Einführende Informationen zum schnellen Erstellen von einer eigenen Website

D. A. Waldvogel

Inhalt

1 TBZWIKI

Viele Informationen finden Sie auf dem TBZ Wiki-System Modul 307:

http://www.tbzwiki.ch/index.php?title=Modul_307

Konkrete Informationen zu Ihrem Kurs finden Sie im Klassenportal 307_2013:

http://www.tbzwiki.ch/index.php?title=M307_2013

2 HTML Tutorial

http://de.html.net/

- Einführung
  Eine kurze Einführung in das Tutorial und was Sie lernen können werden.
- Lektion 1: Fangen wir an!
  Welche Werkzeuge brauchen Sie um Ihre eigene Webseite zu erstellen.
- Lektion 2: Was ist HTML?
  Verstehen Sie was HTML ist und bedeutet.
- Lektion 3: Elemente and Tags?
  Was sind Elemente und Tags und wofür werden Sie benötigt.
- Lektion 4: Ihre erste Webseite
  Erstellen Sie Ihre erstes HTML-Dokument – die Grundlage aller Ihrer zukünftigen Seiten.
- Lektion 5: Was Sie bisher gelernt haben!
  Wiederholung von dem, was Sie bisher gelernt haben und Ausblick auf das, was Sie in den nächsten Lektionen erwartet.
- Lektion 6: Einige weitere Elemente
  Machen Sie sich mit den sieben meistgebrauchten Elementen vertraut.
- Lektion 7: Attribute
  Lernen Sie den Tags zusätzliche Angaben hinzuzufügen und Kommandos direkt zu formulieren.
- Lektion 8: Links
  Finden Sie heraus, wie Sie Links zu Ihren eigenen und anderen Seiten im Netz setzen können.
- Lektion 9: Bilder
  Entdecken Sie, wie einfach es ist, Bilder in Ihre Webseiten einzubinden.
- Lektion 10: Tabellen
  Konstruieren Sie HTML-Tabellen, um strukturierte Inhalten zu präsentieren.
- Lektion 11: Mehr über Tabellen
  Erstellen Sie noch anspruchsvollere Tabellen.
- Lektion 12: Layout (CSS)
  Verstehen Sie, wie Cascading Style Sheets (CSS) verwendet werden können, um Ihren Seiten ein fantastisches Layout zu verpassen.
- Lektion 13: Seiten ins Netz laden
  Finden Sie heraus, wie Sie Ihre Seiten publizieren können, damit auch andere Leute diese ansehen können.
- Lektion 14: Web-Standards und Validierung
  Lesen Sie über den HTML-Standard und wie Sie die Richtigkeit Ihres Codes kontrollieren können.
- Lektion 15: Abschließende Hinweise
  Ein paar gute Ratschläge für Ihre Webprojekte.

