

Thomas Wenzlaff

Beispiele zur Kurz-Dokumentation zu HTML 4

für den

Hobby-Programmierer



Hinweise zum Dokument

Autor: Thomas Wenzlaff
www.twseite.de

Stand des Dokumentes: Samstag, 24. März 2007

Rechtswahrung, Haftung und Verbesserungsvorschläge:

Das Dokument richtet sich an den Nutzerkreis der Hobby-Programmierer.

Die Informationen in diesem Produkt werden ohne Rücksicht auf einen eventuellen Patentschutz veröffentlicht. Warennamen, Hardware- und Softwarebezeichnungen werden ohne Gewährleistung der freien Verwendbarkeit benutzt und gehören - wenn eingetragen - dem jeweiligen Eigentümer, oder sollten als solche betrachtet werden.

Der Autor kann für fehlerhafte Angaben und deren Folgen weder eine juristische Verantwortung noch irgendeine Haftung übernehmen.

Für Verbesserungsvorschläge und Hinweise ist der Autor dankbar.

Berlin, 24. März 2007



Hinweise zur Kodierung von Quelltext in diesem Dokument

Die in diesem Dokument für die Kodierung von HTML-Quelltext verwendeten Einschachtelungen von HTML-Elementen wie Tags, Scripten usw. dienen ausschließlich der Übersichtlichkeit des Quelltextes.

HTML lässt Einschachtelungen wie in Java LEIDER nicht zu, da die Position der HTML-Elemente DIREKT in das Layout des HTML-Dokumentes übernommen werden.

Dazu ein Beispiel:

Für die Kodierung einer Tabelle wäre es übersichtlicher, jede Tabellenzelle als 1 Quelltext-Zeile zu kodieren. Eine Tabellenzelle wird syntaktisch an der Tag-Eingrenzung erkannt. Alle dazwischenliegenden Zeichen, also auch ein durch die Return-Taste (Enter-Taste) zum Zweck der Quelltextgestaltung erzeugter Zeilenumbruch, werden zum z.T. unerwünschten Zelleninhalt. Leider ist es damit eventuell nötig, unübersichtliche Quelltextzeilen mit Tag-Folgen zu kodieren.

Wer strukturiert programmieren möchte, sollte sich mit JavaScript beschäftigen.



Inhaltsverzeichnis

1.	BROWSER	12
1.1.	Browser allgemein	12
1.1.1.	Browserdifferenzen bei pixelgenauer Darstellung vermeiden	12
1.1.2.	Kontextmenü für rechte Maus sperren	12
1.2.	Microsoft Internet Explorer	12
1.2.1.	Autoausfüllen	12
1.2.2.	Formular mit Pulldown-Auswahlfeld mit farbigen Einträgen	13
1.2.3.	Linklayout verändern	13
1.2.3.1.	Linkunterstreichung abschalten	13
1.2.3.2.	Standardlinkfarben ändern	13
1.2.3.3.	Link-Layout verändern wenn Maus über Link fährt	13
1.2.3.4.	Typen von Links-Layouts per CSS definieren	13
1.2.3.5.	Cursor für einen Link ändern	13
2.	BUTTON	14
2.1.	Buttonbeschriftung mit mehrzeiligem Text	14
3.	FORMULAR-TEXTAREA OHNE ROLLBALKEN (SCROLLBALKEN)	14
4.	GRAFIK	14
4.1.	Viereck-Grafik	14
4.1.1.	Viereck-Grafik zerlegen in kleine Elemente und als blinde Tabelle zusammenbauen	14
4.1.2.	Viereck-Grafik zerlegen in kleine Elemente, als blinde Tabelle zusammenbauen und jedem Teil 1 Link zuweisen	14
4.2.	Bilderrahmen aus verschachtelten Tabellen	15
4.3.	Schnelleres Laden auch bei abgeschalteter Grafikanzeige	15
4.4.	Filter für HTML-Elemente ab IE 5.5	15
4.4.1.	Prinzip der Kodierung von Filtern	15
4.4.2.	Alpha-Filter für HTML-Grafik-Elemente ab IE 5.5	16
4.4.2.1.	Beispiel für transparentes INPUT-Element	16
4.4.2.2.	Beispiel für transparentes Vordergrund-Bild auf dem Hintergrundbild des Dokumentes	16
4.4.2.3.	Beispiel für transparente Überlagerung zweier Bilder	17
4.4.2.4.	Beispiele für Aus- und Einblende eines Bildes	17
4.4.2.5.	Beispiel für Aus- und Einblende von Bildern mit Bildwechsel	17
4.4.3.	Shadow-Filter für HTML-Grafik-Elemente ab IE 5.5	20
4.4.3.1.	Beispiel für Schatten eines Bildes	20
4.4.3.2.	Beispiel für 3D-Effekt eines Bildes	20
4.5.	Grafik zoomen ab IE 5.5	20
5.	TABELLEN	21
5.1.	Tabellen allgemein	21
5.1.1.	Laden einer großen Tabelle	21
5.1.2.	Zerlegen einer großen Tabelle in kleinere Tabellen	21
5.1.3.	Keine Abstände zwischen Tabellen	21
5.1.4.	Verschachtelte Tabellen	21
5.1.5.	Blinde Tabelle (ohne Rahmen) mit zellenweisem Hintergrund	21
5.2.	Zelle	22
5.2.1.	Zelle mit exakten Dimensionen bei älteren Browsern	22
5.2.2.	Leere Zelle mit Hintergrund	22
5.2.3.	Zelle mit Schattengrafik passend zum Hintergrund	22
5.2.4.	Ausgabe von Texten innerhalb einer Tabellenzelle Textlage in der Tabellenzelle: oben, unten UND mittig	22
5.3.	Beispiel: Kalender	23



6.	TEXT	30
6.1.	vertikale Linien zwischen Texten	30
6.2.	Wortlogo unabhängig von aktueller Bildschirmauflösung des Users stets zentriert im Browserfenster anzeigen	30
6.3.	CSS schwebender Text von rechts nach links bei 640 Pixel Bildschirmbreite	31
7.	URL	31
7.1.	Umleitung auf andere Url	31
7.2.	Mehrfachverweis auf ein und dieselbe Url innerhalb einer HTML-Seite	31
7.3.	Möglichkeiten des Laden eines Dokumentes	31
7.3.1.	Laden eines fremden Dokumentes ohne Framedarstellung	31
7.3.2.	Eigenes Dokument wird durch fremde Webseite geladen	32
7.3.2.1.	CopyRight-Meldung auf fremden Host erzeugen	32
7.3.2.2.	Framedarstellung der fremden Webseite sofort und ohne Bildschirmmeldung aufheben	33
7.3.3.	Frame-Dokument mit zugehörigem Frameset laden	33
7.3.4.	Mehrere Frameinhalte gleichzeitig ändern	34
8.	ZUGRIFFSCHUTZ	34
8.1.	robots.txt und Suchmaschinen	34
8.1.1.	robots.txt und ihre Lage auf dem Server	34
8.1.2.	robots.txt die nicht auf dem Server vorhanden ist (Scan-Standard)	34
8.1.3.	robots.txt und Aufbau	35
8.1.3.1.	robots.txt als Zeilenfolge-Script erstellen	35
8.1.3.2.	robots.txt und ihre Blöcke	35
8.1.3.2.1.	robots.txt-Block: Aufbau Zeile 1 (Zulassen bzw. Sperren von Suchmaschinen)	35
8.1.3.2.1.1.	robots.txt und Zeile 1: ALLE Suchmaschinen dürfen scannen	35
8.1.3.2.1.2.	robots.txt und Zeile 1: KEINE Suchmaschine darf scannen	35
8.1.3.2.1.3.	robots.txt und Zeile 1: Eine bestimmte Suchmaschinen darf scannen	35
8.1.3.2.1.4.	robots.txt und Zeile 1: Eine bestimmte Suchmaschinen darf NICHT scannen	35
8.1.3.2.2.	robots.txt-Block: Aufbau ab Zeile 2	35
8.1.3.2.2.1.	robots.txt und ab Zeile 2: Alle Daten auf dem Server dürfen gescannt werden	35
8.1.3.2.2.2.	robots.txt und ab Zeile 2: Alle Daten auf dem Server dürfen NICHT gescannt werden	35
8.1.3.2.2.3.	robots.txt und ab Zeile 2: Bestimmte Daten auf dem Server dürfen NICHT gescannt werden	35
8.1.3.2.2.3.1.	robots.txt und ab Zeile 2: Bestimmte Verzeichnisse auf dem Server dürfen NICHT gescannt werden	35
8.1.3.2.2.3.2.	robots.txt und ab Zeile 2: Bestimmte HTML-Dateien auf dem Server dürfen NICHT gescannt werden	36
8.1.3.2.2.4.	robots.txt und ab Zeile 2: NUR bestimmte Daten auf dem Server dürfen gescannt werden	36
8.1.3.3.	robots.txt und Beispiele	36
8.2.	.htaccess und .htpasswd und Passwordschutz	36
9.	WEBCAM	38
10.	POPUP-FENSTER AB IE 5.5	38
11.	MP3-SOUND-DATEI UND STREAMING AUDIO	39
12.	SERVER ERROR-MELDUNG DURCH EIGENE MELDUNG	39
13.	FAVORITEN-GRAFIK FAVICON.ICO DER WEBSEITE	39
14.	INDEX	41



Hinweise zum Einrichten eines virtuellen Hosts per Apache-HTTP-Server:

Anstelle des HTTP-Webservers vom Internetprovider kann der eigene PC als Test-Server fungieren, um z.B. die Webseite so zu testen, als wäre sie gerade online auf dem Webserver.

Der Hobbyprogrammierer will u.a. kostengünstig testen, also bieten sich die kostenlosen HTTP-Server an, die einen virtuellen Host anhand eines beliebigen Ordners auf der Festplatte erzeugen können, wobei Local Host (127.0.0.1) als virtueller Host einrichtbar sein muss (der nicht von der Firewall-Software des PC allein verwaltet sein darf) und die zu testende Webseite im Browser per Domainnamen aktivierbar sein muss (anstelle der Eingabe von 127.0.0.1). Man google, um festzustellen, welche Produkte diesen Kriterien entsprechen. - Abkürzend fällt die Wahl nicht zufällig auf den HTTP-Server von Apache (www.apache.org), welcher ziemlich schwierig zu konfigurieren ist, wenn mehr als nur localhost genutzt werden soll (Für Hobbyzwecke reicht localhost aus).

Der Hobbyprogrammierer, welcher für den Microsoft Internet Explorer (ab IE 7 heißt der Windows Internet Explorer) in seinen Varianten je nach Windows-Version programmieren will, wird definitiv folgende Probleme bekommen:

Microsoft lässt u.a. die Installation des Internet Explorers in parallelen Versionen nicht zu, obwohl Browserversionen nachweislich nicht kompatibel sind und Microsoft Browserversionen innerhalb Windowsversionen supportet werden. Daher benötigt man pro Version des Internet Explorers eine Windows-Installation. Pro Windows-Installation wird eine Windows-Lizenz fällig, auch wenn auf anderer Festplatte am ansonsten identischen PC installiert wird (Windows-Online-Update erkennt solche Doppelversionen und verweigert den Support). Auch wer Windows unter VM emulieren will (oder auf Apple mit Intel-Technik Windows parallel mit Apple nutzen will), benötigt eine Lizenz. Mit anderen Worten: Auch wenn die Mehrfachinstallation nicht parallel nutzbar wäre, sondern immer nur genau 1, will Microsoft Geld haben - nicht umsonst ist Microsoft-Chef so beliebt wie reich und nicht umsonst migrieren immer mehr IT-Anwender zu Linux-Derivaten, die fast identische Browser bezüglich Windows haben. Ergo, der Hobbyprogrammierer wird wohl sämtliche Bekannte mit Testwünschen nerven, oder illegal testen, oder auf andere Browser-Hersteller und deren HTML- sowie Script-Versionen ausweichen, die nicht nur Parallelinstallationen des Browsers zulassen (z.B. Opera unter Windows), sondern auch noch ziemlich gut kompatibel sind (weil identische Scriptmaschine nutzend). Dass Microsoft eventuell keinen kostenlosen HTTP-Server anbietet (der ansonsten auch noch Javascript, Active-X-Control- Aufrufe per Script, DirectX-Zugriff kennen müsste), ist das kleinere - vorallem lösbarere Problem: Eben ohne Microsoft.

Pfade für Dateien im Script einer Webseite, die per HTTP-Server oder lokal von Festplatte gestartet wird:

Relative Pfade per '.' sind unabhängig davon, wie die Wurzel (Root) der Webseite heißt.
Absolute Pfade sind abhängig davon, wie die Wurzel (Root) der Webseite heißt.

Wird die Webseite (z.B. www.twseite.de) von einem HTTP-Server gestartet, dann sind absolute Pfade bezüglich http://www.twseite.de möglich, wobei http://www.twseite.de die Root der Webseite darstellt.

(Achtung: Auf einem HTTP-Server des Internetproviders liegt die Webseite natürlich auch auf einer Festplatte, also dort in einem Ordner. Dieser Ordner muss also mit www.twseite.de logisch verbunden sein - wie, das teilt der Internet-Provider mit.

Tipp: Namen des Ordners auf der Festplatte des Internet-Providers kann man genauso nennen wie den des Ordners auf der lokalen Festplatte des PC. Ordnernamen mit Pfadzeichen wie '.' oder '/' sind natürlich nicht erlaubt)

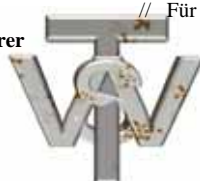
Wird die Webseite (z.B. www.twseite.de) von lokaler Festplatte aus gestartet, dann sind absolute Pfade bezüglich http://www.twseite.de nicht möglich, da es solchen Ordnernamen nicht geben kann.
Festplattenordner möglich z.B. c:\twseite\

Will man die Webseite identisch verwalten, egal ob man die Webseite von einem HTTP-Server oder von lokaler Festplatte aus startet, dann schaue man sich folgendes Beispiel für www.twseite.de an, das allerdings mit JavaScript oder JScript realisiert wird. Der Festplattenordner ist c:\twseite.

```
var BrowserAufOnlinePruefen=true;    // false für Browser nicht auf online prüfen
                                     // online: Webseite wurde auf HTTP-Server aktiviert
var DomainOhneHTTP='www.twseite.de'; // Host der Webseite ohne http:// und ohne Port
var BrowserIstOnline=false;          // Annahme: Browser ist nicht online
var PfadDerDateien="";               // Annahme: Browser ist nicht online also
                                     // alle Pfade unterhalb von c:\twseite\
                                     // wobei die Startdatei index.html der
                                     // Webseite eben in diesem Ordner liegt

if(BrowserAufOnlinePruefen)          // wenn auf online geprüft werden soll
{if(window.location.hostname!=null); // aktuell gefundener Host
  BrowserIstOnline=(window.location.hostname== DomainOhneHTTP);}
                                     // Host www.xxx.yyy prüfen auf aktuelle gefundenen Host
                                     // true so Browser online
}

if(BrowserIstOnline) {PfadDerDateien="http://" + DomainOhneHTTP;}
                                     // Pfad wenn Browser online ist: Alle Pfade unterhalb von
                                     // http://www.twseite.de
                                     // Für den HTTP-Server liegt die Webseite natürlich
```



```
// auf einer Festplatte, also dort in einem Ordner.
// Jeder Pfad in der Webseite wird in den Ordnerpfad
// der Festplatte automatisch konvertiert (HTTP zu
// Festplatte per Dienst: Daher der Name HTTP-
// Server)
```

Nachfolgend die beispielhafte Einrichtung einer Webseite per Apache-HTTP-Server 2.2.2 bis 2.2.4. unter Windows XP ab SP 1:

Webseitendomain heißt www.twseite.de mit index.html als Startdatei
 Festplattenordner der Domain, die als virtueller Host über localhost (also 127.0.0.1) laufen soll
 c:\twseite\
 Apache wurde installiert unter e:\wpx\apache\
 wobei gelten muss
 DNS-Dienst von Windows muss aktiv sein (in der Regel ist der Dienst automatisch aktiv)
 Installationstyp All Users, on Port 80, as a Service - Recommended
 Domain und Servername 127.0.0.1 (nicht localhost kodieren)
 Email beliebig@localhost
 Mit der Installation wird der Dienst-Monitor von Apache bei Windowsstart ebenfalls starten.
 Es wird Apache als permanenter Dienst eingerichtet, der über den Dienst-Monitor von Apache
 aktivierbar / deaktivierbar ist. Es sind zwar mehrere Apache-Dienste einrichtbar, aber genau 1
 kann nur immer aktiv sein.

