



# HTML & Co

Ein kurzer Einblick

# Was ist HTML?

- ◆ HTTP = Datenübertragungsprotokoll
- ◆ HTML (Hypertext Markup Language)  
= Formatiersystem für Textdokumente
- ◆ Entwickelt in Genf (CH) am CERN
- ◆ Befehle zur Formatierung, Ausrichtung von Texten, Bildern, anderen Objekten
- ◆ Von Browsern ausgeführt und dargestellt

# HTML-Grammatik

- ◆ HTML-Elemente (Tags) stehen in spitzen Klammern z.B. `<Befehl>`
- ◆ Befehl gilt bis zur Aufhebung („Container“)  
`<Befehl>Bereich</Befehl>`
- ◆ Verschachtelung möglich  
`<center><bold>fett zentriert</bold></center>`
- ◆ bestimmte Befehle haben Attribute (Argumente)  
`<p align=„left“>linksbündiger Absatz</p>`

# Grundgerüst HTML-Datei

## ◆ Dateikopf (head) und Dateikörper (body)

`<html>` (*Beginn der HTML-Datei*)

`<head>` (*Beginn des Dateikopfes*)

Im Head-Container allgemeine Info, im Browser nicht angezeigt

`</head>` (*Ende des Dateikopfes*)

`<body>` (*Beginn des Dateikörpers*)

Hier alle Elemente, die im Browserfenster gezeigt werden

`</body>` (*Ende des Dateikörpers*)

`</html>` (*Ende der Datei*)

# Dateikopf (Container <head>)

- ◆ Titel-Tag

- ◆ `<title>` max. 64 Zeichen `</title>`

Anzeige in der Titelleiste des Browsers  
(FrontPage – Seiteneigenschaften)

- n Aussagekräftige, sprechende Titel,

- n Kurze Titel, Name

# Dateikopf (2) – Meta-Tags

- ◆ `<Meta>`-Tag: Info über das Dokument selber (meta = über)

- ◆ Nur auf der Eingangsseite (z.B. *index.htm*)

- ◆ Kann mehrmals vorkommen

- n Attribut **name** = Name der Metainformation (GENERATOR, copyright, keywords, Author, description...)

- n Attribut **content** = Inhalt der Metainformation

- `<meta name=„Author“ content=„Wannerer“ >`

- `<meta name=„GENERATOR“ content=„Microsoft FrontPage 5.0“ >`

# Meta-Tags

## ◆ Vor allem Info für die Suchrobots

Beispiel Stichworte

```
<meta name=„keywords“ content=„Stichwort1, Stichwort2, Stichwort3,  
usw.“>
```

## ◆ Beispiel Dateikopf

```
<head>  
<title>Franz Webmeisters private Site</title>  
<meta name=„GENERATOR“ content=„Microsoft FrontPage 4.0“>  
<meta name=„copyright“ content=„Franz Webmeister, private Homepage“>  
<meta name=„keywords“ content=„Motrorrad, motorradfahren, radfahren, Blues,  
Weinbau, Obstbau, Schnaps“>  
</head>
```

# Meta-Tags

`<META HTTP-EQUIV="Content Type" CONTENT="text/html; charset=ISO-8859-1">`

- Art der Datei (html-Text)
- Zeichensatz (iso-Latin): damit werden Sonderzeichen automatisch richtig dargestellt

`<META HTTP-EQUIV="Language" CONTENT="de">`

Hilfe für Suchmaschinen zur Spracherkennung

`<meta name="author" content="Nachname Vorname">`

ermöglicht das schnelle Auffinden aller Webdokumente, für die eine Person verantwortlich ist.

`<meta name="robots" content="index, follow">`

teilt dem Robot der Suchmaschine mit, dass er alle Links auf der Seite weiter verfolgen soll.

`<meta name="revisit-after" content="3 weeks">`

teilt dem Robot der Suchmaschine mit, dass er alle die Seite alle 3 Wochen neu indexieren soll.

# Meta-Tags

`<meta name="description" content="Der Mann im Eis: Neuste Forschungsergebnisse ...">`

Die Beschreibung wird z.B. von Altavista mit dem URL zurückgegeben.

`<meta name="robots" content="index,follow">`

teilt dem Robot der Suchmaschine mit, dass er alle Links auf der Seite weiter verfolgen soll.

`<meta name="revisit-after" content="3 weeks">`

teilt dem Robot der Suchmaschine mit, dass er alle die Seite alle 3 Wochen neu indexieren soll.