2.1 Kurzreferenz: HTML

http://de.selfhtml.org/navigation/html.htm

2.2 Grundgerüst einer HTML-Datei

http://de.selfhtml.org/html/allgemein/grundgeruest.htm

3 CSS Tutorial

http://www.css4you.de/wscss/css02.html

http://de.html.net/tutorials/css/

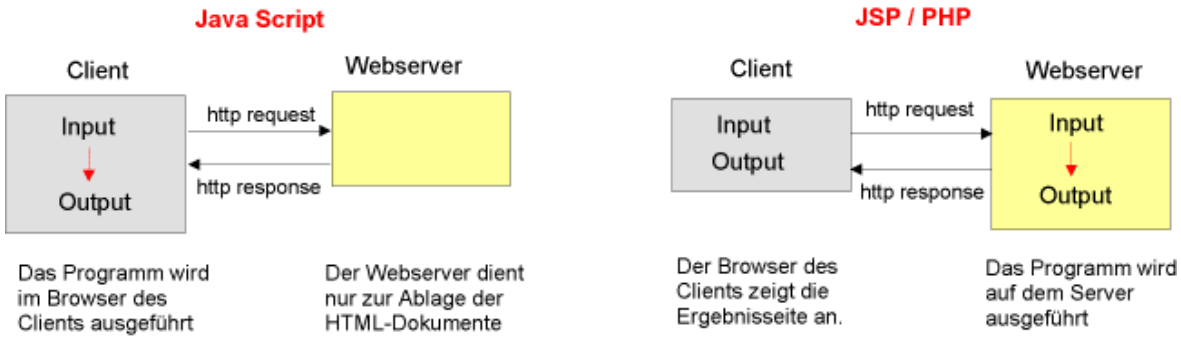
- Einführung
  Eine kurze Einführung in das Tutorial und was Sie lernen können werden.
- Lektion 1: Was ist CSS?
  Ein klein wenig darüber, warum CSS entwickelt wurde und warum es clever ist, CSS anstatt HTML beim layouten und designen zu verwenden.
- Lektion 2: Wie funktioniert CSS?
  Lernen Sie die grundlegende Syntax und erstellen Sie Ihr erstes Stylesheet.
- Lektion 3: Farben und Hintergründe
  Lernen Sie, wie sie Ihrer Webseite Farben und Hintergrundfarben hinzufügen und wie Sie Hintergrundbilder verwenden.
- Lektion 4: Zeichensätze (Fonts)
  In dieser Lektion werden Sie etwas über Schriftarten erfahren und wie Sie diese mit Hilfe von CSS einsetzen können.
- Lektion 5: Text
  In dieser Lektion werden Sie in die unglaublichen Möglichkeiten eingeführt, die Ihnen CSS bietet, um Ihren Text zu layouten.
- Lektion 6: Links
  Wie können Sie Ihren Links tolle und nützliche Effekte geben und wie arbeitet man mit Pseudoklassen.
- Lektion 7: Identifikation und Gruppierung von Elementen (class und id)
  Ein genauer Blick auf die Verwendung von `class` und `id`, um Eigenschaften für bestimmte Elemente festzulegen.
- Lektion 8: Gruppierung von Elementen (span und div)
  Ein genauer Blick auf die Verwendung von `span` und `div`, weil genau diese beiden HTML-Elemente von zentraler Bedeutung für CSS sind.
- Lektion 9: Das Box-Modell
  Das Box-Modell in CSS beschreibt Kästen, die um HTML-Elemente herum generiert werden.
- Lektion 10: Das Box-Modell - Außen- und Innenabstand (margin und padding)
  Ändern Sie die Präsentation von Elementen durch den Einsatz der Eigenschaften `margin` und `padding`.
- Lektion 11: Das Box-Modell - Umrandung (border)
  Lernen Sie die endlosen Möglichkeiten kennen, wenn Sie Umrandungen auf Ihren Seiten einsetzen.
- Lektion 12: Das Box-Modell - Breite und Höhe
  In dieser Lektion schauen wir uns genauer an, wie einfach man Breite und Höhe eines Elementes definieren kann.
- Lektion 13: Schwimmende Positionierung von Elementen (float)
  Ein Element kann mit Hilfe der Eigenschaft `float` auf der linken oder rechten Seite eines Dokumentes positioniert werden.
- Lektion 14: Positionierung von Elementen
  Mit der CSS-Positionierung können Sie ein Element an einer beliebigen Stelle auf Ihrer Seite platzieren.
- Lektion 15: Ebene auf Ebene mit z-index (Layer)
  In dieser Lektion zeigen wir, wie sich verschiedene Elemente überlappen können.
- Lektion 16: Web-Standards und Validierung
  Diese letzte Lektion ist über W3C-Standards und wie Sie prüfen können, ob Ihr CSS korrekt ist.

4 Dynamische Webseiten

Bei statischen Webseiten werden HTML-Dateien auf dem Server abgelegt und zur Darstellung unverändert vom Server zum Client übertragen.

Bei dynamischen Webseiten werden die Seiteninhalte erst zur Laufzeit, d.h. beim Aufrufen der Webseite generiert. Die Webseiten enthalten neben dem HTML-Code weitere Programmanweisungen, die auf dem Webserver ausgeführt werden. Dabei kann der Server zum Aufbau der Seite Informationen von verschiedenen Quellen, beispielsweise einer Datenbank beziehen.

Verschiedene Technologien



Java Script

Java Script ist ein Scriptsprache, die als Erweiterung von HTML dient. Java Script - Anweisungen ermöglichen es, Elemente einer Webseite durch Einwirken des Benutzers zu ändern.
Die Programme werden als Quelltext im Browser des Clients zur Laufzeit interpretiert. Der Programm-Code ist dadurch als Quellentext für jedermann ersichtlich, ein Vorteil für Lernende, aber ein Nachteil für proffessionelle Programmierer. Java Script eignet sich daher nicht für komplexe Applikationen. Sie verfügt aber über unzählige Event-Handler- und Window-Methoden, mit denen sich Web-Seiten optimieren lassen.