Apache-Software einrichten:

Dienst-Monitor von Apache: siehe oben
 aber: Da in Windows für jeden Aufruf der Webseite www.twseite.de unter Apache eine
 Einstellung getroffen werden muss, empfiehlt es sich, den Apache-Dienst
 grundsätzlich manuell zu starten und zu stoppen anhand nachfolgend
 vorgestellter BAT-Dateien.
 Dafür muss aber einmalig folgendes eingestellt werden:
 Der Apache-Dienst-Name ist per Dienst-Monitor zu sehen (und der Status ob
 aktiv oder deaktiv).
 Dienste sind unter Windows per Systemsteuerung-Verwaltung-Dienste
 verwaltbar (auch per Apache-Monitor ist die Dienstverwaltung
 aktivierbar).
 Der Apache-Dienst muss zuerst deaktiviert werden, dann auf Starttyp manuell
 gesetzt werden: Mit Windows-Start startet der Dienst nicht automatisch.

Windows anpassen an den Virtuellen Host

Die Anpassung erfolgt so, dass Apache per Batch-File, die unten erklärt werden, gestartet
 und deaktiviert wird (Batch-Files sind passend zur Anpassung von Windows).

Unter c:\windows\system32\drivers\etc\ liegt die Datei hosts

Copyright (c) 1993-1999 Microsoft Corp.

```
#
# Dies ist eine HOSTS-Beispieldatei, die von Microsoft TCP/IP
# für Windows 2000 verwendet wird.
#
# Diese Datei enthält die Zuordnungen der IP-Adressen zu Hostnamen.
# Jeder Eintrag muss in einer eigenen Zeile stehen. Die IP-
# Adresse sollte in der ersten Spalte gefolgt vom zugehörigen
# Hostnamen stehen.
# Die IP-Adresse und der Hostname müssen durch mindestens ein
# Leerzeichen getrennt sein.
#
# Zusätzliche Kommentare (so wie in dieser Datei) können in
# einzelnen Zeilen oder hinter dem Computernamen eingefügt werden,
# aber müssen mit dem Zeichen '#' eingegeben werden.
#
# Zum Beispiel:
#
# 102.54.94.97 rhino.acme.com # Quellserver
# 38.25.63.10 x.acme.com # x-Clienthost
#
127.0.0.1 localhost
```

Diese Datei verwaltet -wie man sieht - auch den localhost.



Standardgemäß ist localhost auf 127.0.0.1 gelegt (nur deswegen sind localhost und 127.0.0.1 synonym)

Soll aber die Webseite www.twseite.de über 127.0.0.1 getestet werden, muss also die Zeile

127.0.0.1 localhost

ersetzt werden durch

127.0.0.1 www.twseite.de

Nach dem Test muss localhost wieder für 127.0.0.1 verfügbar gemacht werden, also der Standard gesetzt werden.

Genau dieses Ersetzen machen die Batch-Files siehe unten. Dafür benötigen sie nur 2 neue Ordner, die einmalig mit Inhalt manuell angelegt werden müssen in c:\windows\system32\drivers\etc\

Schritt 1: Ordner c:\windows\system32\drivers\etc_hosts_standard\

manuell erzeugen, und dorthin die bisher unveränderte, also originale hosts-Datei kopieren (Datei enthält 127.0.0.1 localhost)

```
# Die IP-Adresse und der Hostname müssen durch mindestens ein
# Leerzeichen getrennt sein.
#
# Zusätzliche Kommentare (so wie in dieser Datei) können in
# einzelnen Zeilen oder hinter dem Computernamen eingefügt werden,
# aber müssen mit dem Zeichen '#' eingegeben werden.
#
# Zum Beispiel:
#
#      102.54.94.97      rhino.acme.com      # Quellserver
#      38.25.63.10      x.acme.com          # x-Clienthost
#
127.0.0.1      localhost
```

Schritt 2 Ordner c:\windows\system32\drivers\etc_hosts_mit_www_twseite_de\
manuell erzeugen, und dorthin die bisher unveränderte, also originale hosts-Datei kopieren (Datei enthält 127.0.0.1 localhost)
dann diese Datei in diesem Ordner per notepad.exe (nicht Word etc.) auf
127.0.0.1 www.twseite.de

```
# Copyright (c) 1993-1999 Microsoft Corp.
#
# Dies ist eine HOSTS-Beispieldatei, die von Microsoft TCP/IP
# für Windows 2000 verwendet wird.
#
# Diese Datei enthält die Zuordnungen der IP-Adressen zu Hostnamen.
# Jeder Eintrag muss in einer eigenen Zeile stehen. Die IP-
# Adresse sollte in der ersten Spalte gefolgt vom zugehörigen
# Hostnamen stehen.
# Die IP-Adresse und der Hostname müssen durch mindestens ein
# Leerzeichen getrennt sein.
#
# Zusätzliche Kommentare (so wie in dieser Datei) können in
# einzelnen Zeilen oder hinter dem Computernamen eingefügt werden,
# aber müssen mit dem Zeichen '#' eingegeben werden.
#
# Zum Beispiel:
#
#      102.54.94.97      rhino.acme.com      # Quellserver
#      38.25.63.10      x.acme.com          # x-Clienthost
#
127.0.0.1      www.twseite.de
```

Virtual-Host genau 1x einrichten in e:\wxp\apache\conf\ dort in der Datei httpd.conf



erst httpd.conf kopieren in einen Sicherungsordner freier Wahl

dann per notepad.exe (nicht per Word etc.) am Ende der httpd.conf folgenden Text einfügen

Achtung: Es muss natürlich dann	
www.twseite.de	ersetzt werden durch zu
	testende Domain
c:/twseite	ersetzt werden durch den
	wirkliche Pfad

```
# Virtual Host für www.twseite.de auf 127.0.0.1:80 gehostet

# ERST Virtual Host für den Server, der damit alle Servernamen ungleich
www.twseite.de abfängt
# also Angaben aus Serverinstalltion verwendet
NameVirtualHost 127.0.0.1
<VirtualHost 127.0.0.1>
    ServerName localhost
    ServerAlias 127.0.0.1
    DocumentRoot e:/wxp/apache/htdocs
    ErrorLog e:/wxp/apache/logs/error.log
    TransferLog e:/wxp/apache/logs/access.log
    ScriptAlias /cgi-bin/ e:/wxp/apache/cgi-bin/
</VirtualHost>

# Virtual Host für www.twseite.de
<VirtualHost 127.0.0.1>
    ServerName www.twseite.de
    ServerAlias 127.0.0.1
    DocumentRoot c:/twseite
    DirectoryIndex index.html
    ErrorLog c:/twseite/apache_error.log
    TransferLog c:/twseite/apache_access.log
    <Directory c:/twseite>
        Options Indexes FollowSymLinks MultiViews
        AllowOverride None
        Order allow,deny
        Allow from all
    </Directory>
</VirtualHost>
```

Apache-Start und Stop per folgender BAT-Dateien, die z.B. in C:\ liegen (egal wo auf Festplatte):

Achtung: Falls der Start von Apache wegen nicht vorhandenem Dienst nicht erfolgen kann, dann gilt:

Mit der Apache-Installation wurde ein Dienst eingerichtet: siehe oben. Nur leider, der Apache-Start verlangt einen weiteren Dienst mit anderem Namen. Da bekannt ist, dass aber nur 1 Dienst zu jedem Zeitpunkt aktiv sein kann, ist es sehr verwunderlich, wieso Apache-Start den bereits vorhandenen Dienst nicht nimmt.

Anstelle von wundern bitte folgendes genau 1 mal ausführen:

```
e:\wxp\apache\bin\ httpd.exe -k install
```

im bin-Verzeichnis liegen die ausführbaren Dateien von Apache.
httpd.exe ist die Hauptkomponente von Apache.
-k install installiert einen weiteren Apache-Dienst.

Logischerweise ist dieser Dienst analog zum Dienst, der bei Apache-Installation erzeugt wurde, ebenfalls per Dienste-Verwaltung zu deaktivieren und auf Starttyp manuell zu setzen. Dieser neue Dienst hat aber den passenden Namen, der vom Start von Apache akzeptiert wird.

Hier ein Beispiel für den Dienste-Wirrwarr:

Apache-Installation erzeugt Dienst 'Apache2.2'.
Apache-Start will aber den Dienst 'Apache 2'

Batch-File 'ApacheStarten.bat' für Start des Apache, wobei der Dienst 'Apache2' aktiviert wird



```

@echo off
cls
echo Apache als Anwendung starten bei inaktivem Apache-Dienst 'Apache2'
echo          (Apache-Dienst darf nicht bei Windows-Start aktiv sein)
echo          (Apache stoppen immer per ApacheStoppen.bat)
echo hosts-Datei mit www.twseite.de bereitstellen
echo.
copy /V /Y C:\WINDOWS\system32\drivers\etc\_hosts_mit_www_twseite_de\hosts
C:\WINDOWS\system32\drivers\etc\hosts > NUL
e:\wxp\apache\bin\httpd.exe -k start
echo.
echo.
echo      ..... Apache-Anwendung wurde nur gestartet, wenn oben keine
echo              Fehlermeldung angezeigt wurde !
echo              Falls Fehlermeldung dann im Monitor Apache Servers
echo              zu allen Eintraege STOP-Button klicken (falls klickbar)
echo              und Batch-File neu starten
echo.
echo.
echo      Apache kann keine ActiveX durchreichen (z.B. Logox WebSpeech)
echo              per JScript-erzeugte Popups nicht immer korrekt
rendern
echo.
echo.
pause
echo.
echo.
echo aktuelle hosts-Datei lautet
echo.
type C:\WINDOWS\system32\drivers\etc\hosts
echo.
pause

```

Batch-File 'ApacheStoppen.bat' für Start des Apache, wobei der Dienst 'Apache2' de-aktiviert wird

```

@echo off
cls
echo Apache als Anwendung stoppen bei aktiven Apache-Dienst 'Apache2'
echo          (Apache-Dienst aktiviert per ApacheStarten.bat)
echo hosts-Datei ohne www.twseite.de bereitstellen
echo.
copy /V /Y C:\WINDOWS\system32\drivers\etc\_hosts_standard\hosts
C:\WINDOWS\system32\drivers\etc\hosts >NUL
e:\wxp\apache\bin\httpd.exe -k stop
echo.
echo.
echo      ..... Apache-Anwendung wurde nur gestoppt, wenn oben keine
echo              Fehlermeldung angezeigt wurde !
echo              Falls Fehlermeldung dann im Monitor Apache Servers
echo              dann war kein Appache-Server aktiv.
echo.
echo.
pause
echo.
echo.
echo aktuelle hosts-Datei lautet
echo.
type C:\WINDOWS\system32\drivers\etc\hosts
echo.
pause

```

Batch-File 'ApacheHostsDateiAnzeigen.bat' für Anzeige der hosts-Datei



```
@echo off
cls
echo aktuelle hosts-Datei anzeigen
echo.
type C:\WINDOWS\system32\drivers\etc\hosts
echo.
echo.
echo.
pause
```

Webseite aktivieren per Apache:

Da die Webseite nur per o.g. Batch-Dateien gestartet bzw. gestoppt werden kann, kann der Apache-Dienst-Monitor nur noch zur Kontrolle benutzt werden, ob eine Apache-Dienst aktiv ist.

Schritt 1: Vor dem Start der Webseite darf kein Apache-Dienst aktiv sein

Schritt 2 ApacheHostsDateiAnzeigen.bat aktivieren. Die hosts-Datei muss enthalten

127.0.0.1 localhost

Schritt 3: ApacheStarten.bat aktivieren. Apache startet.

Wer will kann jetzt Schritt 2 wiederholen und sieht dann

127.0.01. www.twseite.de

Wenn jetzt die Firewall-Software sich meldet, dann Apache erlauben
am Port 80 (also HTTP) von 127.0.0.1

Achtung: Nutzt die zu testende Webseite andere Ports z.B. die von
Plugins, dann können die Plugins selbst einen
Firewalleintrag verlangen, aber Apache kann natürlich
nur HTTP (Port 80) und muss nur zwischen Webseite
und den Plugins vermitteln (eben über Port 80)

Schritt 4: Browser der Wahl aktivieren
Dort die Domain www.twseite.de eintippen die Webseite öffnet sich anhand
c:\twseite\index.html

Tipp: Sollte die Webseite bereits auf einem Webserver online sein, dann würde
ohne Apache-Start natürlich die online-Variante aktiviert werden
und nicht die Webseite von der Festplatte aus dem Ordner c:\twseite !

Wichtig: Sollte die Webseite im Ordner c:\twseite\ geändert worden sein, so
muss der Browser-Cache gelöscht werden, bevor die die Webseite
erneut aktiviert wird. Apache lässt den Browser genauso agieren, als ob
der online die Webseite laden würde, also über den Browser-Cache.
Manche Browser oder Webseiten unterbinden dann das Neuladen einer
Datei, die im Browser-Cache dem Namen nach vorhanden ist, aber in
c:\twseite verändert vorliegt also zwingend neu geladen werden muss....
Ergo Browser-Cache vorher löschen.

Browser-Cache im löschen im
Internet Explorer per Internetoptionen, die
im Browser per Menüpunkt Extra oder per Systemsteuerung
aktivierbar sind (ev. Verknüpfung auf Desktop manuell
erzeugen)
Opera per Menüpunkt Extra-Einstellungen-Erweitert

Schritt 5: Webseite getestet (man hat jetzt ein Smily-Gesicht oder graue Haare mehr, oder den Chef,
Kunden, Frau am Hals *g)
also Zeit zum Deaktivieren von Apache per ApacheStoppen.bat

Schritt 6: Webseitendaten aus c:\twseite sichern.

Errormeldungen des HTTP-Servers per .htaccess-Datei verwalten:

HTTP-Server-Fehlercode auf die der Webmaster reagieren kann per .htaccess

xxx	Meldung	Bedeutung
204	No Content	Das angeforderte Dokument enthält keine Daten
301	Moved Permanently	Das angeforderte Dokument wurde verschoben
302	Moved Temporarily	Das angeforderte Dokument wurde zeitweise verschoben
400	Bad Request	Zugriff des Client auf Server ist unbekannt



401	Unauthorized	Zugriff des Client auf Server nicht erlaubt
404	Not Found	Das angeforderte Dokument ist nicht vorhanden
408	Request Time Out	Antwort des Servers hat Zeitüberschreitung
500	Internal Server Error	Server hat internen Fehler
501	Not Implemented	Zugriff des Client auf Server nicht ausführbar
503	Service Unavailable	Server ist nicht zu erreichen

Für jeden der obigen Fehler kann z.B. in die Root der Webseite ein HTML-Dokument hinterlegt werden, dass bei auftreten des Fehler aktiviert wird, den Fehler beschreibt und /oder Ersatzreaktion durchführt.

in .htaccess muss Zeile hinterlegt werden

```
ErrorDocument xxx /yyy.html
```

xxx siehe oben

yyy freier Dateiname in der Root, Achtung: Datei MUSS vorhanden sein !!!

