




JavaScript

Grundlagen


Grundlagen

SELFHTML: Version 8.1.1 vom 24.11.2005, Autor: Stefan Münz



Die Energie des Verstehens
HTML-Dateien selbst erstellen

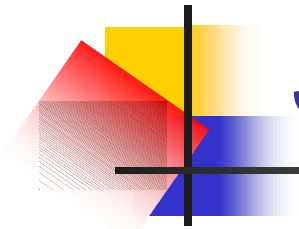
SELFHTML



[News](#)
Online-News
rund um
SELFHTML

Suche nach:

Inhalt: Allgemeines	Navigation: Einstieg
<ul style="list-style-type: none">EditorialEinführung	<ul style="list-style-type: none">Wie fange ich an?Häufig gestellte Fragen
Inhalt: Web-Technologien	Navigation: Kurzreferenzen
<ul style="list-style-type: none">HTML/XHTMLStylesheets (CSS)XML/DIUsJavaScript/DOMDynamisches HTMLPerlPHP	<ul style="list-style-type: none">Kurzreferenz: HTMLKurzreferenz: CSS
Inhalt: Ergänzendes Wissen	Navigation: Verzeichnisse
<ul style="list-style-type: none">InternationalisierungGrafikWeb-ProjektverwaltungWebserver/CGIDiverse technische Ergänzungen	<ul style="list-style-type: none">InhaltsverzeichnisSyntaxverzeichnisStichwortverzeichnis
	Navigation: Extras
	<ul style="list-style-type: none">QuickbarSidebarsSuche

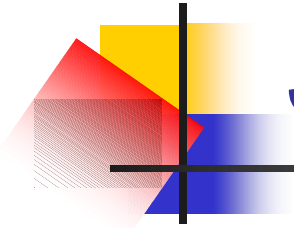


JavaScript in HTML verwenden

- JavaScript einbinden
 - Direktes Einbinden

```
<script type="text/javascript">  
<!--  
    JavaScript-Anweisungen  
//-->  
</script>
```

- Kann sowohl im Kopf (<head>) als auch im Rumpf (<body>) eingefügt werden.
- Beim Laden eines HTML-Dokuments werden die JavaScript-Anweisungen sequentiell abgearbeitet.



JavaScript in HTML verwenden

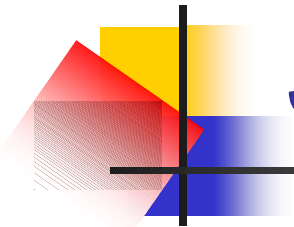
- JavaScript einbinden
 - Meistens werden JavaScripts im <head>-Tag eingefügt für
 - oft verwendete Funktionen
 - globale Variablen
 - JavaScript im <body>-Tag für
 - Ausgabe in das HTML-Dokument
 - um auf ein Ereignis zu reagieren (z.B. Klick auf Button)



JavaScript in HTML verwenden

- Scriptsprache angeben
 - über das Attribut language
 - Bsp: JavaScript Versionen.html
 - Die Methode `document.write()` gibt Text im HTML-Dokument aus

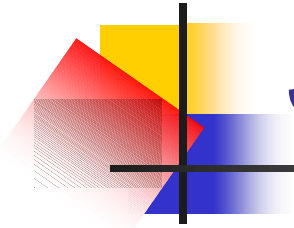
```
<html>
  <head>
    <title>JavaScript</title>
  </head>
  <body bgcolor="#FFFFFF" text="#000000" link="#0000FF" vlink="#800080" alink="#FF0000">
    <script language="javascript">
      window.document.write("Der Browser unterstützt JavaScript<HR>");
    </script>
    <script language="javascript1.1">
      window.document.write("Der Browser unterstützt JavaScript v1.1<HR>");
    </script>
    <script language="javascript1.2">
      window.document.write("Der Browser unterstützt JavaScript v1.2<HR>");
    </script>
```