JSP (Java Server Pages)

JSP's sind HTML-Seiten, in denen Java Code eingebettet ist. HTML-Code dient zur Darstellung der Webseite, mit Java Code wird der Inhalt und die Anwendungslogik beschrieben. Im Unterschied zum Java Script, wo der Programmcode im Browser des Clients ausgeführt wird, wird der JSP-Programmcode auf dem Webserver kompiliert und ausgeführt. Die fertiggestaltete Antwortseite wird über HTTP an den Browser zurückgeschickt. Da Java-Code von JSP auf umfangreiche Java-Bibliotheken zugreifen kann, stellt JSP im vergleich zu anderen Script-Sprachen viel mächtigeres und eleganteres Instrumentarium zur Verfügung und eignet sich gut für komplexe Applikationen und Online-Datenbanken. Für Arbeit mit JSP sind Grundkenntisse des Programmierens mit Java erforderlich.

4.1 Client-seitig mit JavaScript

http://clab1.phbern.ch/webdesign/index.php?inhalt_links=jscript/nav_jscript.inc.php&inhalt_mitte=jscript/home.inc.php

JavaScript ist eine objektbasierte Skriptsprache. Obwohl der Name etwas anders suggeriert, hat JavaScript sehr wenig mit der Programmiersprache Java zu tun. JavaScripts werden direkt im HTML-Dokument oder in einer

separaten Datei notiert. Sie werden zur Laufzeit vom Web-Browser interpretiert. Dazu besitzen moderne Web-Browser entsprechende Interpreter-Software. JavaScripts ermöglichen es HTML Dokumente interaktiv zu machen, d.h. die Eingaben von Benutzern zu registrieren und zu verarbeiten.
Die JavaScript Programme können mit einem Texteditor oder mit einem Webdesign-Tool geschrieben und mit der Erweiterung ".htm" oder ".html" gespeichert. Zum Testen der Programme wird ein Browser benutzt. JavaScript Programme sind ohne Anpassung lauffähig auf allen Systemen (Windows, Mac, Linux). Leider müssen häufig für den Microsoft Internet Explorer gewissen Anpassungen vorgenommen werden, das sich dieser Browser oft anders benimmt als alle übrigen Browser (Firefox, Safari, Opera).
JavaScript läuft in einer so genannten "Sandbox". Das ist eine Art Sicherheitskäfig, in dem die Programmiersprache eingesperrt ist. Sie wird dabei um typische Möglichkeiten anderer Programmiersprachen beschnitten, vor allem um die Möglichkeit, beliebig Daten aus Dateien zu lesen und in Dateien schreiben zu können. So wird verhindert, dass JavaScript-Programmierer auf den Rechnern von Benutzern Unfug treiben können. Nach einigen Anfangsproblemen wurde die Browser Software stark verbessert. Eine Daktivierung von JavaScript ist nicht mehr notwendig.

Typische Anwendungsgebiete von JavaScript

- Überprüfung (Validierung) von Formulareingaben vor dem Absenden
- Rollover-Grafiken, die beim Überfahren mit der Maus wechseln
- Banner oder Laufschriften
- dynamische Manipulation von Webseiten über das Document Object Model
- Senden und Empfangen von Daten, ohne dass der Browser die Seite neu laden muss ().

Nachteile:

- Schwache Typisierung von Variablen
- Der Programm-Code ist als Quellentext für jedermann ersichtlich und kopierbar, ein Vorteil für Lernende, aber ein Nachteil für proffessionelle Programmierer
- JavaScript-Programme werden auf dem Rechner des Benutzers ausgeführt. Deshalb deaktivieren einige Internet-Benutzer JavaScript in ihrem Browser aus Angst vor Hackerangriffen.

4.2 Einbinden von JavaScript in HTML-Dokumente

Ein JavaScript-Code beginnt immer mit dem Tag `<SCRIPT LANGUAGE = "JavaScript">` und endet mit dem Tag `</SCRIPT>`.

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

```
<SCRIPT LANGUAGE="JavaScript">
  name = prompt("Geben sie ihren Namen an "," ");
  document.write("Hallo " + name);
</SCRIPT>
</BODY>
</HTML>
```

In einer separaten Datei

```
<HTML>
<HEAD>
</HEAD>
<BODY>
<SCRIPT LANGUAGE = "JavaScript" src = "eingabe.js"></SCRIPT>
</BODY>
</HTML>
```

```
(eingabe.js)
name = prompt("Geben sie ihren Namen an "," ");
document.write("Hallo " + name)
```

Die Datei *eingabe.js* sollte im gleichen Verzeichnis gespeichert sein, wie die HTML Datei, anderseits muss der Pfad zum gewünschten Verzeichnis angegeben werden.

