschter Name

Beschriftung eines Feldes

- wie bei normalen HTML-Dokument

Größe der Eingabefelder

```
<INPUT NAME="n" SIZE="n" MAXLENGTH="n" >
```

`SIZE` - legt Größe des Textfeldes fest

`MAXLENGTH` - wie viel Zeichen darf der enthaltene Text lang sein