Pfade auch möglich bezüglich Root /

Es ist auch möglich

```
ErrorDocument xxx "freier Text"
```

1. Browser

1.1. Browser allgemein

1.1.1. Browserdifferenzen bei pixelgenauer Darstellung vermeiden

```
<BODY MARGINWIDTH=0
MARGINHEIGHT=0
LEFTMARGIN=0
TOPMARGIN=0
```

```
>
```

1.1.2. Kontextmenü für rechte Maus sperren

unsinnig da Browser-Menü die gesperrten Aktionen zulässt

1.2. Microsoft Internet Explorer

1.2.1. Autoausfüllen

ein- und ausschaltbar

besteht aus folgenden wichtigsten Elementen:

vcard.# die die Daten aufnehmen,

die z.T. von anderen Programmen z.B. Email verarbeitet werden können

#	Art
Cellular	Handynummer
Company	Firma
Department	Abteilung
Email	Emailadresse
FirstName	Vorname
Gender	Geschlecht
Home.City	Stadt privat
Home.Country	Land privat
Home.Fax	Faxnummer privat
Home.Phone	Telefon privat
Home.StreetAddress	Straße privat
Homepage	Homepage privat
JobTitle	Beruf
LastName	Nachname
Notes	Notizen
Office	Büro
Business.City	Stadt Firma
Business.Country	Land Firma
Business.Fax	Fax Firma
Business.Phone	Telefon Firma
Business.StreetAddress	Strasse Firma
Business.URL	Homepage der Firma

Die Daten gelangen per automatischer Vervollständigung im IE auf die Festplatte und können in HTML ausgelesen werden:

```
Bsp.: <INPUT TYPE=TEXT
NAME="email"
VCARDNAME="vcard.email"
>
```



Das Aus- und Einlesen im IE ist von außen steuerbar per

```
<INPUT ..... VCARDNAME .....AUTOCOMPLETE="off"> // für abschalten
```

1.2.2. Formular mit Pulldown-Auswahlfeld mit farbigen Einträgen

```
<FORM NAME="Test">
  <SELECT
    SIZE=1
    NAME="Farbe"
    onChange=.....
  >
    <OPTION STYLE="color:#FF0000;">
      Rot
    <OPTION STYLE="color:#00FF00;">
      Gruen
    <OPTION STYLE="color:#0000FF;">
      Blau
  </SELECT>
</FORM>
```

1.2.3. Linklayout verändern

1.2.3.1. Linkunterstreichung abschalten

```
<STYLE TYPE="text/css">
a:link {text-decoration:none;}
a:visited{text-decoration:none;}
a:active{text-decoration:none;}
</STYLE>
```

1.2.3.2. Standardlinkfarben ändern

Standardlinkfarben sind:

noch nie angeklickt:	blau
schon mal angeklickt	violett
gerade angewählt	rot

Abänderung wenn möglich vermeiden, da User sich an Standards gewöhnen

Bsp:

```
<HEAD>
  <STYLE>
  <!--
    a {color:navy}           // blau als Standard
    a:visited {color:orange} // violett als Standard auf orange
    a:active {color:purple}  // rot als Standard auf purpur
  -->
</STYLE>
```

1.2.3.3. Link-Layout verändern wenn Maus über Link fährt

Bsp:

ohne Maus:	grau
mit Maus:	rot unterstrichen

```
A:visited {color:gray;text-decoration:none;}
A:hover {color:red;text-decoration:underline;}
```

Achtung: hover MUSS nach visited kodiert werden

1.2.3.4. Typen von Links-Layouts per CSS definieren

```
<HEAD>
<STYLE TYPE="text/css">
<!--
  a.link_typ1:link{color:#FF0000}
  a.link_typ1.visited{color:#00FF00}
  a.link_typ1:hover{color:#0000FF} // Mouse over

  a.link_typ2 ....
  .....

//-->
</STYLE>
</HEAD>
<BODY>
.....
  <A CLASS="link_typ1" HREF="....">.....>/A> // Zuweisen des Linklayout über CLASS
</BODY>
```

1.2.3.5. Cursor für einen Link ändern

ab IE 4.x

```
<HEAD>
<STYLE TYPE="text/css">
<!--
```



```

oder      a.link:{cursor: help;}      // für Fragezeichen
          a.link:{cursor:url(name.gif)} // für eigenen Cursor

//-->
</STYLE>
</HEAD>

<BODY>
.....
      <A CLASS="link"  HREF="....">.....>/A>      // Zuweisen des Linklayout über CLASS

```

2. Button

2.1. Buttonbeschriftung mit mehrzeiligem Text

nur beim IE möglich

```
<BUTTON>Klick<BR>mich !</BUTTON>
```

falls der Text länger ist als die Buttondimensionen, so wird er abgeschnitten

3. Formular-TEXTAREA ohne Rollbalken (Scrollbalken)

Rollbalken sind nur im IE unterdrückbar (von anderen Browsern ignoriert):

```
<TEXTAREA STYLE="overflow:hidden">
```

4. Grafik

4.1. Viereck-Grafik

4.1.1. Viereck-Grafik zerlegen in kleine Elemente und als blinde Tabelle zusammenbauen

kleine Grafiken sind schneller ladbar
eine Tabelle wird erst angezeigt, wenn sie komplett geladen ist

Beispiel: Viereckgrafik aus 4 Teilen

```

bild1.gif      bild2.gif
bild3.gif      bild4.gif

```

```

<TABLE BORDER="0" CELLPADDING="0" CELLSPACING="0" WIDTH="310">
<TR>
<TD><IMG SRC="bild1.gif" WIDTH="154" HEIGHT="137" BORDER="0"></TD>
<TD><IMG SRC="bild2.gif" WIDTH="156" HEIGHT="137" BORDER="0"></TD>
</TR>
<TR>
<TD><IMG SRC="bild3.gif" WIDTH="154" HEIGHT="137" BORDER="0"></TD>
<TD><IMG SRC="bild4.gif" WIDTH="156" HEIGHT="137" BORDER="0"></TD>
</TR>
</TABLE>

```

Hinweis zu WIDTH der Grafiken: $154 + 156 = 310 = \text{Tabellenbreite}$

4.1.2. Viereck-Grafik zerlegen in kleine Elemente, als blinde Tabelle zusammenbauen und jedem Teil einen eigenen Link zuweisen (verweis-sensitive Grafik)

kleine Grafiken sind schneller ladbar
eine Tabelle wird erst angezeigt, wenn sie komplett geladen ist

Beispiel: Viereckgrafik aus 4 Teilen

```

bild1.gif      bild2.gif
bild3.gif      bild4.gif

```

```

<TABLE BORDER="0" CELLPADDING="0" CELLSPACING="0" WIDTH="310">
<TR>
<TD>
<A HREF="bild1.htm">
<IMG SRC="bild1.gif"
      WIDTH="154"      HEIGHT="137"
      BORDER="0"      ALT="link bild1.gif">
</A>
</TD>

```



```

<TD>
<A HREF="bild2.htm">
<IMG SRC="bild2.gif"          WIDTH="156"    HEIGHT="137"
                                BORDER="0"      ALT="link bild2.gif">
</A>
</TD>
</TR>
<TR>
<TD>
<A HREF="bild3.htm">
<IMG SRC="bild3.gif"          WIDTH="154"    HEIGHT="137"
                                BORDER="0"      ALT="link bild3.gif">
</TD>
</A>
<TD>
<A HREF="bild4.htm">
<IMG SRC="bild4.gif"          WIDTH="156"    HEIGHT="137"
                                BORDER="0"      ALT="link bild4.gif">
</A>
</TD>
</TR>
</TABLE>

```

Hinweis zu WIDTH der Grafiken: 154 + 156 = 310 = Tabellenbreite

4.2. **Bilderrahmen aus verschachtelten Tabellen**

Beispiel: Bilderrahmen mit drei Erhebungen und 2 Vertiefungen (wie vom Tischler ausgefräst)

```

<TABLE BORDER="1" CELLPADDING="2" WIDTH="190" BGCOLOR="navy" CELLSPACING="0">
<TR>
<TD WIDTH="100%">
<TABLE BORDER="1" CELLPADDING="2" WIDTH="185" BGCOLOR="blue" CELLSPACING="0">
<TR>
<TD WIDTH="100%">
<TABLE BORDER="1" CELLPADDING="2" WIDTH="180" BGCOLOR="black" CELLSPACING="0">
<TR>
<TD WIDTH="100%"><IMG BORDER="0" SRC="bild.gif" WIDTH="175" HEIGHT="276"></TD>
</TR>
</TABLE>
</TD>
</TR>
</TABLE>
</TD>
</TR>
</TABLE>
</TD>
</TR>
</TABLE>

```

4.3. **Schnelleres Laden auch bei abgeschalteter Grafikanzeige**

Angabe von Grafik mit Bilddimension verhindert permanente Neuberechnung des Seitenlayouts während des Ladens, da anhand der Grafikdimension gleich ein korrekter Platzhalter einmalig ermittelt wird.

```

<IMG SRC="...../bild.gif"
      WIDTH="120"
      HEIGHT="400">

```

4.4. **Filter für HTML-Elemente ab IE 5.5**

Filter für HTML-Elemente werden in Verbindung mit Style-Sheet eingesetzt.

Es können beim IMG-Tag alle Bildformate, die der Browser kennt, verarbeitet werden, also z.B. GIF und JPG.

4.4.1. **Prinzip der Kodierung von Filtern**

Variante 1

Im HTML-Element wird das CLASS-Attribut verwendet. Innerhalb <HEAD> ..</HEAD> wird mit dem <STYLE>-Tag die CSS-Klasse mit den Alpha-Filtern kodiert.

```

<HTML>
<HEAD>
<STYLE>
tag_name_1.filter_name_1 { filter: .....; }
.....
tag_name_1.filter_name_n { filter: .....; }
....
tag_name_n.filter_name1 { filter: .....; }
.....
tag_name_n.filter_namen { filter: .....; }
</STYLE>
</HEAD>

```



```

<BODY>
  <tag_name_1 ... CLASS="filter_name_1"> .... </tag_name_1>
  ....
<BODY>
</HTML>

```

Es können pro Tag mehrere Filter kodiert werden.

Jedes HTML-Element laut Tag kann aber nur einen Filter besitzen. Es können also mehrere HTML-Elemente vom gemeinsamen Tag aber mit verschiedenen Filtern kodiert werden.

Der IE bietet aber die Layer-Technik z.B. anhand des DIV-Tags, so dass Überlagerungen an einer gemeinsamen Position möglich sind.

Variante 2

Direkt im HTML-Element wird das STYLE-Attribut verwendet. Damit entfällt die Klassendeklaration per <STYLE> innerhalb <HEAD>. Es wird aber ein internes Filterprogramm benötigt z.B. für Fade. Typisches HTML-Element ist der <DIV>-Tag, der auch die Layer-Technik unterstützt.

```

<HTML>
<HEAD>
</HEAD>
<BODY>
  <DIV ID="DIV_ID"
    STYLE=".....;filter:progid:DXImageTransform.Microsoft.Fade(duration=y);">
    ....
  </DIV>
</BODY>
</HTML>

```

y ist numerischer Wert der Dauer

4.4.2. Alpha-Filter für HTML-Grafik-Elemente ab IE 5.5

Grafiken können transparent dargestellt werden: Die Deckungsstärke ist einstellbar.

4.4.2.1. Beispiel für transparentes INPUT-Element

Das INPUT-Element kann den Hintergrund des Dokumentes in der INPUT-Dimension stufig abdecken.

```

<HTML>
<HEAD>
<STYLE>
  INPUT.filter_name {filter:Alpha(opacity=y);}
</STYLE>
</HEAD>
<BODY>
  <INPUT TYPE="button" .... CLASS="filter_name">
</BODY>
</HTML>

```

opacity:	Deckkraft
	y von 0 bis 100
	0 keine Deckkraft
	100 alles abdeckend

4.4.2.2. Beispiel für transparentes Vordergrund-Bild auf dem Hintergrundbild des Dokumentes

Ziel ist es, eine per Filter gesteuerte Deckkraft zu erreichen. Beispiel dafür: Ein Bild im Vordergrund soll sich in den Hintergrund aus einem Bild einprägen, also mit dem Hintergrund per Deckkraft verschmelzen. Auch andersrum wäre es möglich: Der Hintergrund erscheint optisch vor dem Vordergrundbild.

```

<HTML>
<HEAD>
<STYLE>
  IMG.filter_name {filter:Alpha(opacity=y);}
</STYLE>
</HEAD>
<BODY BACKGROUND="hintergrund_bild.gif">
  <IMG SRC="bild.gif" .... CLASS="filter_name">
</BODY>
</HTML>

```

opacity:	Deckkraft
	y von 0 bis 100
	0 keine Deckkraft
	100 alles abdeckend



4.4.2.3. *Beispiel für transparente Überlagerung zweier Bilder*

Ziel ist es, eine per Filter gesteuerte Deckkraft zu erreichen. Beispiel dafür: Ein Bild im Vordergrund soll mit einem zweiten in der Dimension kleineren Bild überlagert werden, so dass dieses kleinere mit dem größeren per Deckkraft verschmilzt. Auch andersrum wäre es möglich: Das größere Bild erscheint optisch vor dem kleineren.

```
<HTML>
<HEAD>
<STYLE>
    IMG.filter_name {filter:Alpha(opacity=y);}
</STYLE>
</HEAD>
<BODY>
    // je nach Variante die Reihenfolge der IMG verändern.
    <IMG SRC="hintergrund_bild.gif" ....>
    <BR>
    <IMG SRC="bild.gif" .... CLASS="filter_name">
</BODY>
</HTML>
```

opacity: Deckkraft
 y von 0 bis 100
 0 keine Deckkraft
 100 alles abdeckend

4.4.2.4. *Beispiele für Aus- und Einblende eines Bildes*

```
<HTML>
<HEAD>
    <SCRIPT LANGUAGE="JavaScript">
    <!--
        var Sichtbar=true;    // DIV ist natürlich nach dem Dokument-Laden sichtbar
        var StandardFarbe="green";

        function Blenden()
        {
            // Fade zurücksetzen
            DIV_ID.filters[0].Apply();

            if (Sichtbar)
            {
                Sichtbar=false;
                DIV_ID.style.visibility="hidden";
            }
            else
            {
                Sichtbar=true;
                DIV_ID.style.visibility="visible";
                DIV_ID.style.backgroundColor=StandardFarbe;
            }

            // Fade starten
            DIV_ID.filters[0].Play();
        }
    // -->
    </SCRIPT>
</HEAD>
<BODY>
    <BUTTON onclick="Blenden()">
        Aus- und Einblenden
    </BUTTON>
    <BR>
    <BR>
    <DIV     ID="DIV_ID"
        STYLE="height:250px;width:250px;background-color:red;
            filter:progid:DXImageTransform.Microsoft.Fade(duration=1);"
            // alles EINE Zeile !!
    >
        <IMG SRC="Bild.gif" ....HEIGHT=250 WIDTH=250>
            // Bild-Dimension muss in die DIV-Dimension reinpassen !
    </DIV>
</BODY>
</HTML>
```

4.4.2.5. *Beispiel für Aus- und Einblende von Bildern mit Bildwechsel*

Variante 1



Es werden zwei DIV an absoluter Position überlagert. Jedes DIV hat einen eigenen Inhalt, hier im Beispiel sein eigenes Bild. Durch das Ein- und Ausblenden wird ein Slide-Show-Effekt der beiden Bilder erreicht.

Nachteil: Pro Bild muss ein eigenes DIV erzeugt werden.

Es ist möglich, nur mit einem DIV zu arbeiten, da der IE den Ersatz den inneren DIV-Bereiches zwischen <DIV> und </DIV> zulässt. Pro Ersatz wird ein Bild eingebunden.

DIV_ID.innerHTML=

Da es sich nur um einen einzigen <DIV> handelt, ist dieser per Layertechnik des IE auch bewegbar.

```
<HTML>
<HEAD>
  <SCRIPT LANGUAGE="JavaScript">
    <!--
      var DIV_ID1_Sichtbar=true;    // DIV_ID1 ist nach dem Dokument-Laden sichtbar
                                   // DIV_ID2 aber nicht

      function BildWechsel()
      {
        // Fade zurücksetzen
        DIV_ID1.filters[0].Apply;
        DIV_ID2.filters[0].Apply;

        if (DIV_ID1_Sichtbar)
        {
          DIV_ID1_Sichtbar = false;
          DIV_ID1.style.visibility="hidden";
          DIV_ID2.style.visibility="visible";
        }
        else
        {
          DIV_ID1_Sichtbar = true;
          DIV_ID1.style.visibility="visible";
          DIV_ID2.style.visibility="hidden";
        }
        // Fade starten
        DIV_ID1.filters[0].Play();
        DIV_ID2.filters[0].Play();
      }
    // -->
  </SCRIPT>
</HEAD>
<BODY>
  <BUTTON onclick="BildWechsel()">
    Bild wechseln mit Aus- und Einblenden
  </BUTTON>
  <BR>
  <BR>
  <DIV ID="DIV_ID1"
    STYLE="height:250px;position:absolute;top:50;width:250px;
      background-color:red; visibility:visible
      filter:progid:DXImageTransform.Microsoft.Fade(duration=1);"
    // alles EINE Zeile
  >
    <IMG SRC="DIV_ID1_Bild.gif" ....HEIGHT=250 WIDTH=250>
    // Bild-Dimension muss in die DIV-Dimension reinpassen !
  </DIV>

  <DIV ID="DIV_ID2"
    STYLE="height:250px;position:absolute;top:50;width:250px;
      background-color:red; visibility:hidden
      filter:progid:DXImageTransform.Microsoft.Fade(duration=1);"
    // alles EINE Zeile
  >
    <IMG SRC="DIV_ID2_Bild.gif" ....HEIGHT=250 WIDTH=250>
    // Bild-Dimension muss in die DIV-Dimension reinpassen !
  </DIV>
</BODY>
</HTML>
```

Variante 2

Es wird der innere DIV-Bereich zwischen <DIV> und </DIV> ersetzt und damit je ein Bild dargestellt. Durch das Ein- und Ausblenden wird ein Slide-Show-Effekt der Bilder erreicht.