4.3 Eingabe überprüfen

Die Eingabe in den Formularfeldern muss in der Regel überprüft werden, da leere oder fehlerhafte Eingaben bei der Formularauswertung Fehler verursachen können.
alert("....") erzeugt eine Dialogbox, in der eine Fehlermeldung angezeigt werden kann.

Beispiel 1: Beispiel anzeigen (checkText.htm)
Die Eingabe im Textfeld wird auf "leer" überprüft. Falls die Eingabe leer ist, erscheint ein *Alert*-Fenster mit einer Fehlermeldung.

```
<HTML>

<HEAD>

<SCRIPT LANGUAGE = "JavaScript">
```

```
  function antwort()
  {
    name = document.form1.eingabe.value;
    if (name.length == 0)
      alert("Bitte Name eingeben!");
    else
      document.form1.ausgabe.value = "Sie heissen " + name + ".";
  }
</SCRIPT>

</HEAD>

<BODY>

<FORM NAME = "form1">
  <P>Name: <INPUT TYPE = "text" NAME = "eingabe" VALUE = ""SIZE = 15 MAXLENGTH =
10></P>
  <P><INPUT TYPE = "button" VALUE = "Hier klicken"onClick = "antwort();"></P>
  <P><INPUT TYPE = "text" NAME = "ausgabe" VALUE = "" SIZE = 40></P>
</FORM>

</BODY>

</HTML>
```

Erklärungen zum Programmcode:
name.length gibt die Anzahl Zeichen des Textes, der in der Variable *name* gespeicher wurde. Wurde im Textfeld Name kein Name eingegeben, ist name.length = 0)

Beispiel 2: Beispiel anzeigen (checkZahl.html)
Ueberprüft, ob eine Zahl eingegeben wurde.

```
<HTML>

<HEAD>

<SCRIPT LANGUAGE = "JavaScript">

  function antwort()
  {
    zahl = document.form1.eingabe.value;
    z = parseInt(zahl);
    if (zahl.length == 0)
      alert("Bitte Alter eingeben!");
    else
      if (isNaN(z))
        alert("Die Zahleneingabe ist ungültig!");
      else
        document.form1.ausgabe.value = "Sie sind " + z + "Jahre alt.";
```

```
  }
</SCRIPT>

</HEAD>

<BODY>

<FORM NAME = "form1">
  <P>Zahl: <INPUT TYPE = "text" zahl = "eingabe" VALUE = ""SIZE = 15 MAXLENGTH =
10></P>
  <P><INPUT TYPE = "button" VALUE = "Hier klicken"onClick = "antwort();"></P>
  <P><INPUT TYPE = "text" NAME = "ausgabe" VALUE = "" SIZE = 40></P>
</FORM>

</BODY>

</HTML>
```

4.4 Server-seitig PHP

PHP ist eine in HTML-eingebettete Scriptsprache. Ihre Stärke liegt in der Gestaltung von dynamischen Webseiten und im Programmieren von online Datenbank-Applikationen mit MySQL. PHP ist wesentlich einfacher als JSP und verfügt über eine grosse Anzahl von vordefinierten Funktionen, die die Programmierarbeit vereinfachen.
PHP-Scripte werden auf dem Server ausgeführt, sie werden nicht im Vorfeld kompiliert wie JSP, sondern bei jedem HTML-Request neu durch den sog. PHP-Parser auf dem Webserver. Dadurch sind PHP-Applikationen langsammer als JSP, dies spielt aber erst bei umfangreicheren Applikationen eine Rolle.

Ajax (Asynchronous JavaScript and XML)
Ajax ist eine Technologie, die durch die asynchrone Übertragung der Daten zwischen Client und Server charakterisiert wird. Ajax-Anwendungen ermöglichen es, innerhalb einer HTML-Seite Anfrage durchzuführen ohne die gesamte Webseiten neu laden zu müssen. Die zusätzlichen Daten werden beim Bedarf vom Server angefordert und nur Teile der aktuellen Webseite aktualisiert. Zum Programmieren von Ajax Webseiten wird eine Kombination mehrerer Webtechniken verwendet: JavaScript, XML, CSS, PHP usw.