JavaScript in HTML verwenden

- JavaScript für nicht scriptfähige Browser
 - Benutzerfreundliche Textausgabe bzw. HTML-Code falls JavaScript nicht aktiviert wurde
 - Bsp: `KeinJavaScript.html`

```
<script language="javascript">
<!--
    window.document.write("Der Browser unterstützt JavaScript<HR>");
//-->
</script>
<noscript>
    Ihr Browser unterstützt nicht JavaScript, oder es ist deaktiviert!
</noscript>
```

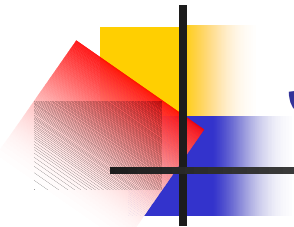


JavaScript in HTML verwenden

- JavaScript einbinden
 - durch externe Dateien

```
<script type="text/javascript" src="URL"></script>
```

- Es können relative oder absolute Pfadangaben verwendet werden.
- JavaScript-Dateien besitzen die Endung *.js
- Bsp: Externe JavaScript Datei.html



JavaScript in HTML verwenden

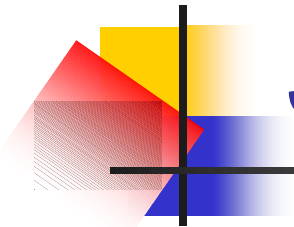
- JavaScript einbinden
 - durch Verwendung in Funktionen und HTML-Tags
 - Zur Behandlung von Ereignissen und bei der Verwendung in HTML-Tags kann JavaScript-Code auch ohne `<script>`-Tag angegeben werden

```
<button type="button" onclick="document.write('Sie haben mich gedr&uuml;ckt');">
    Klicken Sie auf mich
</button>
```

- oder

```
<a href="javascript:window.print()">Diese Seite drucken</a>
```

- Bsp: JavaScript Aufruf.html



JavaScript in HTML verwenden

- JavaScript Ausführungsreihenfolge
 - Ausführung in der Reihenfolge, in der der Code im HTML-Dokument vorkommt (z.B. Kopf- vor Rumpfbereich).
 - Ausführung von Code (z.B. beim Klick auf einen Link Ausgaben im Dokument durchführen) überschreibt den Inhalt des alten Dokuments.
- Bsp: Ausführungsreihenfolge.html
 - Beachte zu ``
Da die Textausgabe in das HTML-Dokument nach dem Ladevorgang erfolgt, wird der bisherige Inhalt überschrieben (Anzeige Quelltext im Browser).