## Beispiel

```
<META NAME="language"      CONTENT="de">
<META NAME="author"       CONTENT="Wannerer">
<META NAME="publisher"    CONTENT="Wannerer">
<META NAME="copyright"    CONTENT="Wannerer">
<META NAME="description"  CONTENT="das ist meine seite über....">
<META NAME="keywords"     CONTENT="wort1, wort2, wort3, wort4">
<META NAME="page-topic"   CONTENT="Webdesign">
<META NAME="page-type"    CONTENT="Kurs, Unterlagen">
<META NAME="audience"    CONTENT="Schüler, Studenten">
<META NAME="robots"       CONTENT="index,follow">
```

# Editieren von <meta> Tags

- ◆ In der HTML-Ansicht
- ◆ z.B. Frontpage Seiteneigenschaften
  - n Register benutzerdefiniert
  - n Längere Listen unübersichtlich

# Dateikörper (Container <body>)

◆ alles im Browser-Fenster dargestellt

Attribut für body	Variable	Angabe
Oberer Rand	<code>topmargin="5"</code>	Optional, sonst Rand=0
Linker Rand	<code>leftmargin="5"</code>	Optional, sonst Rand=0
Hintergrundfarbe	<code>bgcolor="#10BCF5"</code> (RGB – HEX)	Optional, sonst Standard
Hintergrundbild	<code>background="images/bild.gif"</code>	Alternativ zur Hintergrundfarbe

# einige Tags

Tag	Bedeutung
<code>&lt;h1&gt;</code> bis <code>&lt;h6&gt;</code>	Überschriften (Header)
<code>&lt;p&gt;</code>	Absatz (paragraph)
<code>&lt;br&gt;</code>	Zeilenumbruch (allein)
<code>&lt;ul&gt;</code>	Aufzählung (unordered list)
<code>&lt;ol&gt;</code>	Nummerierung (ordered list)
<code>&lt;li&gt;</code>	einzelner Listeneintrag
<code>&lt;dl&gt;</code> <code>&lt;dt&gt;</code> <code>&lt;dd&gt;</code>	Definitionsliste, Begriff (term), Definition

Tag	Bedeutung
<code>&lt;a href&gt;</code>	Link (anchor hyper reference)
<code>&lt;a name&gt;</code>	Textmarke (benannter Anker)
<code>&lt;b&gt;</code>	fett (bold)
<code>&lt;i&gt;</code>	kursiv (italic)
<code>&lt;table&gt;</code>	Tabelle
<code>&lt;u&gt;</code>	unterstrichen
<code>&lt;span&gt;</code>	inline-Bereich
<code>&lt;div&gt;</code>	allgemeiner Bereich (division)

# Verweise (links)

1

```
<a href=„images/bilder.htm“ >Bild sehen</a>
```

Link-Ziel (URL)

Text (oder Bild)  
für Benutzer

# Verweise (links) 2

## ◆ Intern (mit relativer Adressierung)

`<a href=„images/bilder.htm“ >` `</a>`

↓  
Untergeordneter Ordner

`<a href=„../..//start.htm“ >` `</a>`

↓  
übergeordneter Ordner

## ◆ Extern (vollständige URL)

`<a href=„http://orf.at“ >` `</a>`

# Verweise – Anker 3

- ◆ Benannte Anker (Textmarken) als Ziel im (längeren) Dokument

```
<a name=„anfang“ > </a>
```

- ◆ Link (Sprung) zum Anker

```
<a href=„#anfang“ >zum Anfang </a>
```

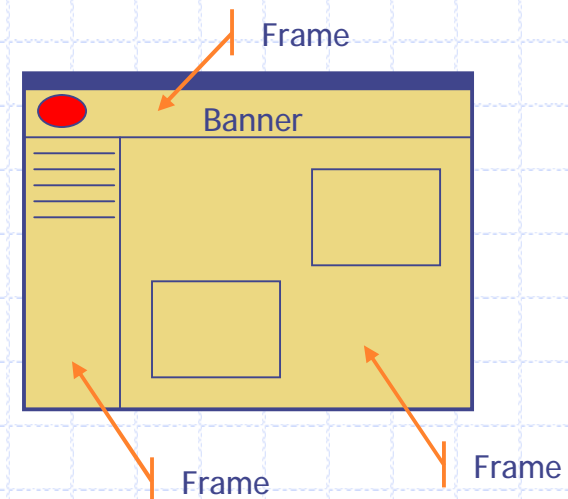
# Verweise (Links)

## 4

- ◆ Aussagekräftig
- ◆ Nicht zu kurz und nicht zu lang
- ◆ Bezug zu Überschrift/Thema der Seite
- ◆ Verweis auf Homepage
- ◆ Textmarken bei längeren Seiten
- ◆ Zu viele links im Text vermeiden

# Frames (1)

- ◆ Browserfenster geteilt
- ◆ Jeder Frame ein Dokument
- ◆ Teile der Seite ständig sichtbar
  - n Navigation
  - n Logo
  - n Hilfetext