4.5 PHP Tutorial

http://de.html.net/tutorials/php/

- Einführung
  Eine kurze Einführung in das Tutorial und was Sie hier lernen können.
- Lektion 1: Was ist PHP
  Ein wenig darüber, wie PHP funktioniert und was es bedeutet, dass PHP eine serverseitige Technologie ist und was Sie in den folgenden Lektion lernen werden.
- Lektion 2: Server
  Wir beleuchten verschiedene Möglichkeiten, PHP auf dem eigenen Computer oder einem gehosteten Webserver zu verwenden.
- Lektion 3: Ihre erste PHP-Seite
  In dieser Lektion erstellen Sie Ihre erste, sehr einfache PHP-Seite. Hier können Sie auch testen, ob Ihr Server für das Ausführen von PHP konfiguriert ist.
- Lektion 4: Arbeiten mit Zeit und Daten
  Einführung in Functions (Funktionen), die zum Arbeiten mit Zeiten und Daten genutzt werden können.
- Lektion 5: Schleifen
  Schleifen können Teile eines Skriptes wiederholen. In dieser Lektion sehen wir uns die *while*- und *for*-Schleifen genauer an.
- Lektion 6: Conditions (Bedingungen)
  Sog. Conditions können dazu verwendet werden, die Ausführung eines PHP-Skriptes zu kontrollieren. Wir behandeln hier *if ... elseIf ... else ...* und *switch ... case*.
- Lektion 7: Skripte erläutern/kommentieren
  Kommentare machen Ihre PHP-Skripte klarer und einfacher verständlich. Erläuterungen können eine große Hilfe sein, wenn Sie oder jemand anderes, später Änderungen an Ihrem Code vornehmen muss.
- Lektion 8: Arrays (Felder)
  In dieser Lektion lernen Sie, was ein sog. Array ist, wie und wofür er genutzt werden kann.
- Lektion 9: Functions (Funktionen)
  In den vorherigen Lektionen haben Sie gelernt, unterschiedliche PHP-eigene Functions zu benutzen. Jetzt werden Sie lernen, Ihre eigenen zu erstellen.
- Lektion 10: Variablen über die URL weitergeben
  Lernen Sie, wie man Variablen und deren Werte von einer Seite an eine andere mit Hilfe des sog. 'HTTP query string' weitergibt.

- Lektion 11: Verarbeitung von Formular-Variablen
  Interaktive Webseiten verlangen Eingaben von Nutzern. Eine der üblichsten Arten, an diese Eingaben zu gelangen, sind Formulare.
- Lektion 12: Sessions (Sitzungen)
  Sog. Sessions können dazu verwendet werden, Informationen während eines Besuches eines Nutzers auf Ihrer Seite zu speichern und wieder abzurufen.
- Lektion 13: Cookies
  Cookies können verwendet werden, um Informationen über einen Nutzer über einen Besuch hinaus zu speichern und (z.B. beim nächsten Besuch) wieder abzurufen.
- Lektion 14: Filesystem-Functions (Dateisystem-Funktionen)
  Hiermit können Sie auf das Dateisystem des Servers zugreifen. Dies erlaubt Ihnen, Dateien, Ordner oder Laufwerke mit PHP-Scripten zu verändern.
- Lektion 15: Aus einer Text-Datei lesen
  In dieser Lektion werden wir die Filesystem-Functions benutzen, um aus einer Text-Datei zu lesen. Text-Dateien können sehr nützlich sein, um Daten verschiedener Art zu speichern.
- Lektion 16: In eine Text-Datei schreiben
  Diese Lektion handelt von den Möglichkeiten, mit den Filesystem-Functions in eine Text-Datei zu schreiben. Text-Dateien können sehr nützlich sein, um Daten verschiedener Art zu speichern.
- Lektion 17: Datenbanken
  In diesem Tutorial verwenden wir MySQL-Datenbanken. MySQL ist das am häufigsten verwendete Datenbanksystem, wenn man anfängt Datenbanken mit PHP nutzen wollen.
- Lektion 18: Datenbanken und Tabellen erstellen
  In dieser Lektion betrachten wir zwei Arten, wie man Datenbanken und Tabellen anlegen kann. Zuerst, wie es mit PHP gemacht wird und dann mit dem benutzerfreundlicheren Software-Tool: PhpMyAdmin.
- Lektion 19: Daten in eine Datenbank einfügen
  Lernen Sie, wie Sie SQL-Anweisungen benutzen, um Daten in eine Datenbank einzufügen. Wir betrachten auch Datentypen und weisen auf die häufigsten Anfängerfehler hin.
- Lektion 20: Daten aus einer Datenbank abrufen
  Lernen Sie, wie man eine SQL-Anfrage nutzt, um Daten aus einer Datenbank abzurufen.
- Lektion 21: Daten aus einer Datenbank löschen
  Lernen Sie, wie man mit SQL Datensätze aus einer Datenbank löscht.
- Lektion 22: Daten in einer Datenbank aktualisieren/ändern
  In dieser letzten Lektion werden Sie lernen, wie man Daten mit einer SQL-Anweisung aktualisiert/ändert.