Da es sich nur um einen einzigen <DIV> handelt, ist dieser per Layertechnik des IE auch bewegbar.

```
<HTML>
<HEAD>
```



```

<SCRIPT LANGUAGE="JavaScript">
<!--
    var BildUrlFeld=Array
    (
        // hier beliebig viele Bilder eintragen
        "test1.jpg",
        "test2.jpg",
        "test3.jpg"
    );
    var BildUrlFeld_Laenge= BildUrlFeld.length;

    //      fuer alle Bilder die identischen Dimensionen !!!
    var BildHoehe=444;
    var BildBreite=640;

    // Zeigerfeld zum Vorladen der Grafiken
    var BildZeigerFeld = Array();

    var Index=1;          // Starten mit test2.jpg, da test1.jpg bereits angezeigt mit Laden des DIV

    function BildWechsel()
    {
        // Fade zurücksetzen
        DIV_ID.filters[0].Apply();

        // Bild im DIV anzeigen
        document.AktuellesBild.src= BildZeigerFeld[Index].src;

        // Fade starten
        DIV_ID.filters[0].Play();

        // nächstes Bild einstellen
        Index++;

        // Index korrigieren
        if (Index >= BildUrlFeld_Laenge)
        {Index=0;}
    }
    // nachfolgender Code wird mit dem Laden des HEAD-Teiles abgearbeitet

    // Bildobjekte vorladen: allokieren und Zeiger merken per Prototyping
    for (i=0; i<= BildUrlFeld_Laenge; i++)
    {
        // Image-Zeiger bilden als Feldeintrag
        BildZeigerFeld[i] = new Image();

        // Url dem Zeiger zuweisen und Bild somit vorladen
        BildZeigerFeld[i].src= BildUrlFeld[i];
    }

    // DIV erzeugen mit test1.gif als Startbild
    //      in Dimension aller Bilder
    document.write(
        '<DIV '
        + 'ID="DIV_ID" '
        + 'STYLE="height:' + BildHoehe + 'px;width:' + BildBreite + 'px;'
        + 'filter:progid:DXImageTransform.Microsoft.Fade(duration=1);'
        + '""'
        + '>\n'
    );

    document.write(
        '<IMG NAME="AktuellesBild"'
        + 'SRC="" + BildZeigerFeld[0].src + "'
        + 'WIDTH=' + BildBreite + '" '
        + 'HEIGHT=' + BildHoehe
        + '>\n'
    );

    document.write(
        '</DIV>\n');

    // Button zum Bildwechsel anzeigen
    document.write(
        '<BUTTON onclick="BildWechsel()">'
        + 'Nächstes Bild'
        + '</BUTTON>\n'
    );

// -->
</SCRIPT>
</HEAD>

```



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

4.4.3. Shadow-Filter für HTML-Grafik-Elemente ab IE 5.5

4.4.3.1. Beispiel für Schatten eines Bildes

```
<HTML>
<HEAD>
  <STYLE>
    DIV.filter_name {filter:shadow(color=#000000,direction=120); width:100; color: #FF0000;}
  </STYLE>
</HEAD>
<BODY>
  <DIV CLASS="filter_name" STYLE="width:400px;">
    <IMG SRC="bild.gif">
    Das Bild und dieser Text haben einen Schatten
    <BR>
    Die Bild-Dimension muss mit dem des DIV uebereinstimmen !
  </DIV>
</BODY>
</HTML>
```

direction: Schattenrichtung in Grad; von 0 bis 360 Grad

4.4.3.2. Beispiel für 3D-Effekt eines Bildes

Dieser Effekt dient dem verschobenen Überlagern ein und desselben Bilder, so dass ein 3D-Effekt mit Schatten erzeugt wird. Die Verschiebung ist damit die Schattendicke bzw. die Effekt-Stärke des 3D-Effektes. Besonders günstig sieht man diesen Effekt, wenn die Grafik einen Schriftzug enthält.

Die Überlagerung erfolgt anhand zweier DIV's, die je eine CSS-Klasse erhalten und ansonsten identisch sind.

```
<HTML>
<HEAD>
  <STYLE>
    DIV.DIV1_filter:{ filter:shadow(color=#FFFF00,direction=300, strength=5);
                      width:100; color: #FF0000;}
                      // alles EINE Zeile
    DIV.DIV2_filter:{ filter:shadow(color=#000000,direction=120, strength=8);
                      width:100; color: #FF0000;}
                      // alles EINE Zeile
  </STYLE>
</HEAD>
<BODY>
  <DIV CLASS="DIV1_filter " STYLE="width:400px; position:absolute; top:50px; left:50px;">
    <BLOCKQUOTE>
      <IMG SRC="bild.gif">
      Das Bild und dieser Text haben einen Schatten
      <BR>
      Die Bild-Dimension muss mit dem des DIV uebereinstimmen !
    </BLOCKQUOTE>
  </DIV>
  <DIV CLASS="DIV2_filter " STYLE="width:400px; position:absolute; top:50px; left:50px;">
    <BLOCKQUOTE>
      <IMG SRC="bild.gif">
      Das Bild und dieser Text haben einen Schatten
      <BR>
      Die Bild-Dimension muss mit dem des DIV uebereinstimmen !
    </BLOCKQUOTE>
  </DIV>
</BODY>
</HTML>
```

direction: Schattenrichtung in Grad; von 0 bis 360 Grad

strength: Verschiebung in Pixel, also Schattenbreite und somit 3D-Effekt-Stärke

4.5. Grafik zoomen ab IE 5.5

Natürlich kann die Bildauflösung nicht geändert werden. Je höher der Zoom, um so pixeliger das Bild, welches von sich aus schon hochauflösend sei sollte.

```
<HTML>
<HEAD>
<SCRIPT LANGUAGE="JavaScript">
<!--
  var BildUrl="zoom.jpg";
  var BildBreite=250;
  var BildHoehe=372;
```



```

// Bildobjekte vorladen
// Image-Zeiger
BildZeiger = new Image();

// Url dem Zeiger zuweisen und Bild somit vorladen
BildZeiger.src = BildUrl;

// .style.zoom
//
//          ändert leider die DIV-Dimensionen, so dass die Input-Button VOR dem
//          zu zommenden Bild liegen müssen und somit ein DIV nicht nötig wird
//          kann nicht mit einem Zeichenkettenwert belegt werden, sondern muss direkt kodiert
//          werden !
document.write(
'<FORM NAME="Formular">'
+ '<INPUT NAME="Button20" TYPE=button VALUE="20%" onclick="Bild_ID.style.zoom=\'20%\''>'
+ '<INPUT NAME="Button40" TYPE=button VALUE="40%" onclick="Bild_ID.style.zoom=\'40%\''>'
+ '<INPUT NAME="Button60" TYPE=button VALUE="60%" onclick="Bild_ID.style.zoom=\'60%\''>'
+ '<INPUT NAME="Button80" TYPE=button VALUE="80%" onclick="Bild_ID.style.zoom=\'80%\''>'
+ '<INPUT NAME="Button100" TYPE=button VALUE="100%" onclick="Bild_ID.style.zoom=\'100%\''>'
+ '<INPUT NAME="Button120" TYPE=button VALUE="120%" onclick="Bild_ID.style.zoom=\'120%\''>'
+ '<INPUT NAME="Button140" TYPE=button VALUE="140%" onclick="Bild_ID.style.zoom=\'140%\''>'
+ '<INPUT NAME="Button160" TYPE=button VALUE="160%" onclick="Bild_ID.style.zoom=\'160%\''>'
+ '<INPUT NAME="Button180" TYPE=button VALUE="180%" onclick="Bild_ID.style.zoom=\'180%\''>'
+ '<INPUT NAME="Button200" TYPE=button VALUE="200%" onclick="Bild_ID.style.zoom=\'200%\''>'
+ '<INPUT NAME="Button300" TYPE=button VALUE="300%" onclick="Bild_ID.style.zoom=\'300%\''>'
+ '<INPUT NAME="Button400" TYPE=button VALUE="400%" onclick="Bild_ID.style.zoom=\'400%\''>'
+ '</FROM>\n'
+ '<BR>\n'
);
document.write(
+ '<IMG ID="Bild_ID" SRC="' + BildZeiger.src + "' '
+ 'HEIGHT=' + BildHoehe
+ 'WIDTH=' + BildBreite
+ '>\n');

document.all.Bild_ID.style.zoom='100%'; // .style ist eine document.all-Eigenschaft
// Bild_ID ist Objekt von document
// entspricht document.Formular.Button100.click();

document.Formular.Button100.focus(); // entspricht document.Formular.elements[4].focus();
// funktioniert leider nur, wenn Dokument nicht aus IE-Cache
// gelesen wird (sonst Übernahme der letzten
// Fokusposition)

// -->
</SCRIPT>
</HEAD>
<BODY>
</BODY>
</HTML>

```

5. Tabellen

5.1. Tabellen allgemein

5.1.1. Laden einer großen Tabelle

erst wenn die gesamte Tabelle geladen wurde, kann sie angezeigt werden

5.1.2. Zerlegen einer großen Tabelle in kleinere Tabellen

wegen schnellerem Laden

Angaben relativ zur Browseroberfläche oder in absoluten Pixelmaßen

bei IE: <TABLE STYLE="table-layout:fixed"> </TABLE>

5.1.3. Keine Abstände zwischen Tabellen

bei IE UND Netscape:

```

<BODY MARGINHEIGHT=0
      MARGINWIDTH=0
      TOPMARGIN=0
      LEFTMARGIN=0
>

```

5.1.4. Verschachtelte Tabellen

je höher die Verschachtelung um so Länger dauern das Laden und das anschliessende Darstellen.

5.1.5. Blinde Tabelle (ohne Rahmen) mit zellenweisem Hintergrund

ermöglicht z.B. ein Schachbrettmuster ohne Rahmen mit BGCOLOR in der Zellendefinition

Soll eine Zelle leer bleiben, also nur den Hintergrund besitzen, so muss trotzdem ein

Zellinhalt definiert werden, damit der Hintergrund angezeigt wird:

entweder
 oder



5.2. Zelle

5.2.1. Zelle mit exakten Dimensionen bei älteren Browsern

eine 1x1 Pixel große Grafik per einbinden und innerhalb von durch die Attribute WIDTH und HEIGHT auf exakte Größe skalieren
Grafik muss Hintergrundfarbe der Tabelle haben

5.2.2. Leere Zelle mit Hintergrund

Soll eine Zelle leer bleiben, also nur den Hintergrund besitzen, so muss trotzdem ein Zellinhalt definiert werden, damit der Hintergrund angezeigt wird:
entweder
 oder

5.2.3. Zelle mit Schattengrafik passend zum Hintergrund

Soll eine Zelle leer bleiben, also nur den Hintergrund besitzen, so muss trotzdem ein Zellinhalt definiert werden, damit der Hintergrund angezeigt wird:
entweder
 oder

Mit BGCOLOR wird die Hintergrundfarbe zellenweise gesetzt.

Je nach Lage des Schattens, z.B. am unteren Zellenrahmen, muss die benachbarte Zelle als Hintergrundbild die Schattengrafik erhalten, also die unterhalb liegende.
Schattengrafik und Zellbreite müssen korrespondieren.

```
<TABLE BORDER="0" WIDTH="300" CELSPACING="0" CELLPADDING="0">
<TR>
<TD WIDTH="50" BGCOLOR="brown">&nbsp;</TD>
<TD WIDTH="250" BGCOLOR="brown">
<B><FONT COLOR="white">Tabelle mit Schattenzelle</FONT></B>
</TD>
</TR>
<TR>
<TD WIDTH="50">&nbsp;</TD>
<TD WIDTH="250" BGCOLOR="silver">
hier der Inhalt der Zelle, die unterhalb einen Schatten erhalten soll
</TD>
</TR>
<TR>
<TD WIDTH="50">&nbsp;</TD>
<TD WIDTH="250" VALIGN="top">
<IMG BORDER="0" SRC="schatten.gif" WIDTH="250" HEIGHT="5">
</TD>
</TR>
</TABLE>
```

5.2.4. Ausgabe von Texten innerhalb einer Tabellenzelle

Textlage in der Tabellenzelle: oben, unten UND mittig

HTML unterstützt standardgemäß nicht die mittige Ausgabe

Lösung: Die betroffene Zelle muss als eigene Tabelle dargestellt werden aus

1 Spalte
und
3 Zeilen

```
.....
// hier der Code für die betroffene und als Tabelle darzustellende Zelle
<TD>
<TABLE BORDER=1 WITH=100 ....>

// hier die oberste Zelle
<TR> <TD HEIGHT=5% > text_oben </TD> </TR>

// hier die mittlere Zelle
<TR> <TD WIDTH=100% HEIGHT=90% > text_mitte </TD> </TR>

// hier die untere Zelle
<TR> <TD HEIGHT=5% > text_unten </TD> </TR>

</TABLE>
</TD>
```

Hinweis: Die Gesamthöhe (HEIGHT) aller 3 Zellen muss 100 % betragen.
Alle 3 Zellen sind gleich breit, also muss nur in einer Zelle
WIDTH kodiert werden
je höher die Verschachtelung um so Länger dauern das Laden und Darstellen



5.3. Beispiel: Kalender

```

<HTML>
<HEAD>
<SCRIPT LANGUAGE="JScript">
<!--
// Kalender aktueller Monat
// nur für IE

// ##### änderbare Variablen #####

// +++++ Position des Kalenders +++++
var Kalender_Top=20;
var Kalender_Left=20;

// +++++ Farben und Schrift-Layout +++++

// Schriften: Es sollten nur Fonts verwendet werden, die auf jedem Windows-PC installiert sind

// ----- Kalenderkopf -----

var Kalender_Kopf_SchriftArt='Times New Roman, Arial';
var Kalender_Kopf_SchriftGroesse= 5;
var Kalender_Kopf_SchriftFarbe='#CCEEFF';
var Kalender_Kopf_HintergrundFarbe='#3A6EA5'; // identisch mit BGCOLOR des BODY
var Kalender_SpaltenKopf_SchriftFarbe='#AAFFFF';

// ----- Tag -----

var Tag_SchriftArt='Times New Roman, Verdana,Arial';
var Tag_SchriftGroesse=4;
var Tag_SchriftFarbe='#CCCCCC';
var Tag_HintergrundFarbe=Kalender_Kopf_HintergrundFarbe;
var Tag_Sonntag_SchriftFarbe='#FFDD00';
var Tag_Heute_SchriftFarbe='#FFFFFF';
var Tag_Heute_HintergrundFarbe='#1A4E85';
var Tag_Heute_Sonntag_SchriftFarbe=Tag_Heute_SchriftFarbe;
var Tag_Heute_Sonntag_HintergrundFarbe=Tag_Heute_HintergrundFarbe;

// +++++ Timer +++++

// Timer für Kalender zur Erkennung von Tages-, Monats- und Jahreswechsel verwenden
var Kalender_Timer_Verwenden=false; // true für verwenden, false für nicht verwenden

// Timer in Millisekunden für Aktualisierung des Kalenders
// nur verwendet wenn Kalender_Timer_Verwenden auf true
// Wert so hoch wie möglich setzen damit durch den Timer wenig Ressourcen genutzt werden
var Kalender_TimerWert=5000;

// +++++ Monatsnamen +++++

var MonatsNamenFeld = new Array
(
'Januar',
'Februar',
'M&auml;r',
'April',
'Mai',
'Juni',
'Juli',
'August',
'September',
'Oktober',
'November',
'Dezember'
);

// +++++ Tagnamen kurz +++++

var TagKurzNamenFeld = new Array // Woche beginnt mit Montag
(
'Mo',
'Di',
'Mi',
'Do',
'Fr',