# Frames: Vorteile/Nachteile

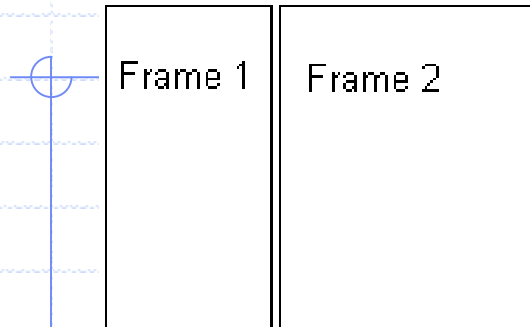
## ◆ Vorteile

- n Kein ermüdender Neuaufbau
- n Einfache Gliederung

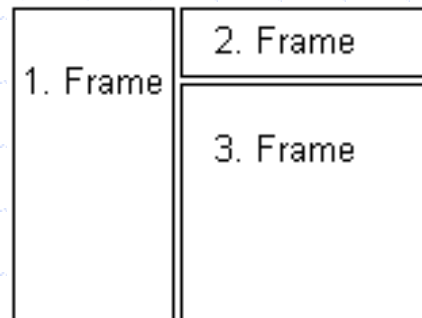
## ◆ Nachteile

- n Von Suchmaschinen ignoriert
- n Speichern und Drucken schwierig
- n Probleme für Links und Bookmarks
- n (Von „alten“ Browsern nicht unterstützt)

# Frames - HTML



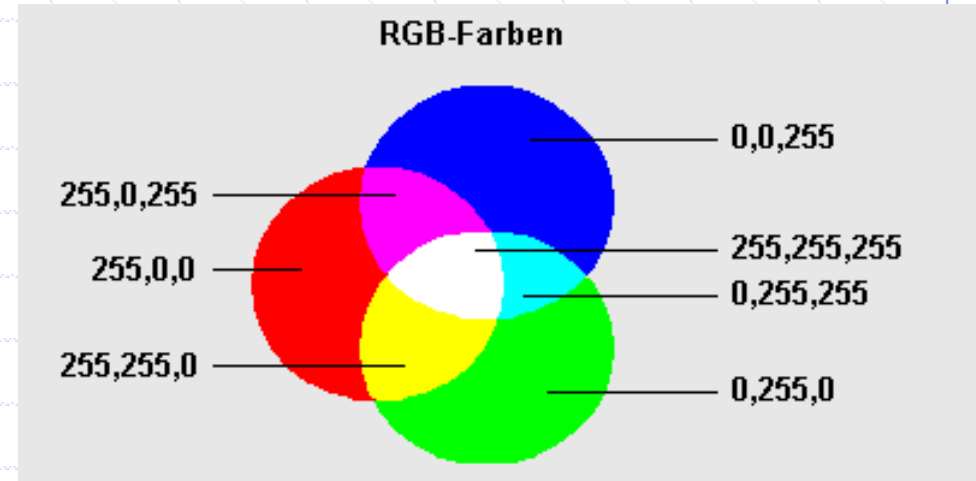
```
<frameset cols="200,*">  
  <!-- Dadurch ergeben sich zwei Framefenster,  
  deren Inhalt hier bestimmt wird -->  
</frameset>
```



```
<frameset cols="40%,60%">  
  <!-- zwei Framefenster, deren Inhalt hier bestimmt  
  wird Das erste Framefenster wird normal definiert,  
  anstelle des zweiten folgt jedoch das folgende  
  untergeordnete Frameset: -->  
  <frameset rows="20%,80%">  
    <!-- Noch mal zwei Framefenster,  
    deren Inhalt hier bestimmt wird -->  
  </frameset>  
</frameset>
```

Frame-Inhalt: `<frame src="startseite.htm" name="Daten">`

# Farben in HTML



## ◆ RGB-Modell

- n Rot-, Grün-, Blau-Anteil (0 – 255)
- n additiv (alle Farben voll = weiß)

## ◆ Hexadezimal in HTML

n `<body bgcolor=#0000CC`

Rot-Wert 00  
(kein Rotanteil)

Grün-Wert 00  
(kein Grünanteil)

Blau-Wert CC  
(Blauanteil CC = 204 dez.)

# Cascading Style Sheets - CSS

- ◆ Befehlssystem für HTML-Elemente bzw. Erscheinungsregeln für die HTML-Tags
- ◆ Trennung: Funktionalität und Layout
- ◆ 2 Teile einer CSS-Regel
  - n **SELECTOR** = HTML-Element, Tag
  - n **DECLARATION** = wie ist das HTML-Element darzustellen?  
z.B.: **h1** { **font-family:Arial;** }  
allgemein:  
**selector** {**declaration1;declaration2;declatartion3;**}

# Drei Arten CSS Einbindung (1)

- ◆ Inline-Styles im Tag: Formatangaben in HTML-Befehlszeile (keine Selektoren; Verwendung des Attributes **style**)

```
<html>
<head> <title>Titel der Datei</title></head>
<body style="background-color:#000000; font-family:Verdana; color:#FFFFFF;">
<h1 style="background-color:#000055;color:rgb(200,200,200);">Überschrift 1</h1>
</body>
</html>
```

Anwendung in einzelnen Fällen sinnvoll!