4.6 Empfehlenswertes zum PHP-Tutorial

http://www.php-kurs.com/

Besonders die Kapitel:

- PHP für Fortgeschrittene (Sessions, require, include)
- Objektorientierte Programmierung in PHP
- MySQL Tutorial Datenbank (Gästebuch, MySQL sauber schreiben)
- PHP und Sicherheit (Session Hijacking, SQL-Injection, Cross-Site Scripting

http://www.php-kurs.info/tutorial-sql_injection.html

http://www.php-kurs.info/tutorial-sessions_cookies.html

http://www.php-kurs.info/tutorial-variablen_uebergeben_include.html

5 Layouten ohne Tabellen

http://www.css4you.de/wslayout1/index.html#example

5.1 Warum keine Tabelle?

Obwohl Tabellen nie zum Layouten von Seiten vorgesehen waren, wurden und werden sie meistens verwendet. Die Gründe liegen auf der Hand: Unerfahrene Webdesigner können mit einem WYSIWYG-Editor sehr schnell ein Design entwickeln. Mit CSS und div-Container ist das ohne Kenntnisse der Materie etwas problematischer. Weiterhin unterstützen erst die modernen Browser der letzten 1-2 Jahre CSS soweit, dass sich ein Umstieg lohnt. Da viele Seiten im Web noch nicht überarbeitet sind, finden sich auch noch viele Seiten, die mit Tabellen layoutet wurden. Dabei ist ein Umstieg mittlerweile durchaus lohnenswert.

Betrachtet man den Quellcode von Webseiten, die mit Tabellen erstellt wurden, so findet man häufig komplexe Strukturen verschachtelter Tabellen. Das gleiche Layout, erzeugt mit CSS, kann durchaus bis zu 40% weniger Quellcode bedeuten. Jeder Modembesitzer wird dankbar sein, da es spürbare Geschwindigkeitsvorteile bringt. Auf Seite der Webseiten-Betreiber bedeuten kleinere Dateien geringeres Transfervolumen. Reduziert man die Größe einer Datei von z.B. 5kB auf 3kB und geht man von 100.000 Besucher pro Monat aus, ergibt sich ein verringertes Transfer-Volumen von 2,4 GB pro Jahr für nur eine Datei. Hochgerechnet auf eine ganze Seite kann sich eine Umstellung durch diese Kostenersparnis durchaus rechnen.

Einfache Anpassung für unterschiedliche Ausgabemedien:

Die Zahl der unterschiedlichen Ausgabemedien steigt. Ob Bildschirm, Drucker, PDA, Handy, Beamer, Textbrowser oder Sprachausgabe, eine Webseite kann von allen Medien angezeigt, bzw. ausgedruckt werden. Und mit tabellenlosem Layout hast du die Möglichkeit, das Layout an die unterschiedlichen Ausgabemedien optimal anzupassen.

Private Homepages

Die oben genannten Argumente treffen sicherlich nicht alle auf private Homepages zu. Auf der anderen Seite steht hier der Spass am Schreiben der Webseite im Vordergrund. Also solltest du doch an diesem Thema interessiert sein und etwas tiefer einsteigen, oder?

5.2 Navigation

Und hier noch ein Link für CSS, HTML und Navigation - schlicht klar einfach!