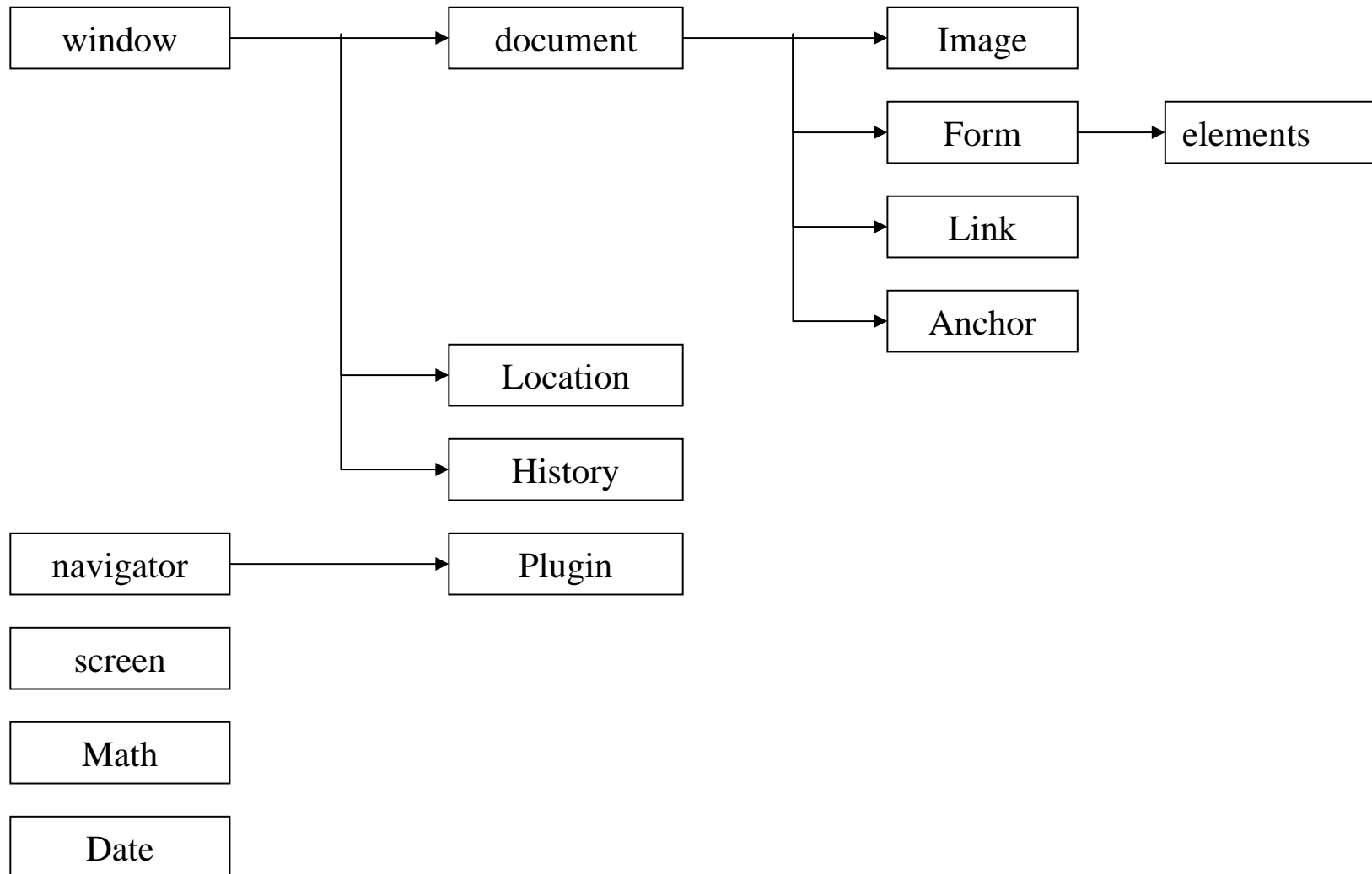


JavaScript in HTML verwenden

- JavaScript Kommentare
 - `//`
 - Zeilenweiser Kommentar
 - `/* ... */`
 - Mehrzeiliger Kommentar