```



```

'Sa',
'So'
);

// ##### interne Variablen #####
// alle Variablen möglichst Wert-Typ-gerecht initialisieren (bei Zeiger kein "nil" oder
// "null" verwenden, also Zeiger ohne init).

var TagKurzNamenFeld_Laenge=TagKurzNamenFeld.length;

// Zeit- und Datum-Berechnungen
var Zeit_Jetzt;
var Zeit_Jetzt_Kette="";
var Zeit_Jetzt_Tag_Numerisch=0;
var Zeit_Jetzt_Monat_Numerisch=0;
var Zeit_Jetzt_Jahr_Numerisch=0;
var Zeit_Jetzt_Stunden_Numerisch=0;
var Zeit_Jetzt_Minuten_Numerisch=0;
var Zeit_Jetzt_Sekunden_Numerisch=0;
var Zeit_Vormonat;
var Zeit_VormonatErsterTag;

// Kalender-Anzeige: Globale Variablen, da Anzeige per Timer gesteuert werden kann
// und somit nicht pro Aufruf die Variablen neu erzeugt werden müssen.
var Anzeige_Kalender_TagesZahler=0;
var Anzeige_Kalender_HTMLCode="";
var Anzeige_Kalender_Kette="";
var Anzeige_Kalender_Zahler_TR=0;
var Anzeige_Kalender_Zahler_TD=0;
var Anzeige_Kalender_Tag_Kette1="";
var Anzeige_Kalender_Tag_Kette2="";
var Anzeige_Kalender_Tag_Kette3="";
var Anzeige_Kalender_Sonntag_Kette1="";
var Anzeige_Kalender_Sonntag_Kette2="";
var Anzeige_Kalender_Sonntag_Kette3="";

// ##### Funktionen der Zeit- und Datum-Berechnungen #####

function Zeit_AktuelleAngabenHolen()
{
    Zeit_Jetzt          = new Date();
    Zeit_Jetzt_Tag_Numerisch = Zeit_Jetzt.getDate();
    Zeit_Jetzt_Monat_Numerisch = Zeit_Jetzt.getMonth() + 1;
    Zeit_Jetzt_Jahr_Numerisch = Zeit_Jetzt.getFullYear();
    Zeit_Jetzt_Stunden_Numerisch = Zeit_Jetzt.getHours();
    Zeit_Jetzt_Minuten_Numerisch = Zeit_Jetzt.getMinutes();
    Zeit_Jetzt_Sekunden_Numerisch = Zeit_Jetzt.getSeconds();
    Zeit_Vormonat        = new Date(Zeit_Jetzt_Jahr_Numerisch, Zeit_Jetzt_Monat_Numerisch-1, 1);
    Zeit_VormonatErsterTag = Zeit_Vormonat.getDay();

    if(Zeit_Jetzt_Jahr_Numerisch < 100)
    {Zeit_Jetzt_Jahr_Numerisch+=1900;}

    if(Zeit_Jetzt_Jahr_Numerisch < 100)
    {Zeit_Jetzt_Jahr_Numerisch+=1900;}
}

function Zeit_NumerischNachString_MitVornull(NumerischerWert)
{
    Zeit_Jetzt_Kette=NumerischerWert.toString();

    // auf Vornull prüfen
    if (NumerischerWert < 10)
    {Zeit_Jetzt_Kette='0' + Zeit_Jetzt_Kette;}

    return Zeit_Jetzt_Kette;
}

// ##### Funktionen des Kalenders #####

// +++++ TR erzeugen +++++

function Anzeige_Kalender_HTMLCode_TR_Erzeugen(
    FarbeBackground,
    BreiteAlsString,
    HoeheAlsString,
    Ausrichtung_Horizontal,

```




```

        Ausrichtung_Vertikal,
        SpaltenSpannweiteAlsString,
        SchriftGroesse,
        SchriftFarbe,
        SchriftArt,
        Kette
    )

// erweitert Anzeige_Kalender_HTMLCode
{
// +++++ TR erzeugen: Beginn-Tag
Anzeige_Kalender_HTMLCode= Anzeige_Kalender_HTMLCode + '<TR>';

// +++++ TD (Zelle) erzeugen

// ----- Tag-Beginn
Anzeige_Kalender_HTMLCode= Anzeige_Kalender_HTMLCode + '<TD>';

// ----- Attribut Background
if (FarbeBackground != "")
{Anzeige_Kalender_HTMLCode= Anzeige_Kalender_HTMLCode + ' BGCOLOR="' + FarbeBackground + '"';}

// ----- Attribut Breite
if (BreiteAlsString != "")
{Anzeige_Kalender_HTMLCode= Anzeige_Kalender_HTMLCode + ' WIDTH=' + BreiteAlsString;}

// ----- Attribut Höhe
if (HoeheAlsString != "")
{Anzeige_Kalender_HTMLCode= Anzeige_Kalender_HTMLCode + ' HEIGHT=' + HoeheAlsString;}

// ----- Attribut Ausrichtung horizontal
if (Ausrichtung_Horizontal != "")
{Anzeige_Kalender_HTMLCode= Anzeige_Kalender_HTMLCode + ' ALIGN=' + Ausrichtung_Horizontal;}

// ----- Attribut Ausrichtung vertikal
if (Ausrichtung_Vertikal != "")
{Anzeige_Kalender_HTMLCode= Anzeige_Kalender_HTMLCode + ' VALIGN=' + Ausrichtung_Vertikal;}

// ----- Attribut Spannweite
if (SpaltenSpannweiteAlsString != "")
{Anzeige_Kalender_HTMLCode= Anzeige_Kalender_HTMLCode + ' COLSPAN=' + SpaltenSpannweiteAlsString;}

// ----- Tag-Ende
Anzeige_Kalender_HTMLCode= Anzeige_Kalender_HTMLCode + '>';

// ----- Inhalt der Zelle als Font mit Text
Anzeige_Kalender_HTMLCode=
    Anzeige_Kalender_HTMLCode
    + ' <FONT SIZE=' + SchriftGroesse
    + ' COLOR=' + SchriftFarbe
    + ' FACE="' + SchriftArt + '"
    + '>'
    + Kette
    + '</FONT>'

// ----- Ende-Tag
Anzeige_Kalender_HTMLCode= Anzeige_Kalender_HTMLCode + '</TD>'

// +++++ TR erzeugen: Ende-Tag
Anzeige_Kalender_HTMLCode= Anzeige_Kalender_HTMLCode + '</TR>';
}

// +++++ TD erzeugen +++++
+++++

function Anzeige_Kalender_HTMLCode_TD_Erzeugen(
    Inhalt,
    HintergrundFarbe,
    SchriftFarbe,
    SchriftGroesse,
    SchriftArt,
    BorderColor
)

// erweitert Anzeige_Kalender_HTMLCode
// alle Variablen alle global, damit nicht pro Funktionsaufruf die Variablen neu lokal erzeugt werden.
{
// +++++ Inhalt auf FETT setzen
Inhalt='<B>' + Inhalt + '</B>';

// +++++ TD erzeugen

```



```
// Inhalt der TD ist eine Tabelle
Anzeige_Kalender_HTMLCode= Anzeige_Kalender_HTMLCode
    + '<TD BGCOLOR="' + BorderColor + '">'
    + '<TABLE BORDER=0'
    + '    CELLSPACING=0'
    + '    CELLPADDING=5'
    + '    WIDTH=100%'
    + '    HEIGHT=100%'
    + '>'

Anzeige_Kalender_HTMLCode_TR_Erzeugen(HintergrundFarbe,'30','30','center','middle',"
    SchriftGroesse,SchriftArt,Inhalt);

Anzeige_Kalender_HTMLCode= Anzeige_Kalender_HTMLCode
    + '</TABLE>'
    + '</TD>';
}

// +++++ Tag (nicht Sonntag) TD erzeugen ++++++

function Anzeige_Kalender_Tag_HTMLCode_TD_Erzeugen(Inhalt,Heute)
// erweitert Anzeige_Kalender_HTMLCode
// Heute true, so Tag = heute
// alle Variablen alle global, damit nicht pro Funktionsaufruf die Variablen neu lokal erzeugt werden.
{
    // Annahme: Tag ist nicht heute
    Anzeige_Kalender_Tag_Kette1=Tag_HintergrundFarbe;
    Anzeige_Kalender_Tag_Kette2=Tag_SchriftFarbe;
    Anzeige_Kalender_Tag_Kette3=Kalender_Kopf_HintergrundFarbe;

    // prüfen auf Tag == heute
    if (Heute)
    {
        Anzeige_Kalender_Tag_Kette1=Tag_Heute_HintergrundFarbe;
        Anzeige_Kalender_Tag_Kette2=Tag_Heute_SchriftFarbe;
        Anzeige_Kalender_Tag_Kette3=Tag_SchriftFarbe;
    }

    Anzeige_Kalender_HTMLCode_TD_Erzeugen(
        Inhalt,
        Anzeige_Kalender_Tag_Kette1,
        Anzeige_Kalender_Tag_Kette2,
        Tag_SchriftGroesse,
        Tag_SchriftArt,
        Anzeige_Kalender_Tag_Kette3
    );
}

// +++++ Sonntag TD erzeugen ++++++

function Anzeige_Kalender_SonnTag_HTMLCode_TD_Erzeugen(Inhalt,Sonntag)
// erweitert Anzeige_Kalender_HTMLCode
// Sonntag true, so Tag = Sonntag
// alle Variablen alle global, damit nicht pro Funktionsaufruf die Variablen neu lokal erzeugt werden.
{
    // Annahme: Tag ist nicht Sonntag
    Anzeige_Kalender_Sonntag_Kette1=Tag_HintergrundFarbe;
    Anzeige_Kalender_Sonntag_Kette2=Tag_Sonntag_SchriftFarbe;
    Anzeige_Kalender_Sonntag_Kette3=Kalender_Kopf_HintergrundFarbe;

    // prüfen auf Tag == Sonntag
    if (Sonntag)
    {
        Anzeige_Kalender_Sonntag_Kette1=Tag_Heute_Sonntag_HintergrundFarbe;
        Anzeige_Kalender_Sonntag_Kette2=Tag_Heute_Sonntag_SchriftFarbe;
        Anzeige_Kalender_Sonntag_Kette3=Tag_Sonntag_SchriftFarbe;
    }

    Anzeige_Kalender_HTMLCode_TD_Erzeugen(
        Inhalt,
        Anzeige_Kalender_Sonntag_Kette1,
        Anzeige_Kalender_Sonntag_Kette2,
        Tag_SchriftGroesse,
        Tag_SchriftArt,
        Anzeige_Kalender_Sonntag_Kette3
    );
}
```



```
// +++++ Kalender erzeugen und anzeigen ++++++

function Anzeige_Kalender()
// periodischer Aufruf per Timer NUR zur Erkennung des Tages- oder Monats- oder Jahreswechsel, falls
// genau zu diesen Zeitpunkten der Kalender aktiv ist.
// Es spart Ressourcen, wenn Timer nicht benutzt wird !!
// Kalender_Timer_Verwenden auf false für Timer nicht verwenden
//          auf true für Timer verwenden
//
// alle Variablen alle global, damit nicht pro Funktionsaufruf die Variablen neu lokal erzeugt werden.
{
// +++++ aktuelle Zeitangaben holen
Zeit_AktuelleAngabenHolen(); // füllt
// Zeit_Jetzt
// Zeit_Jetzt_Tag_Numerisch
// Zeit_Jetzt_Monat_Numerisch
// Zeit_Jetzt_Jahr_Numerisch
// Zeit_Jetzt_Stunden_Numerisch
// Zeit_Jetzt_Minuten_Numerisch
// Zeit_Jetzt_Sekunden_Numerisch
// Zeit_Vormonat
// Zeit_VormonatErsterTag

// +++++ Start- und Endwert der Tage für Erzeugung des Kalenders holen

// ----- Erster Tag des Vormonats als Startwert
var Start = Zeit_VormonatErsterTag;
if(Start > 0)
{Start--;} // minus 1
else
{Start = 6;} //

// ----- Endwert für Monatstage ermitteln
// Annahme: Monat hat 31 Tage
var Stop = 31;

// für alle Monate mit 30 Tagen korrigieren
if( (Zeit_Jetzt_Monat_Numerisch==4)
|| (Zeit_Jetzt_Monat_Numerisch==6)
|| (Zeit_Jetzt_Monat_Numerisch==9)
|| (Zeit_Jetzt_Monat_Numerisch==11)
)
{--Stop;}

// für Februar korrigieren: 29 oder 28 Tage, also Schaltjahr
if(Zeit_Jetzt_Monat_Numerisch==2)
{
// Februar
// Annahme: Kein Schaltjahr, also 28 Tage
Stop=28;

// Schaltjahr mit 29 Tagen ermitteln
if ((Zeit_Jetzt_Jahr_Numerisch%4) == 0)
{Stop++;} // 29

if ( (Zeit_Jetzt_Jahr_Numerisch%100) == 0)
{Stop--;} // wieder 28

if ((Zeit_Jetzt_Jahr_Numerisch%400) ==0 )
{Stop++;} // 29
}

// +++++ HTML-Code des Kalenders als Tabelle erzeugen

// +++++ äusserste Tabelle also die des Kalenders insgesamt
Anzeige_Kalender_HTMLCode=
    '<TABLE ID="TABLE_Kalender"'
    + ' BORDER=0'
    + ' CELLPADDING=1'
    + ' CELLSPACING=0'
    + ' STYLE="position:absolute;'
    + 'left:' + Kalender_Left + ';'
    + 'top:' + Kalender_Top
    + '";'
    + '>';
```



```
// +++++ 1. TR des Kalenders als Anzeige aktueller Monat und aktuelles Jahr in Fettschrift

// ----- Text der 1. TR ermitteln: aktueller Monat und aktuelles Jahr in Fettschrift
Anzeige_Kalender_Kette+<B>';
Anzeige_Kalender_Kette=MonatsNamenFeld[Zeit_Jetzt_Monat_Numerisch-1]; // Monatsname als String
Anzeige_Kalender_Kette+= ' ';
Anzeige_Kalender_Kette+=Zeit_Jetzt_Jahr_Numerisch.toString(); // Jahr zu String
Anzeige_Kalender_Kette+</B>';

// ----- Abstand zum Nachfolger
Anzeige_Kalender_Kette+<BR>';
Anzeige_Kalender_Kette+<BR>';

// ----- und erzeugen
Anzeige_Kalender_HTMLCode_TR_Erzeugen(
    Kalender_Kopf_HintergrundFarbe,"","center','middle','7',
    Kalender_Kopf_SchriftGroesse,
    Kalender_Kopf_SchriftFarbe,
    Kalender_Kopf_SchriftArt,
    Anzeige_Kalender_Kette
);

// +++++ 2. TR Spaltenkopf der Monatstage; pro Spalte ein TD
Anzeige_Kalender_HTMLCode+<TR>';

// ----- Tage der Woche abklappen: Woche beginnt mit Montag
for(Anzeige_Kalender_TagesZahler=0;
    Anzeige_Kalender_TagesZahler < TagKurzNamenFeld_Laenge;
    Anzeige_Kalender_TagesZahler++)
{
    // ----- Text als Kurzname des Tages holen
    Anzeige_Kalender_Kette=TagKurzNamenFeld[Anzeige_Kalender_TagesZahler];

    // ----- und Text erzeugen
    Anzeige_Kalender_HTMLCode_TD_Erzeugen(
        Anzeige_Kalender_Kette,
        Kalender_Kopf_HintergrundFarbe,
        Kalender_SpaltenKopf_SchriftFarbe,
        Tag_SchriftGroesse,
        Kalender_Kopf_SchriftArt,
        Kalender_Kopf_HintergrundFarbe
    );
}

Anzeige_Kalender_HTMLCode+</TR>'; // Ende der Wocheneinteilung in 7 Tage (Spaltenköpfe)

// +++++ ab 3. TR erfolgt die Wochenauflistung
//   pro Woche 1 Zeile
//   pro Zeile 1 TR
//   ab 3. TR im gesamten Kalender

// ----- init des Tageszähler des Monats
Anzeige_Kalender_TagesZahler = 1;

// ----- Wochen abklappen also Zeilen (TR)
for(Anzeige_Kalender_Zahler_TR=0;
    Anzeige_Kalender_Zahler_TR < 6;
    Anzeige_Kalender_Zahler_TR++)
{
    // - - - aktuelle Woche (Zeile) erzeugen
    Anzeige_Kalender_HTMLCode+<TR>';

    // - - - pro Spalte 1 TD erzeugen: Nur die ersten 6 Tage der Woche, OHNE Sonntag !
    for(Anzeige_Kalender_Zahler_TD=0;
        Anzeige_Kalender_Zahler_TD < 6; // ohne Sonntag !
        Anzeige_Kalender_Zahler_TD++)
    {
        // - - - prüfen ob 1. TR-Zeile noch nicht abgeklappt wurde
        if( (Anzeige_Kalender_Zahler_TR == 0)
            && (Anzeige_Kalender_Zahler_TD < Start)
        )
        {
            // 1. TR und nur 1. TD
            Anzeige_Kalender_Tag_HTMLCode_TD_Erzeugen('&#160;','false);
```



```

    }
    else
    {
        // 1. TR ab 2.TD oder ab 2.TR

        // - - - prüfen ob alle Tage bereits abgeklappert wurden
        if (Anzeige_Kalender_TagesZahler > Stop)
        {
            // alle Tage sind abgeklappert
            Anzeige_Kalender_Tag_HTMLCode_TD_Erzeugen('&#160;','false');
        }
        else
        {
            // 1. TR ab 2.TD oder ab 2.TR
            // es sind noch Tage abzuklappen

            // - - - prüfen ob aktueller Tag der Tag von JETZT ist, also Heute ist
            if(Anzeige_Kalender_TagesZahler==Zeit_Jetzt_Tag_Numerisch)
            {
                // 1. TR ab 2.TD oder ab 2.TR
                // es sind noch Tage abzuklappen
                // aktueller Tag ist heute

                Anzeige_Kalender_Tag_HTMLCode_TD_Erzeugen(Anzeige_Kalender_TagesZahler,true);
            }
            else
            {
                // 1. TR ab 2.TD oder ab 2.TR
                // es sind noch Tage abzuklappen
                // aktueller Tag ist nicht heute
                Anzeige_Kalender_Tag_HTMLCode_TD_Erzeugen(Anzeige_Kalender_TagesZahler,false);
            }

            // - - - Tageszähler erhöhen für nächsten Tag als neuen aktuellen
            Anzeige_Kalender_TagesZahler++;
        }
    }
}

// - - - ALLE TD der ersten 6 Tage der aktuellen Woche (Zeile) wurden erzeugt
//      offen ist der 7. Tag, also der Sonntag (falls vorhanden)

//      prüfen ob alle Tage abgeklappert wurden, also ob es keinen Sonntag gibt
if(Anzeige_Kalender_TagesZahler > Stop)
{
    // keine Tage mehr abzuklappen, kein Sonntag mehr da
    Anzeige_Kalender_SonnTag_HTMLCode_TD_Erzeugen('&#160;','false');
}
else
{
    // - - - Es sind noch Tage abzuklappen, also 7. Tag = Sonntag vorhanden

    // - - - prüfen ob der aktuelle Tag der von heute ist,
    //      wenn ja, dann ist heute Sonntag
    if(Anzeige_Kalender_TagesZahler==Zeit_Jetzt_Tag_Numerisch)
    {
        // heute ist Sonntag, also den Tag mit anderem Hintergrund erzeugen
        // als alle anderen Sonntage
        Anzeige_Kalender_SonnTag_HTMLCode_TD_Erzeugen(Anzeige_Kalender_TagesZahler,true);
    }
    else
    {
        // heute ist nicht Sonntag, also den Tag dem normalen Hintergrund für Sonntage erzeugen
        Anzeige_Kalender_SonnTag_HTMLCode_TD_Erzeugen(Anzeige_Kalender_TagesZahler,false);
    }

    // - - - Tageszähler erhöhen für nächsten Tag als neuen aktuellen
    Anzeige_Kalender_TagesZahler++;
}

// - - - aktuelle Woche also TD beenden
Anzeige_Kalender_HTMLCode+="|>";
}

// +++++ Ende des Kalenders
Anzeige_Kalender_HTMLCode+="

|  |