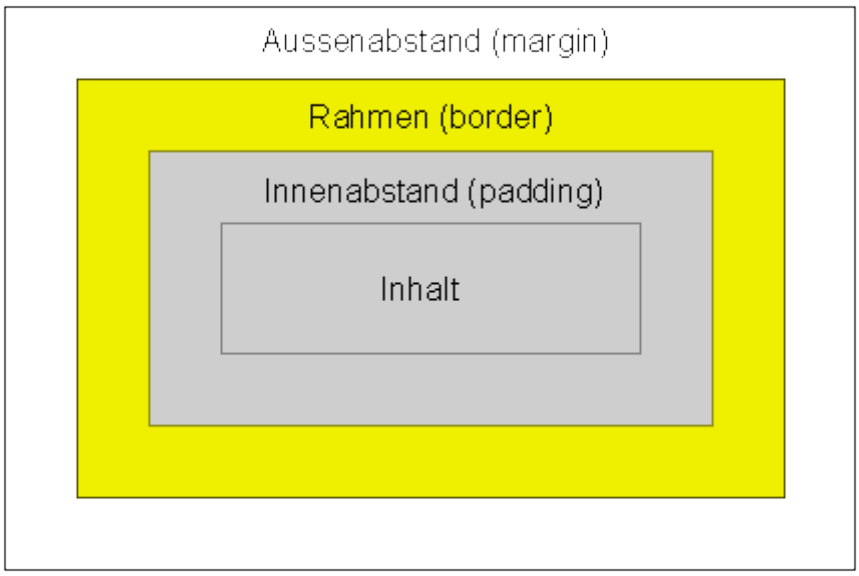
ex0005.html

http://www.css4you.de/wslayout1/index.html#example

5.3 Das Boxmodell

http://www.css4you.de/wsboxmodell/index.html#padding

Für jedes Element einer Seite wird nach den CSS-Regeln ein rechteckiger Bereich reserviert, der in dem sog. Boxmodell beschrieben ist. Dieser Bereich besteht aus dem eigentlichen Inhalt, einem Innenabstand zu dem Rahmen des Elements, dem Rahmen und dem Abstand zu anderen Elementen, die auf einer Seite zu finden sind. Die folgende Grafik soll diese veranschaulichen:



Der Inhalt

Der innere Bereich, der aus dem Inhalt, wie z.B. Text und Grafiken besteht, stellt ein Rechteck dar. Die Größe kann von unterschiedlichen Faktoren abhängen: Wenn keine Angaben zur Breite und Höhe gemacht sind, bestimmt der Inhalt die Größe und Oder die width- und height-Eigenschaften legen die Elementabmessungen fest.

6 Formulare und Auswertung

6.1 Kurzreferenz: Formulare

http://de.selfhtml.org/navigation/html.htm#formulare

http://www.uni-duesseldorf.de/~hilberer/html9.php3

Datenübertragung per E-Mail

Im einleitenden <FORM>-Tag gibt man mit ACTION="..." an, was mit den ausgefüllten Formulardaten passieren soll, wenn das Formular abgeschickt wird.

Im einfachsten Fall geschieht die Übertragung durch die Mail-Funktion des Browsers. Dazu gibt man eine E-Mail-Adresse mit vorangestelltem MAILTO: ein, z.B. ACTION="MAILTO:"name@uni-xyz.de"". An diese Adresse werden die Formular-Angaben geschickt.

Dies funktioniert aber nur dann, wenn der Browser für den Mail-Versand richtig konfiguriert ist. Selbst dann gibt es mit manchen Browsern Probleme.

"Ein anderes Problem besteht darin, dass die Formulardaten beim Abschicken per Voreinstellung nach einem bestimmten Mime-Type kodiert werden, dem Mime-Typ application/x-www-form-urlencoded. Dabei werden alle Leerzeichen, verschiedene Sonderzeichen und Umlaute durch spezielle Zeichenfolgen ersetzt. So lautet beispielsweise der Satz Danke für die Hilfe! nach der Umwandlung: Danke+f%FCr+die+Hilfe%21. Für Menschen ist das ziemlich ungenießbarer Lesestoff. Um die Kodierung zu verhindern, können Sie im einleitenden <form>-

Tag zwar die Angabe enctype="text/plain" angeben. Von Anwendern, deren WWW-Browser diese Angabe jedoch nicht interpretiert, werden Sie dennoch kodierte Formulardaten erhalten. Um kodierte Formulardaten nachträglich in lesbaren Text zu dekodieren, können Sie ein Hilfsprogramm wie WebParse für Windows verwenden."

Als METHOD= kommt bei E-Mail-Übertragung nur POST in Frage.