# Drei Arten CSS Einbindung (2)

- ◆ Intern im HTML Kopf (head) eingebunden  
zentral für eine Datei (mit Selektoren)

```
<html>
<head><title>Titel der Datei</title>
<style type="text/css">
<!--
h3 { font-family:Arial;Helvetica; font-size:24pt; color:rgb(200,0,0); }
-->
</style>
</head>
<body> </body>
</html>
```

*In diesem Beispiel wird das Format der Überschrift Größe 3 für das gesamte Dokument festgelegt.*

# Drei Arten CSS Einbindung (3)

## ◆ Externes Style-Sheet: Trennung von Layout und Inhalt einer Seite

HTML-Seite

```
<html>
<head>
<title>Datei</title>
<link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>

</body>
</html>
```

Link  
zu  
Stylesheet

Stylesheetdatei: **style.css**

```
a:link { color:#FFCC00;font-size:10pt;}
a:visited { color:#99CCCC; font-size:10pt;}
a:hover { color:#FF0000;font-size:10pt;}
a:active { color:#66FF00; font-size:10pt;}
body
{
font-family: Verdana,Arial,Helvetica;
background-color: rgb(51,102,204);
}
h1 {color: rgb(255,204,0);}
```

# Stylesheets und FrontPage

- ◆ Inline-Stylesheets nicht unterstützt
- ◆ Internes Stylesheet im Headcontainer
  - n Menü Format/Formatvorlage
  - n Selector Bibliothek
- ◆ Externes Stylesheet – separate Datei
  - n Vorlagen
  - n Layout-Modus (Text-Editor)
  - n Link zu Stylesheet...

*Priorität*

1

2

3

# Stylesheets

## numerische Maßeinheiten

Abkürzung	Bedeutung	Beispiel
pt	Punkt 1pt = 1/72 Zoll	font-size: 12pt;
px	Pixel am Ausgabegerät	margin-right: 60px;
mm	Millimeter	width: 70mm;
cm	Zentimeter	top: 2.54cm
%	Prozent (relativ)	font-size: 200%;

# Stylesheets - Farbangaben

Angabe	Beispiel
hexadezimal (wie HTML): <b>#RRGGBB</b>	<code>h1 {color:#FFCCDD;}</code>
dezimal (nur CSS): <code>rgb(rrr,ggg,bbb)</code>	<code>body { background-color: rgb(51,0,102); }</code>
Farbnamen: 16 Grundfarben	<code>table {background-color: black;}</code>
Farbnamen: Netscape Farbnamen (120)	<code>h3 { color: pink; } hr { color: cornflowerblue; }</code>

à à nachschlagen bei [SELFHTML](http://www.teamone.de) ([www.teamone.de](http://www.teamone.de))

# Formatklassen

im Selektor: Punkt und Klassenbezeichnung

CSS – Angabe (Wirkung)	Anwendung im HTML
<code>.rahmen {border: 1pt solid};</code> (gerahmer Absatz 1pt durchgezogen)	<code>&lt;p class="rahmen"&gt;....&lt;/p&gt;</code>
(nur 1 Wort gerahmt: <code>span</code> Element in HTML)	<code>&lt;p&gt;Nur ein &lt;span class="rahmen"&gt;Wort&lt;/span&gt; gerahmt&lt;/p&gt;</code>
<code>h2.gelb</code> <code>{background-color:#FFFF00;}</code> (Überschrift 2 mit gelbem Hintergrund)	<code>&lt;h2 class="gelb"&gt;ABC...&lt;/h2&gt;</code>

FrontPage: Klassen erscheinen meist im Feld Formatvorlage

# Einige Links mit Javascript

<u>Zurück</u>	<code>&lt;a href= "javascript:history.back()" &gt;Zurück&lt;/a&gt;</code>
<u>Vorwärts</u>	<code>&lt;a href= "javascript:history.forward()" &gt; Vorwärts&lt;/a&gt;</code>
<u>drucken</u>	<code>&lt;a href= "javascript:window.print()" &gt;drucken&lt;/a&gt;</code>
<u>Fenster schließen</u>	<code>&lt;a href= "javascript:window.close()" &gt;Fenster schließen&lt;/a&gt;</code>
<u>Frameset schließen</u>	<code>&lt;a href= "javascript:parent.frames.close()" &gt; Frameset schließen&lt;/a&gt;</code>
<u>Aktualisieren</u>	<code>&lt;a href= "javascript:location.reload()" &gt; &gt;Aktualisieren&lt;/a&gt;</code>