```



```
// +++++ und Kalenderdaten neu rendern
document.all.ID_Kalender.innerHTML = Anzeige_Kalender_HTMLCode;

// +++++ prüfen ob der Timer zur Erkennung der Tages-, Monats- und Jahreswechsel
// verwendet werden soll
if (Kalender_Timer_Verwenden)
{window.setTimeout('Anzeige_Kalender()',5000);}
}

// -->
</SCRIPT>
</HEAD>

<BODY BGCOLOR=#3A6EA5>
<!-- BODY mit identischer Hintergrundfarbe wie laut Kalender_Kopf_HintergrundFarbe -->

<DIV ID="ID_Kalender"></DIV>
<!-- Der DIV wird nach seiner Erzeugung erst gefüllt: per .innerHTML -->

<SCRIPT LANGUAGE="JScript">
<!--
Anzeige_Kalender();
// -->
</SCRIPT>
</BODY>
</HTML>
```

6. Text

6.1. vertikale Linien zwischen Texten

IE stellt senkrechte Linien schmäler dar als Netscape

Eine Spalte:

```
<TABLE BORDER="0" CELLPADDING="2" WIDTH="200">
<TR>
<TD WIDTH="0" BGCOLOR="black"><BR></TD> <!-- <BR> ist Zelleninhalt -->
<TD WIDTH="200">hier steht der text</TD>
<TD WIDTH="0" BGCOLOR="black"><BR></TD> <!-- <BR> ist Zelleninhalt -->
</TR>
</TABLE>
```

Zwei Spalten:

```
<TABLE BORDER="0" CELLPADDING="1" WIDTH="460" CELSPACING="8">
<TR>
<TD WIDTH="0" BGCOLOR="red"><BR></TD> <!-- <BR> ist Zelleninhalt -->
<TD WIDTH="250"><H3>layout in spalten</H3>
<DIV ALIGN="justify">hier steht der text der ersten spalte</DIV> <!-- Blocksatz -->
</TD>
<TD WIDTH="0" BGCOLOR="red"><BR></TD> <!-- <BR> ist Zelleninhalt -->
</TD WITH="250">
<DIV ALIGN="justify">hier steht der text der zweiten spalte</DIV> <!-- Blocksatz -->
</TD>
<TD WIDTH="0" BGCOLOR="red"><BR></TD> <!-- <BR> ist Zelleninhalt -->
</TR>
</TABLE>
```

6.2. Wortlogo unabhängig von aktueller Bildschirmauflösung des Users stets zentriert im Browserfenster anzeigen

```
<TABLE BORDER=0
        ALIGN="center"
        WIDTH="100%"
        HEIGHT="100%"
        CELSPACING="0"
>
<TR>
<TD>
<CENTER> logo_text </CENTER>
</TD>
</TR>
</TABLE>
```



6.3. CSS schwebender Text von rechts nach links bei 640 Pixel Bildschirmbreite

```
<HTML>
<HEAD>
<SCRIPT LANGUAGE="JavaScript">
<!--
    // var objekt NICHT deklarierbar, da Typ der Variable je nach Browserart per Zuweisung
    // erzeugt wird

    function starte_bewegung()
    {
        if (document.layers)                // Netscape
        {
            objekt=document.bewegtes_objekt; // Typ laut Netscape
            bewegen();
        }
        else
        {
            if (document.all)                // Microsoft
            {
                objekt=bewegtes_objekt.style; // Typ laut Microsoft
                bewegen();
            }
        }
    }

    function bewegen() // rekursive Funktion mit setTimeout auf 200 Millisekunden Wartezeit
    {
        ganze_zahl=parseInt(objekt.left); // parseInt erzeugt ganze Zahl, also auch negativ
        objekt.left=ganze_zahl - 3;       // Bewegung um 3 Pixel

        if (ganze_zahl < 0) {objekt.right=640}
        setTimeout('bewegen()',200);
    }
//-->
</SCRIPT>
</HEAD>

<BODY onload="starte_bewegung()">
// Microsoft unterstützt CSS-Layer !
<DIV ID="bewegtes_objekt"
    STYLE="position:absolute;top:100px;left:250px;z-index:2;height:50px;width:150px"
    // Achtung: STYLE alles 1 Zeile da Zeichenkettenparameter
    // keine Blanks wegen Netscape
>
<H1 ALIGN="center">bewegter text</H1>
</DIV>
</BODY>
</HTML>
```

7. Url

7.1. Umleitung auf andere Url

Variante 1 in HTML

```
<META HTTP.EQUIV="Refresh"
      CONTENT="3;URL=http://....."
>

// 3 Sekunden warten, dann umleiten nach http:// .....
```

Variante 2 in Javascript

```
<SCRIPT LANGUAGE="JavaScript">
    top.location.href="http:// ....."
</SCRIPT>
```

7.2. Mehrfachverweis auf ein und dieselbe Url innerhalb einer HTML-Seite

Link stets mit gleicher Schreibweise kodieren (mit / am Ende oder ohne / am Ende), da sonst der Browser versucht, die Daten aus dem Netz zu laden anstelle aus dem Browsercache.

7.3. Möglichkeiten des Laden eines Dokumentes

innerhalb eines Frame: rechtlich nicht zulässig (Abmahnungsgefahr)

7.3.1. Laden eines fremden Dokumentes ohne Framedarstellung

innerhalb eines Frame: rechtlich nicht zulässig (Abmahnungsgefahr)



Variante 1:

innerhalb eines neuen Fensters:

```
<A      HREF=http://www.test.de
      TARGET="_blank">
</A>
```

innerhalb des obersten Fensters:

```
<A      HREF=http://www.test.de
      TARGET="_top">
</A>
```

Variante 2:

```
<HTML>
<HEAD>
  <SCRIPT LANGUAGE="JavaScript">
    <!--
      function laden()
      { location.href = "http://www.test.de"; }
    // -->
  </SCRIPT>
</HEAD>

<BODY>
  <FORM>
    <INPUT TYPE="button"
      VALUE="www.test.de anwaehlen "
      onClick="laden()">
  </FORM>
</BODY>
</HTML>
```

7.3.2. Eigenes Dokument wird durch fremde Webseite geladen

innerhalb eines Frame: rechtlich nicht zulässig (Abmahnungsgefahr)

7.3.2.1. Copyright-Meldung auf fremden Host erzeugen**Variante 1:**

```
<HTML>
<HEAD>
<SCRIPT TYPE="text/javascript" LANGUAGE="JavaScript1.2">
  <!--
    // In diesem Beispiel schreibt das Script Text auf die "entführte" Seite!
    var HostUrl="www.test.de";

    if(
      (parent !=null)
      && (parent != self)
    )
    {
      var host=parent.location.hostname;

      if(host != HostUrl)
      {
        document.write( "<CENTER>Diese Seite wurde ausgeliehen bei "
          + "<A HREF=\""
          + location.href
          + "\" TARGET=\"_parent\">"
          + HostUrl
          + "</A></CENTER>")
      }
    }
  //-->
</SCRIPT>
</HEAD>
<BODY>
</BODY>
</HTML>
```

Variante 2:

```
<BODY>
.....
if (
  && (parent != null)
  && (parent != self)
)
{
```




```

var meine_url = "http://www.meine_angebot.de";
var mein_host_name = "http://" + meine_url;

var fremder_host_name = parent.location.hostname;

if ( fremder_host_name != mein_host_name)
{
    document.write(      " Diese Seite liegt auf "
        + "<A"
        +      "HREF=\"\" + location.href + \"\"\"
        +      " TARGET=\"_parent\"
        + ">"
        + " und stammt von "
        + mein_host_name
        + "</A>"
    );
}
}
.....
</BODY>

```

7.3.2.2. *Framedarstellung der fremden Webseite sofort und ohne Bildschirmmeldung aufheben*

```

<HTML>
<HEAD>
<SCRIPT TYPE="text/javascript" LANGUAGE="JavaScript1.2">
<!--
    var EigenerHost="http://www.test.de";
    // window.location.hostname ist Leerkette, wenn Browser offline
    var StartSeite="_start.html";

    function InaktiveFrameDarstellungAktivieren()
    {
        if (top.frames.length == 0)
        {
            // keine Frame-Darstellung aktiv, also diese aktivieren
            top.location.href = StartSeite; // muss das FRAMESET enthalten !
        }
    }
    if (      (parent != null) // Elternobjekt existiert
        && (parent != self) // Eltern sind vorhanden Eltern: dieses Dokument (self) ist ein Kind
    )
    {
        // aktuellen ElternHost ermitteln
        var ElternHost=parent.location.hostname;

        // ElternHost prüfen ob eigener und nicht leer, also Browser online ist
        if (      (ElternHost != "") // Browser ist online
            && (ElternHost != EigenerHost)
        )
        {
            // fremder Host, also als oberstes Fenster nun die Startseite anzeigen
            // und damit den eigenen Host aktivieren
            top.location.href=EigenerHost + '/' + StartSeite;
        }
        else
        {
            // eigener Host und/oder Browser ist offline
            InaktiveFrameDarstellungAktivieren();
        }
    }
    else
    {
        // Elternobjekt existiert nicht und/oder dieses Dokument (self) ist kein Kind
        InaktiveFrameDarstellungAktivieren();
    }
}
//-->
</SCRIPT>
</HEAD>
<BODY>
</BODY>
</HTML>

```

7.3.3. *Frame-Dokument mit zugehörigem Frameset laden*

Sollte eine Frame per Suchmaschine indiziert sein und der User die Url des Frames bekommt, die er anklickt, so soll dieser Frame mit dem zugehörigen Frameset geladen werden.



```

<SCRIPT TYPE="text/javascript" LANGUAGE="JavaScript1.2">
<!--
    var StartSeite="_start.html";

    if (top.frames.length == 0)
    {
        // keine Frame-Darstellung aktiv, also diese aktivieren
        top.location.href = StartSeite; // muss das FRAMESET enthalten !
    }
//-->
</SCRIPT>

```

7.3.4. Mehrere Frameinhalte gleichzeitig ändern

HTML-Kodierung:

```

.....
<FRAMESET ...>
    <FRAME NAME="frame_name_1 SRC=html_datei_in_frame_1">
        .....
    <FRAME NAME="frame_name_n SRC=html_datei_in_frame_n">
</FRAMESET>

```

Auslösung der Frameänderung kann nur von einem Frame aus erfolgen
z.B. von html_datei_in_frame_1

```

....
<SCRIPT LANGUAGE="Javascript">
    <!--
        function aenderung_variante1()
        {parent.frame_name_1.location="neue_html_datei_1";
        ....
        parent.frame_name_n.location="neue_html_datei_n";}

        oder

        function aenderung_variante2
        (neue_html_datei_1,...,neue_html_datei_n)
        {parent.frame_name_1.location=neue_html_datei_1+".html";
        .....
        parent.frame_name_n.location=neue_html_datei_n+".html";}

// -->
</SCRIPT>
....
<INPUT TYPE="button"
    VALUE="Start der Aenderung" onClick="aenderung_variante1()">
oder <A HREF="Javascript:aenderung_variante1()"> Start der Aenderung </A>
oder <A HREF="Javascript:aenderung_variante1()">
    <IMG SRC=start_button.gif">
</A>
bzw. <A HREF=Javascript:aenderung_variante2
    ('neue_html_datei_1_ohne_suffix', .... 'neue_html_datei_n_ohne_suffix')
    > Start der Aenderung
</A>

```

Hinweis: Standardsuffix einer HTML-Datei ist html oder htm.

8. Zugriffsschutz

8.1. robots.txt und Suchmaschinen

Die Scanvorgänge von Suchmaschinen lassen sich beeinflussen über

META-Tag
ASCII-Datei robots.txt

Der Nutzung von Meta-Tags ist die Anwendung von robots.txt vorzuziehen.

8.1.1. robots.txt und ihre Lage auf dem Server

NUR im Hauptverzeichnis des Webs auf dem Server

Bsp. www.test.de/robots.txt

Dateinamen klein schreiben
als Pfadtrenner nur / und nicht \ kodieren
sobald sich diese Datei auf dem Server befindet, ist sie wirksam

8.1.2. robots.txt die nicht auf dem Server vorhanden ist (Scan-Standard)

Existiert keine robots.txt im Hauptverzeichnis, so gilt folgender Scan-Standard:

JEDE Suchmaschine darf alles (auch Dateien etc.) abscannen.