Also:

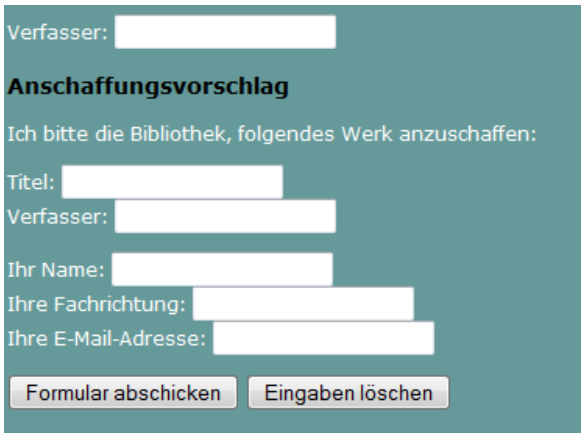
<FORM ACTION="MAILTO:th@hilberer.de" METHOD=POST

ENCTYPE="text/plain">

</FORM>

http://www.uni-duesseldorf.de/~hilberer/html9.php3

Eingabefelder:




```
<H3 ALIGN=CENTER>Anschaffungsvorschlag</H3>
Ich bitte die Bibliothek, folgendes Werk anzuschaffen:<P>
<FORM ACTION="MAILTO:th@hilberer.de" METHOD=POST
ENCTYPE="text/plain">
Titel: <INPUT TYPE="TEXT" NAME="Titel"><BR>
Verfasser: <INPUT TYPE="TEXT" NAME="Verfasser"><P>
Ihr Name: <INPUT TYPE="TEXT" NAME="Name"><BR>
Ihre Fachrichtung: <INPUT TYPE="TEXT" NAME="Fach"><BR>
Ihre E-Mail-Adresse: <INPUT TYPE="TEXT" NAME="EMail"><P>
<INPUT TYPE="SUBMIT" VALUE="Formular abschicken">
<INPUT TYPE="RESET" VALUE="Eingaben löschen">
</FORM>
```

7 Reguläre Ausdrücke

http://www.infos24.de/javascripte/handbuch/7_js_regular_expression.htm

```
<script language=javascript>
zeichenketta="Der eine fragt, was kommt danach, der andere fragt nur, ist es recht, so also unterschei-
det sich, der Freie von dem Knecht";
function testen1()
{
if(speicher=zeichenketta.match("was(.....).........(....).....(.......)"))
{
alert("Treffer bei Zeichen Nr. : "+speicher.index+"\nDie gesamte Zeichenkette: "+speicher.input+"\nDie
gesamte reg.exp: "+speicher[0]+"\nerste Klammer:"+speicher[1]+"\nzweite Klammer: "+speicher[2]);
}
else
{
alert("Das gesuchte Wort ist nicht enthalten");
}
}
</script>
```

8 Grundgerüst generieren lassen

http://www.webmasterarchiv.com/online-grid-layout-generatoren/

9 Verwenden von Bildern

Um Ihre eigenen Bilder zu erstellen, benötigen Sie ein Bildbearbeitungsprogramm. Ein Bildbearbeitungsprogramm ist eines der wichtigsten Werkzeuge um schöne Webseiten zu erstellen.

Leider ist weder im Lieferumfang von Windows, noch bei anderen Betriebssystemen, ein annehmbares Bildbearbeitungsprogramm inklusive.

Für unsere Zwecke genügt es, wenn Sie sich Irfan View, ein exzellentes Bildbearbeitungsprogramm herunterladen. Irfan View ist sogenannte Freeware und kostet aus diesem Grunde nichts.

Oder Sie borgen sich Bilder von anderen Seiten, indem Sie diese downloaden. Aber bitte seien Sie vorsichtig, dass Sie mit dem Download und einer späteren Verwendung auf Ihren Seiten keinerlei Copyrights verletzen. Trotz alle dem ist es sicherlich nützlich zu wissen, wie man Bilder aus dem Internet herunterlädt. So wird’s gemacht:

Klick mit der rechten Maustaste auf das Bild (irgendein Bild im Internet)

Wählen Sie “Bild speichern als …” in dem Menü, welches erscheint.

Suchen Sie einen Ordner, in den Sie das Bild speichern möchten und klicken Sie auf “Speichern”.

10 Webseiten validieren lassen

http://validator.w3.org/check?uri=http://www.css4you.de/wslayout1/index.html

http://validator.w3.org/

D.Waldvogel, 30. Mai 2013
M307_Einfuehrungs_Info_V1.0.docx