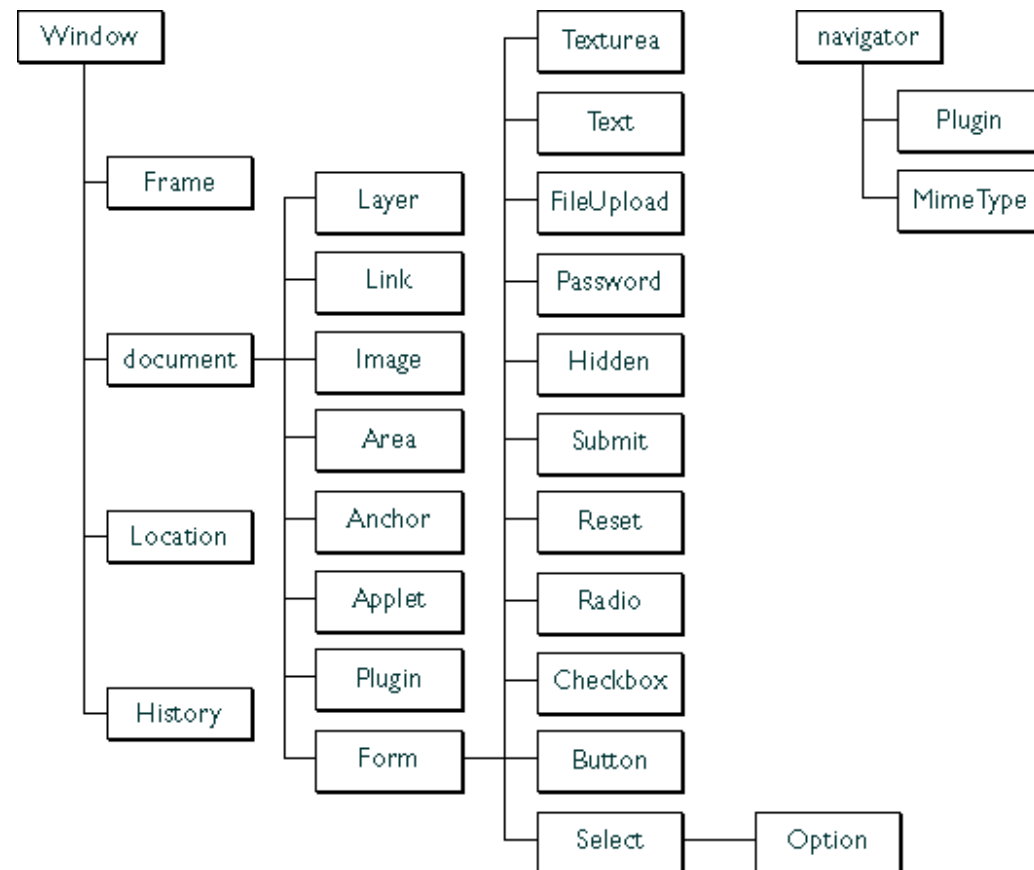
Oberste
Hierarchieebene

Niedrigere
Hierarchieebene





Objekthierarchie





Objekt: window

Eigenschaften:

- ↓ [closed](#) (geschlossenes Fenster)
- ↓ [defaultStatus](#) (Normalanzeige in der Statuszeile)
- ↓ [innerHeight](#) (Höhe des Anzeigebereichs)
- ↓ [innerWidth](#) (Breite des Anzeigebereichs)
- ↓ [locationbar](#) (Adresszeile)
- ↓ [menubar](#) (Menüleiste)
- ↓ [name](#) (Fenstername)
- ↓ [outerHeight](#) (Höhe des gesamten Fensters)
- ↓ [outerWidth](#) (Breite des gesamten Fensters)
- ↓ [pageXOffset](#) (Fensterstartposition von links)
- ↓ [pageYOffset](#) (Fensterstartposition von oben)
- ↓ [personalbar](#) (Zeile für Lieblingsadressen)
- ↓ [scrollbars](#) (Scroll-Leisten)
- ↓ [statusbar](#) (Statuszeile)
- ↓ [status](#) (Inhalt der Statuszeile)
- ↓ [toolbar](#) (Werkzeugleiste)

Methoden:

- ↓ [alert\(\)](#) (Dialogfenster mit Infos)
- ↓ [back\(\)](#) (zurück in History)
- ↓ [blur\(\)](#) (Fenster verlassen)
- ↓ [captureEvents\(\)](#) (Ereignisse überwachen)
- ↓ [clearInterval\(\)](#) (zeitliche Anweisungsfolge abbrechen)
- ↓ [clearTimeout\(\)](#) (Timeout abbrechen)
- ↓ [close\(\)](#) (Fenster schließen)
- ↓ [confirm\(\)](#) (Dialogfenster zum Bestätigen)
- ↓ [disableExternalCapture\(\)](#) (Fremdüberwachung verhindern)
- ↓ [enableExternalCapture\(\)](#) (Fremdüberwachung erlauben)
- ↓ [find\(\)](#) (Text suchen)
- ↓ [focus\(\)](#) (Fenster aktiv machen)
- ↓ [forward\(\)](#) (vorwärts in History)
- ↓ [handleEvent\(\)](#) (Ereignis übergeben)
- ↓ [home\(\)](#) (Startseite aufrufen)
- ↓ [moveBy\(\)](#) (bewegen mit relativen Angaben)
- ↓ [moveTo\(\)](#) (bewegen mit absoluten Angaben)
- ↓ [open\(\)](#) (neues Fenster öffnen)
- ↓ [print\(\)](#) (ausdrucken)
- ↓ [prompt\(\)](#) (Dialogfenster für Werteingabe)
- ↓ [releaseEvents\(\)](#) (Ereignisse abschließen)
- ↓ [resizeBy\(\)](#) (Größe verändern mit relativen Angaben)
- ↓ [resizeTo\(\)](#) (Größe verändern mit absoluten Angaben)
- ↓ [routeEvent\(\)](#) (Event-Handler-Hierarchie durchlaufen)
- ↓ [scrollBy\(\)](#) (Scrollen um Anzahl Pixel)
- ↓ [scrollTo\(\)](#) (Scrollen zu Position)
- ↓ [setInterval\(\)](#) (zeitliche Anweisungsfolge setzen)
- ↓ [setTimeout\(\)](#) (Timeout setzen)
- ↓ [stop\(\)](#) (abbrechen)