8.1.3. robots.txt und Aufbau**8.1.3.1. robots.txt als Zeilenfolge-Script erstellen**

als reinen ASCII-Text, also z.B. mit Notepad
keine Formate etc. wie z.B. unter WORD !!

jede Zeile per Return-Taste-Eingabe beenden

Kodierung ist in ihrer nachfolgend gezeigten Groß-Kleinschreibung einzuhalten

8.1.3.2. robots.txt und ihre Blöcke

Block: Folge von Zeilen, die genau eine Scan-Variante beschreiben

beliebige Anzahl von Blöcken sind in robots.txt möglich

Trennung von Blöcken erfolgt nicht, da jeder neue Block an seiner 1. Zeile erkannt wird

8.1.3.2.1. robots.txt-Block: Aufbau Zeile 1 (Zulassen bzw. Sperren von Suchmaschinen)

Diese Zeile dient zum Ein- bzw. Ausschließen von Suchmaschinen für den Scan des HTML-Dokumentes

8.1.3.2.1.1. robots.txt und Zeile 1: ALLE Suchmaschinen dürfen scannen

user-agent: *

Hinweis: zwischen Doppelpunkt und * bzw. Name der Suchmaschine ist ein Leerzeichen zu kodieren

8.1.3.2.1.2. robots.txt und Zeile 1: KEINE Suchmaschine darf scannen

Diese Variante ist nicht kodierbar.

8.1.3.2.1.3. robots.txt und Zeile 1: Eine bestimmte Suchmaschinen darf scannen

user-agent: WebCrawler

WebCrawler darf scannen

Sollen mehrere Suchmaschinen ausgewählt werden, so muss pro Suchmaschine ein eigener Block mit Zeile 1 wie im Beispiel kodiert werden. Der Name der Suchmaschine muss dafür exakt bekannt sein.

Hinweis: zwischen Doppelpunkt und * bzw. Name der Suchmaschine ist ein Leerzeichen zu kodieren

8.1.3.2.1.4. robots.txt und Zeile 1: Eine bestimmte Suchmaschinen darf NICHT scannen

user-agent: WebCrawler

WebCrawler darf scannen, aber ab Zeile 2 sind alle Inhalte als gesperrt zu markieren

Sollen mehrere Suchmaschinen ausgewählt werden, so muss pro Suchmaschine ein eigener Block mit Zeile 1 wie im Beispiel kodiert werden. Der Name der Suchmaschine muss dafür exakt bekannt sein.

Hinweis: zwischen Doppelpunkt und * bzw. Name der Suchmaschine ist ein Leerzeichen zu kodieren

8.1.3.2.2. robots.txt-Block: Aufbau ab Zeile 2

(Zulassen bzw. Sperren von Inhalten auf dem Server für Suchmaschinen, die scannen)

8.1.3.2.2.1. robots.txt und ab Zeile 2: Alle Daten auf dem Server dürfen gescannt werden

Disallow:

es muss **nur noch diese Zeile** kodiert werden

Daten: Haupt- und alle Unterverzeichnisse und alle HTML-Dateien mit deren Bildern etc.

Für folgende Fälle ist diese Variante nicht zulässig:

Verzeichnisse, die mit User-Passwort geschützt seien sollen
Verzeichnisse, die private Dinge des Web-Programmierers enthalten
ausgewählte Dateien wie HTML-Dateien die niemanden was angehen

8.1.3.2.2.2. robots.txt und ab Zeile 2: Alle Daten auf dem Server dürfen NICHT gescannt werden

Disallow: /

sperrt das Hauptverzeichnis und damit alle Unterverzeichnisse

Hinweis: zwischen Doppelpunkt und / ist ein Leerzeichen zu kodieren

8.1.3.2.2.3. robots.txt und ab Zeile 2: Bestimmte Daten auf dem Server dürfen NICHT gescannt werden**8.1.3.2.2.3.1. robots.txt und ab Zeile 2: Bestimmte Verzeichnisse auf dem Server dürfen NICHT gescannt werden**

Disallow: /pfad_angabe/



pfad_abgabe: Pfadkodierung mit / für dasjenige Verzeichnis, das nicht gescannt werden darf
abschließendes / nicht vergessen !

pro Verzeichnis ist jeweils eine eigene Zeile zu kodieren

Hinweis: zwischen Doppelpunkt und dem ersten / ist ein Leerzeichen zu kodieren

8.1.3.2.2.3.2. robots.txt und ab Zeile 2: Bestimmte HTML-Dateien auf dem Server dürfen NICHT gescannt werden

Disallow: /pfad_angabe/datei_angabe

pfad_abgabe/dateiangabe: Pfadkodierung mit / für diejenige Datei, die nicht gescannt werden darf

pro Datei ist jeweils eine eigene Zeile zu kodieren

Hinweis: zwischen Doppelpunkt und dem ersten / ist ein Leerzeichen zu kodieren

Diese Kodierungsvariante ist wichtig für PHP-Skripte und Dateien, die Passwörter verwalten bzw. enthalten. Sollte die Passwortverwaltung durch den Provider selbst angeboten werden, so prüfen, ob das vorgegebene Verzeichnis mit seinen Dateien vor dem Scannen geschützt sind !

8.1.3.2.2.4. robots.txt und ab Zeile 2: NUR bestimmte Daten auf dem Server dürfen gescannt werden

Diese Variante ist nicht kodierbar.

8.1.3.3. robots.txt und Beispiele

Beispiel: WebCrawler darf ALLES scannen

user-agent: WebCrawler

Disallow:

Beispiel: WebCrawler darf NICHTS scannen

user-agent: WebCrawler

Disallow: /

Beispiel: WebCrawler darf ALLES außer /test/ scannen

user-agent: WebCrawler

Disallow: /test/

Beispiel: WebCrawler darf ALLES außer /test/test.htm scannen

user-agent: WebCrawler

Disallow: /test/test.htm

Beispiel: ALLE Suchmaschinen dürfen ALLES scannen (robots.txt kann entfallen)

user-agent: *

Disallow:

Beispiel: ALLE Suchmaschinen dürfen NICHTS scannen

user-agent: *

Disallow: /

Beispiel: ALLE Suchmaschinen dürfen ALLES außer /temp/ scannen

user-agent: *

Disallow: /temp/

Beispiel: ALLE Suchmaschinen dürfen ALLES außer /test/test.htm scannen

user-agent: *

Disallow: /test/test.htm

Beispiel: ALLE Suchmaschinen dürfen ALLES außer /test/test.htm UND /temp/ scannen

user-agent: *

Disallow: /test/test.htm

Disallow: /temp/

8.2. .htaccess und .htpasswd und Passwordschutz

Der Schutz von Dateien in einem Ordner auf dem Server ist Sache des Inhabers des Web-Projektes **und nicht Sache** des Serverinhabers. Es steht dem Inhaber des Web-Projektes frei, **seine** Daten zu schützen.

FTP-Zugriffschutz:

Browser sind in der Lage, auch FTP-Ordner anzuzeigen und auf diese zuzugreifen. Manche Web-Anbieter nutzen Internet-Adressen

mit ftp://.... **anstelle von** http://....

als öffentliche Systeme, die standardgemäß per FTP-Protokoll les- und schreibbar sind, wenn der Inhaber der FTP-

Ordner

keinen FTP-Schutz installiert hat. Existiert ein Zugriffschutz, so erscheint mit dem Versuch eines Zugriffs im

Browser ein

Login-Fenster, dass diverse Kennungen verlangt. Diese Kennungen müssen zwischen dem User, der per FTP zugreifen will,



und dem Inhaber der FTP-Ordner **vereinbart** sein.

Das Web-Projekt ist standardgemäß per FTP auch schreibbar (FTP-Upload auf den Server), wenn nicht ein FTP-Zugriffsschutz vereinbart wurde: Für den Schutz des FTP-Upload auf den Server besitzt der Inhaber des Web-Projektes ein Passwort, dass dem Serverinhaber-Inhaber bekannt ist. Dieses Passwort liegt in einer Datenbank des Serverinhabers und wird mit der Password-Eingabe des Inhaber des Web-Projektes abgeglichen, sobald FTP-Zugriff versucht wird. Die Datenbank des Servers ist durch den Inhaber des Web-Projektes nicht einsehbar, da der Serverinhaber Komponenten benutzt, die zugleich ein Login-Interface demjenigen bereitstellt, der FTP aktivieren will. Diese Komponenten werden server-lokal abgearbeitet und gelangen nie auf den PC des FTP-Users (außer das Login-Interface).

Der FTP-Schutz hat nichts mit dem Passwordschutz per .htaccess und .htpasswd zu tun.

Password-Schutz per .htaccess und .htpasswd:

Es können Ordner des Web-Projektes derart geschützt werden, dass ein User nur einen **erlaubten Zugriff** auf Webseiten bekommt. Standardgemäß sind Webseiten immer lesbar, um auf den Client (User-PC) übertragbar zu sein, damit sie der User auch im Browser anschauen kann. Mit Aktivierung des Passwordschutzes ist ein User-individueller Zugriff installiert, der auch für den Inhaber des Web-Projektes gilt. Der Inhaber kann pro User ein Password vergeben. Ein User **ohne Password** bekommt keinen Lesezugriff und kann damit den Inhalt des Ordners im Browser nicht anzeigen lassen.

Schreibzugriff auf Ordner kann nur per FTP erfolgen (siehe oben).

Um Dateien im Ordner auf dem Server vor unberechtigtem Zugriff zu schützen, müssen **pro zu schützenden Ordner** die Dateien .htaccess **und** .htpasswd mit ordnerspezifischen Inhalten hinterlegt werden., wobei der jeweilige Dateiname in Kleinbuchstaben **und mit führendem Punkt** kodiert sein muss. Desweiteren muss der Serverinhaber die Verarbeitung dieser Dateien durch den Server unterstützen. Sollte das der Fall sein **und** existieren beide Dateien im jeweiligen zu schützenden Ordner, so ist der Passwordschutz für diese Ordner sofort aktiv.

Die Dateien **.htaccess** und **.htpasswd** sind reine ASCII-Dateien und sollten z.B. nur mit NOTEPAD.EXE erstellt werden, wobei Formatierungen wie bei Word oder Wordpad und deutsche Umlaute und Sonderzeichen nicht verwendet werden dürfen.

Aufbau der Datei .htaccess

```
AuthType Basic
AuthName "freier Text"
AuthUserFile absoluter_pfad/Ordner/.htpasswd
<Limit GET POST PUT>
require user freier_username_1
....
require user freier_username_n
</Limit>
```

freier Text als Hinweis
muss in " " kodiert sein

absoluter_pfad auf dem Server
immer von der Root aus
Bsp.:

/root/xx/yy/Ordner/.htpasswd

mit root als Platzhalter für die Wurzel der Baumstruktur, welche frei benannt sein kann
xx und yy als Platzhalter für frei benennbare Ordner im Pfad
Ordner als Platzhalter für den frei benennbaren und zu schützenden Ordner

Achtung: Gross-Klein wird unterschieden, also die Namen der Pfadelemente korrekt kodieren !

freier_username_1 bis freier_username_n pro Username genau 1 Zeile mit **require user**

z.B. otto_1 bis otto_n

.... als symbolischer und nicht zu kodierender Platzhalter für die User ab 2 bis n-1

Aufbau der Datei .htpasswd

freier_username_1:verschlusstes_password



....

freier_username_n:verschlusstes_password

pro User laut Zeilen

require user freier_username_1

....

require user freier_username_nin der Datei **.htaccess**genau 1 Zeile in der Datei **.htpasswd**

z.B. otto1:Ho/87YN6YqP89

Doppelpunkt muss kodiert werden

....

als symbolischer und nicht zu kodierender Platzhalter für die User ab 2 bis n-1

Die Passwortverschlüsselung muss anhand einer standardisierten Formel erfolgen, die diverse Anbieter als online zu bedienenden Programms anbieten (eventuell der Serverinhaber selbst):

z.B. <http://www.linux-profis.de/php4>
<http://www.inch.com/info/tech/HOWTOS/htaccess/htpasswd.html>

Es ist Sache des Inhabers des Web-Projektes sich die Verschlüsselung zu besorgen.

9. WebCam

Beispiel: HTML-Seiten-Refresh für WebCam

```
<HEADY>
<META      HTTP-EQUIV="refresh"
           CONTENT="sekundenzahl">
<META      HTTP-EQUIV="Expires"
           CONTENT="Tue, 04 Dec 1997 21:29:026 GMT"> // Datum egal, aber auf jedenfall älter als
                                                    // aktuelles, damit die HTML-Seite als
                                                    // verfallen gilt

<META      HTTP-EQUIV="Pragma"
           CONTENT="nocache"> // Seite nicht in Browsercache speichern

<META      HTTP-EQUIV="Cache-Control"
           CONTENT="nocache"> // Seite nicht aus Browsercache laden

<HEAD>
```

10. PopUp-Fenster ab IE 5.5

```
<HTML>
<HEAD>
<SCRIPT LANGUAGE="JavaScript">
<!--
    // HTML-Inhalt des PopUp-Fensters
    var PopUpFenster_HTMLString =
        'Das ist eine Sprechblase!<BR>Bitte ausserhalb der Sprechblase klicken,<BR>um sie zu
          schliessen !';

    // PopUp-Fenster-Start-Pixel-Koordinaten gegenüber den Bildschirm-Dimensionen
    //      (nicht Browserfenster etc.)
    var PopUpFenster_LinkeObereEcke_X =0; // Spalte
    var PopUpFenster_LinkeObereEcke_Y =0; // Zeile

    // PopUp-Fenster-Dimension in Pixel
    var PopUpFenster_Breite =250; // Spalte
    var PopUpFenster_Hoehe =60; // Zeile

    var PopUpFenster_HintergrundFarbe="yellow";

    // Es werden keine Events verarbeitet !!
    //      also bleibt z.B. <INPUT TYPE=button VALUE="PopUpFenster verstecken" onclick=".....">
    //      wirkungslos

    function PopUpInstanzErzeugen_PopUpAnzeigen_PoUpInstanzLoeschen()
    {
        //      Instanz erzeugen
        var PopUpFenster = window.createPopup();

        //      den Inhalt zuweisen
        PopUpFenster.document.body.innerHTML=PopUpFenster_HTMLString;

        //      Das Aussehen gestalten mit der Eigenschaft .style
        //      z.B. Hintergrundfarbe
```



```

        PopUpFenster.document.body.style.backgroundColor=PopUpFenster_HintergrundFarbe;

        //      anzeigen, aber nicht schliessen
        PopUpFenster.show(PopUpFenster_LinkeObereEcke_X,
                           PopUpFenster_LinkeObereEcke_Y,
                           PopUpFenster_Breite,
                           PopUpFenster_Hoehe,
                           document.body);

        //      Instanz löschen
        PopUpFenster=null;
    }
    PopUpInstanzErzeugen_PopUpAnzeigen_PoUpInstanzLoeschen();

    // Jeder Mausklick außerhalb der PopUp-Fensters schliesst es; Tastaturereignisse jedoch nicht
    // Jedes andere Fenster, das danach geöffnet wird (auch per alert();), schliesst das PopUp-Fenster
    //      automatisch
// -->
</SCRIPT>
</HEAD>
<BODY>
</BODY>
</HTML>

```

11. MP3-Sound-Datei und Streaming Audio

Streaming Audio: Datenstrom beliebiger Art (z.B. Sound) aus dem Netz laden ohne eine Datei lokal auf dem PC erzeugen müssen (ohne Datei-Download)
keine Speicherung als Datei möglich

Quelle: MP3-Dateien, die nicht im Streaming-Audio-Format vorliegen
MP3-Playlist-Datei *.M3U muss existieren z.B. play_list.m3u
pro MP3-Datei ein Zeile mit dem Aufbau: url\mp3_datei.mp3
z.B. http://www.test.de/mp3file/mp3_datei.mp3
MP3-Dateien und Playlist-Datei müssen im gleichen Verzeichnis liegen
z.B. http://www.test.de/mp3file/

HTML-Dokument:
MP3-Files anhoeren

MP3-Files werden als Streaming-Audio geliefert

12. Server Error-Meldung durch eigene Meldung kommentieren

Es können die Standard-Fehlermeldungen des Servers durch je eine eigene Fehlerseite kommentiert werden.
Bekanntester Error ist 404, also Seite nicht gefunden.
Falls es der Serverinhaber zulässt, dass in der Root des Webprojektes die Datei .htaccess verarbeitbar ist, dann kann kommentiert werden.
Pro Server-Error ist eine HTML-Seite zu erstellen bzw. eine HTML-Seite für eine Sammlung von Error-Coden.
In der Root des Web-Projektes muss die Datei **.htaccess** hinterlegt werden (Kleinbuchstaben, führender Punkt, reines ASCII und keine deutschen Umlaute sowie Sonderzeichen).

Aufbau der .htaccess

ErrorDocument XXX /absoluter_pfad/dateiname

XXX	ist der 3-stellige Server Fehlercode z.B. 404
absoluter Pfad	im Web-Projekt auf dem Server, also immer ab Root
dateiname	HTML-Dokument zur Kommentierung des Fehlers laut XXX
Achtung: Die Datei muss korrekt mit Pfad existieren !	