Unterobjekte:

- ≡ [document](#)
- ≡ [event](#)
- ≡ [history](#)
- ≡ [location](#)



Objekt: document

Eigenschaften:

- ↓ [alinkColor](#) (Farbe für Verweise beim Anklicken)
- ↓ [bgColor](#) (Hintergrundfarbe)
- ↓ [charset](#) (verwendete Zeichenkodierung)
- ↓ [cookie](#) (beim Anwender speicherbare Zeichenkette)
- ↓ [defaultCharset](#) (voreingestellte Zeichenkodierung)
- ↓ [fgColor](#) (Farbe für Text)
- ↓ [lastModified](#) (letzte Änderung am Dokument)
- ↓ [linkColor](#) (Farbe für Verweise)
- ↓ [referrer](#) (zuvor besuchte Seite)
- ↓ [title](#) (Titel der Datei)
- ↓ [URL](#) (URL-Adresse der Datei)
- ↓ [vlinkColor](#) (Farbe für Verweise zu besuchten Zielen)

Methoden:

- ↓ [captureEvents\(\)](#) (Ereignisse überwachen)
- ↓ [close\(\)](#) (schließen)
- ↓ [createAttribute\(\)](#) (Attributknoten erzeugen)
- ↓ [createElement\(\)](#) (Elementknoten erzeugen)
- ↓ [createTextNode\(\)](#) (Textknoten erzeugen)
- ↓ [getElementById\(\)](#) (HTML-Elementzugriff über id-Attribut)
- ↓ [getElementsByName\(\)](#) (HTML-Elementzugriff über name-Attribut)
- ↓ [getElementsByTagName\(\)](#) (HTML-Elementzugriff über Elementliste)
- ↓ [getSelection\(\)](#) (selektierter Text)
- ↓ [handleEvent\(\)](#) (Ereignisse verarbeiten)
- ↓ [open\(\)](#) (Dokument öffnen)
- ↓ [releaseEvents\(\)](#) (Ereignisse abschließen)
- ↓ [routeEvent\(\)](#) (Event-Handler-Hierarchie durchlaufen)
- ↓ [write\(\)](#) (ins Dokumentfenster schreiben)
- ↓ [writeln\(\)](#) (zeilenweise schreiben)

Unterobjekte:

- ≡ [HTML-Elemente](#)
- ≡ [node](#)
- ≡ [all](#)
- ≡ [anchors](#)
- ≡ [applets](#)
- ≡ [forms](#)
- ≡ [images](#)
- ≡ [layers](#)
- ≡ [links](#)



Programmierung

- ≡ Allgemeine Regeln für JavaScript
- ≡ Variablen und Werte
- ≡ Objekte, Eigenschaften und Methoden
- ≡ Funktionen
- ≡ Steuerzeichen und besondere Notationen
- ≡ Operatoren
- ≡ Bedingte Anweisungen (if-else/switch)
- ≡ Schleifen (while/for/do-while)
- ≡ Reservierte Wörter
- ≡ Event-Handler

Siehe Programmierbeispiele beginnend mit `inkdek.htm`