Bsp.: **ErrorDocument** 404 /error_msg/404.htm

pro Fehlercode 1 Zeile

13. Favoriten-Grafik favicon.ico der Webseite

Immer wenn ein IE-Benutzer eine Seite seinen Favoriten hinzufügt, wird vor dem Favorit ein Symbol angezeigt. Dieses ist inhaltlich veränderbar: Liegt im der Root der Webseite im Webpace die Datei favicon.ico (mit genau diese Dateinamen) wird diese als Symbol verwendet. Alternativ kann auch pro HTML-Dokument kodiert werden:

<LINK REL="SHORTCUT ICON" href="http://pfad/dein-icon.ico">

favicon.ico ist ein Icon im Microsoft Windows Icon File Format

16×16 Pixel
32×32 Pixel
Farbtiefe 4 Bit für 16 Farben



8 Bpt für 256 Farben
sinnlos ist wegen geringer Pixelanzahl 24 Bit für 16 Millionen Farben
bis 256 Farben eine transparente Farbe möglich ist.
Man sollte Bildsoftware verwenden, die .ico als Dateityp kennt.



14. Index

*.M3U	39	Bildschirmbreite	31
.Apply();	17	Bildwechsel	17
.createPopup()	38	Blinde Tabelle (ohne Rahmen) mit zellenweisem Hintergrund	21
.filters[0].Apply();	17	Browerdifferenzen bei pixelgenauer Darstellung	12
.filters[0].Play();	17	Browserfenster	30
.htaccess	11	Büro	12
.htaccess für eigene Fehlermeldung bei Server-Error	39	Business.City	12
.htaccess und .htpasswd und Passwordschutz	36	Business.Country	12
.htaccess und Passwordschutz	36	Business.Fax	12
.htpasswd und Passwordschutz	36	Business.Phone	12
.innerHTML	38	Business.StreetAddress	12
.M3U	39	Business.URL	12
.Play();	17	Button	14
.show()	39	Buttonbeschriftung mit mehrzeiligem Text	14
.style.visibility	17	Cellular	12
.style.zoom	21	CLASS	16
.visibility	17	Company	12
.zoom	21	CopyRight-Meldung	32
<STYLE>	15	CSS schwebender Text von rechts nach links	31
3D-Effekt eines Bildes	20	CSS Typen von Links-Layouts definieren	13
a:active	13	Cursor für einen Link ändern	13
a:link	13	Datei speichern	39
a:visited	13	Datei-Download	39
abgeschaltete Grafikanzeige	15	Dateipfad	6
Abmahnungsgefahr	31, 32	Dateipfad absolut	6
absolute Pfade	6	Dateipfad relativ	6
Abteilung	12	Datenstrom	39
active	13	Deckkraft	16
aktuelle Bildschirmauflösung	30	Deckungsstärke	16
Alle Daten auf dem Server dürfen gescannt werden	35	Department	12
Alle Daten auf dem Server dürfen NICHT gescannt werden ...	35	direction	20
ALLE Suchmaschinen dürfen ALLES außer /temp/ scannen...	36	Disallow:	35
ALLE Suchmaschinen dürfen ALLES außer /test/test.htm		Disallow: /	35
scannen	36	Disallow: /pfad_angabe/	35
ALLE Suchmaschinen dürfen ALLES außer /test/test.htm UND		Disallow: /pfad_angabe/datei_angabe	36
/temp/ scannen	36	Dokument Eigenes Dokument wird durch fremde Webseite	
ALLE Suchmaschinen dürfen ALLES scannen	36	geladen	32
ALLE Suchmaschinen dürfen NICHTS scannen	36	Dokument Framedarstellung der fremden Webseite sofort und	
ALLE Suchmaschinen dürfen scannen	35	ohne Bildschirmmeldung aufheben	33
ALLES außer /temp/ scannen	36	Dokument Frame-Dokument mit zugehörigem Frameset	33
ALLES außer /test/ scannen	36	Dokument fremdes Dokument laden ohne Framedarstellung ...	31
ALLES außer /test/test.htm scannen	36	Dokument Laden eines fremden Dokumentes ohne	
ALLES außer /test/test.htm UND /temp/ scannen	36	Framedarstellung	31
ALLES scannen	36	Download	39
Alpha-Filter für HTML-Grafik-Elemente ab IE 5.5	16	DXImageTransform.Microsoft.Fade()	16
ältere Browser	22	eigene Fehlermeldung bei Server-Error	39
Apache-HTTP-Server	6	Eigenes Dokument wird durch fremde Webseite geladen	32
Apply();	17	Einblende eines Bildes	17
Audio-Format	39	Einblende von Bildern mit Bildwechsel	17
Aus- und Einblende von Bildern mit Bildwechsel	17	Eine bestimmte Suchmaschinen darf NICHT scannen	35
Ausgabe von Texten innerhalb einer Tabellenzelle	22	Eine bestimmte Suchmaschinen darf scannen	35
Aus- und Einblende eines Bildes	17	Einrichten eines virtuellen Hosts per Apache-HTTP-Server	6
Autoausfüllen	12	Einrichtung einer Webseite per Apache-HTTP-Server	7
AUTOCOMPLETE=	13	Email	12
Bad Request	11	Emailadresse	12
Beispiel für 3D-Effekt eines Bildes	20	Error 404	12
Beispiel für Aus- und Einblende von Bildern mit Bildwechsel	17	ErrorDocument	12
Beispiel für Schatten eines Bildes	20	Errormeldungen des HTTP-Servers	11
Beispiel für transparente Überlagerung zweier Bilder	17	Fade	16
Beispiel für transparentes INPUT-Element	16	Fax Firma	12
Beispiel für transparentes Vordergrund-Bild auf dem		Faxnummer privat	12
Hintergrundbild	16	Fehlermeldung bei Server-Error	39
Beispiele für Aus- und Einblende eines Bildes	17	Festplattenordner der Webseite	6
Beispiele robots.txt	36	Filter für HTML-Elemente ab IE 5.5	15
Beruf	12	Filter kodieren	15
Bestimmte Daten auf dem Server dürfen NICHT gescannt		filter:Alpha()	16
werden	35	filter:progid:DXImageTransform.Microsoft.Fade()	16
Bestimmte HTML-Dateien auf dem Server dürfen NICHT		filter:shadow()	20
gescannt werden	36	filters[0].Apply();	17
Bestimmte Verzeichnisse auf dem Server dürfen NICHT		filters[0].Play();	17
gescannt werden	35	Firma	12
Bildauflösung	20	FirstName	12
Bilder mit Bildwechsel	17	Formular mit Pulldown-Auswahlfeld mit farbigen Einträgen ...	13
Bilderrahmen aus verschachtelten Tabellen	15	Formular-TEXTAREA ohne Rollbalken	14
Bildschirmauflösung	30		



Framedarstellung der fremden Webseite sofort und ohne Bildschirmmeldung aufheben.....	33	Lage auf dem Server	34
Framedarstellung und Abmahnungsgefahr	31, 32	Land Firma	12
Framedarstellung und rechtliche Zulässigkeit.....	31, 32	Land privat	12
Frame-Dokument mit zugehörigem Frameset laden.....	33	LastName	12
Frameinhalte	34	Layertechnik des IE.....	18
Frameset	33	Leere Zelle mit Hintergrund	22
fremde Webseite	32	LEFTMARGIN	12, 21
Fremdes Dokument laden ohne Framedarstellung.....	31	link	13
FTP-Zugriffschutz	36	Linklayout verändern	13
Gender	12	Link-Layout verändern wenn Maus über Link fährt	13
Geschlecht.....	12	Linkunterstreichung abschalten	13
Grafik	14	location.hostname	6
Grafik zoomen ab IE 5.5.....	20	MARGINHEIGHT	12, 21
Grafikdimension	15	MARGINWIDTH	12, 21
Grafiken	16	Mehrere Frameinhalte gleichzeitig ändern	34
Grafik-Platzhalter.....	15	Mehrfachverweis auf ein und dieselbe Url innerhalb einer HTML-Seite	31
große Tabelle	21	mehrzeiliger Text	14
Handynummer	12	META-Tag	34
Hauptverzeichnis	35	Microsoft	16
Hauptverzeichnis des Webs auf dem Server.....	34	Microsoft Internet Explorer	12
Hintergrund der Zelle	22	Moved Permanently	11
Hintergrundbild.....	16, 22	Moved Temporarily	11
Hintergrundfarbe der Tabelle	22	MP3-Datei anhören	39
Hinweise zur Kodierung von Quelltext in diesem Dokument	3	MP3-Playlist-Datei.....	39
Home.City	12	MP3-Sound-Datei und Streaming Audio.....	39
Home.Country	12	Nachname	12
Home.Fax	12	Neuberechnung des Seitenlayouts.....	15
Home.Phone	12	NICHTS scannen	36
Home.StreetAddress	12	No Content	11
Homepage	12	Not Found.....	12
Homepage der Firma	12	Not Implemented.....	12
Homepage privat.....	12	Notepad	35
Host der Webseite	6	Notes.....	12
hostname	6	Notizen	12
hover	13	NUR bestimmte Daten auf dem Server dürfen gescannt werden	36
htaccess für eigene Fehlermeldung bei Server-Error	39	Office.....	12
htaccess und httpswd und Passwordschutz	36	opacity	16
htaccess und Passwordschutz	36	overflow.....	14
HTML-Elemente.....	15	Password-Zugriffschutz	37
HTML-Seiten-Refresh für WebCam.....	38	Pfade für Dateien.....	6
httpswd und Passwordschutz.....	36	Pfadtrenner	34
HTTP-Server.....	6	pixelgenaue Darstellung.....	12
IE 5.5	15	Platzhalter für Grafik.....	15
IE Alpha-Filter für HTML-Grafik-Elemente ab IE 5.5.....	16	Play();	17
IE Filter für HTML-Elemente ab IE 5.5.....	15	PopUp-Fenster ab IE 5.5.....	38
IE Filter kodieren	15	Pulldown-Auswahlfeld mit farbigen Einträgen	13
IE Grafik zoomen ab IE 5.5	20	rechte Maus	12
IE Kodierung von Filtern.....	15	Refresh.....	38
IE Layertechnik.....	18	relative Pfade.....	6
IE PopUp-Fenster ab IE 5.5.....	38	Request Time Out	12
IE Shadow-Filter für HTML-Grafik-Elemente ab IE 5.5	20	robots.txt als Zeilenfolge-Script erstellen	35
INPUT-Element	16	robots.txt die nicht auf dem Server vorhanden ist	34
Internal Server Error	12	robots.txt kann entfallen.....	36
Internet Explorer	12	robots.txt und ab Zeile 2: Alle Daten auf dem Server dürfen gescannt werden	35
Internet Explorer Autoausfüllen	12	robots.txt und ab Zeile 2: Alle Daten auf dem Server dürfen NICHT gescannt werden.....	35
Internet Explorer Cursor für einen Link ändern.....	13	robots.txt und ab Zeile 2: Bestimmte Daten auf dem Server dürfen NICHT gescannt werden	35
Internet Explorer Formular mit Pulldown-Auswahlfeld mit farbigen Einträgen	13	robots.txt und ab Zeile 2: Bestimmte HTML-Dateien auf dem Server dürfen NICHT gescannt werden.....	36
Internet Explorer Linklayout verändern	13	robots.txt und ab Zeile 2: Bestimmte Verzeichnisse auf dem Server dürfen NICHT gescannt werden.....	35
Internet Explorer Link-Layout verändern wenn Maus über Link fährt	13	robots.txt und ab Zeile 2: NUR bestimmte Daten auf dem Server dürfen gescannt werden	36
Internet Explorer Linkunterstreichung abschalten	13	robots.txt und Aufbau.....	35
Internet Explorer Pulldown-Auswahlfeld mit farbigen Einträgen	13	robots.txt und Beispiele.....	36
Internet Explorer Standardlinkfarben ändern	13	robots.txt und ihre Blöcke	35
Internet Explorer Typen von Links-Layouts per CSS definieren	13	robots.txt und ihre Lage auf dem Server.....	34
Internet-Provider.....	6	robots.txt und Suchmaschinen	34
JobTitle	12	robots.txt und Zeile 1: ALLE Suchmaschinen dürfen scannen.....	35
Keine Abstände zwischen Tabellen.....	21	robots.txt und Zeile 1: Eine bestimmte Suchmaschinen darf NICHT scannen.....	35
KEINE Suchmaschine darf scannen.....	35	robots.txt und Zeile 1: Eine bestimmte Suchmaschinen darf scannen	35
Kodierung von Filtern.....	15		
Kodierung von Quelltext	3		
Kontextmenü für rechte Maus sperren	12		
Laden einer großen Tabelle	21		
Laden eines fremden Dokumentes ohne Framedarstellung	31		



robots.txt und Zeile 1: KEINE Suchmaschine darf scannen.....	35	transparentes Vordergrund-Bild auf dem Hintergrundbild	16
robots.txt-Block: Aufbau ab Zeile 2	35	Typen von Links-Layouts per CSS definieren	13
robots.txt-Block: Aufbau Zeile 1	35	Überlagerung zweier Bilder	17
Rollbalken	14	Umleitung auf andere Url	31
Root der Webseite	6	Unauthorized	12
Scan-Standard für Suchmaschinen	34	Unterverzeichnisse	35
Scanvorgänge von Suchmaschinen	34	Url	31
Schachbrettmuster	21	user-agent: *	35
Schatten eines Bildes	20	user-agent: WebCrawler	35
Schattenbreite	20	VCARDNAME=	12
Schattengrafik	22	verschachtelte Tabellen	15
Schattenrichtung	20	Verschachtelte Tabellen	21
Schnelleres Laden auch bei abgeschalteter Grafikanzeige	15	vertikale Linien zwischen Texten	30
Scrollbalken	14	verweis-sensitive Grafik	14
Seitenlayout	15	Viereck-Grafik	14
Server	34	Viereck-Grafik zerlegen in kleine Elemente und als blinde Tabelle zusammenbauen	14
Server Error-Meldung	39	Viereck-Grafik zerlegen in kleine Elemente, als blinde Tabelle zusammenbauen und jedem Teil einen eigenen Link zuweisen	14
Server Fehlermeldung	39	virtueller Host	6
Server-Error und eigene Fehlermeldung	39	visibility	17
Service Unavailable	12	visited	13
shadow()	20	Vordergrund-Bild auf dem Hintergrundbild	16
Shadow-Filter für HTML-Grafik-Elemente ab IE 5.5	20	Vorname	12
Slide-Show-Effekt	18	Web auf dem Server	34
Speicherung als Datei	39	WebCam	38
Stadt Firma	12	WebCrawler darf ALLES außer /test/ scannen	36
Stadt privat	12	WebCrawler darf ALLES außer /test/test.htm scannen	36
Standardlinkfarben ändern	13	WebCrawler darf ALLES scannen	36
Strasse Firma	12	WebCrawler darf NICHTS scannen	36
Straße privat	12	Web-Projekt	36
Streaming Audio	39	Webseite Festplattenordner	6
Streaming-Audio-Format	39	Webseite Host	6
strength	20	Webseite per Apache-HTTP-Server	7
style.visibility	17	Webseite Root	6
Suchmaschine	33	window.location.hostname	6
Suchmaschinen	34	Wortlogo zentriert im Fenster anzeigen	30
Suchmaschinen Scan-Standard	34	Zellbreite	22
Suchmaschinen zulassen bzw. sperren	35	Zelle	22
Suchmaschinen, die scannen dürfen	35	Zelle mit exakten Dimensionen bei älteren Browsern	22
Tabelle	14, 15	Zelle mit Schattengrafik passend zum Hintergrund	22
Tabellen	21	zentriert im Browserfenster	30
table-layout:fixed	21	zentriert im Browserfenster anzeigen	30
Telefon Firma	12	Zerlegen einer großen Tabelle in kleinere Tabellen	21
Telefon privat	12	Zoom	20
Text	30	Zugriffsschutz FTP	36
Text innerhalb einer Tabellenzelle	22	Zugriffsschutz Passwort	37
text-decoration:none	13	Zulassen bzw. Sperren von Inhalten auf dem Server für Suchmaschinen, die scannen	35
Textlage in der Tabellenzelle	22	Zulassen bzw. Sperren von Suchmaschinen	35
TOPMARGIN	12, 21		
transparent	16		
transparente Überlagerung zweier Bilder	17		
transparentes INPUT-Element	16		

