

2 JavaScript

| | | |
|----------|--|----------|
| 2 | JavaScript | 1 |
| 2.1 | Die Objekte des Sprachkerns | 4 |
| 2.1.1 | „Top-Level“ | 4 |
| 2.1.2 | Array | 5 |
| 2.1.3 | Boolean | 7 |
| 2.1.4 | Date - Datum | 7 |
| 2.1.5 | Function | 9 |
| 2.1.6 | Math - mathematische Funktionen und Konstanten | 10 |
| 2.1.7 | Number | 10 |
| 2.1.8 | Object - Die Superklasse aller Objekte | 11 |
| 2.1.9 | RegExp - Reguläre Ausdrücke | 12 |
| 2.1.10 | String - Zeichenketten | 13 |
| 2.2 | Die Window-Objekte des Browsers | 16 |
| 2.2.1 | Window - Browserfenster oder Frame | 17 |
| 2.2.2 | Event - Einzelheiten zu Ereignissen | 22 |
| 2.2.3 | Frames | 24 |
| 2.2.4 | History - URL-Aufzeichnungspfad des Browsers | 24 |
| 2.2.5 | Location - die URL-Eingabezeile des Browsers | 25 |
| 2.2.6 | Navigator - Informationen über den Browser | 26 |
| 2.3 | document und darunter (außer forms) | 27 |
| 2.3.1 | HTMLElement | 27 |
| 2.3.2 | Anchor - Ziel von Hypertextlinks | 29 |
| 2.3.3 | Applet - Einbettung eines Java-Applets | 30 |
| 2.3.4 | Area - "Hotspot" in einer Bildkarte | 31 |
| 2.3.5 | Document - Das HTML-Dokument | 31 |
| 2.3.6 | Image | 34 |
| 2.3.7 | Link - Hypertext-Link | 35 |
| 2.4 | Formular und Formularobjekte (forms) | 36 |
| 2.4.1 | Input - Die Eigenschaften von Formularobjekten | 36 |

| | | |
|--------|--|----|
| 2.4.2 | Button - Taster | 38 |
| 2.4.3 | Checkbox - Kästchen zum Ankreuzen | 39 |
| 2.4.4 | Element siehe Input | 41 |
| 2.4.5 | FileUpload | 41 |
| 2.4.6 | Form | 42 |
| 2.4.7 | Hidden | 43 |
| 2.4.8 | Option | 44 |
| 2.4.9 | Password | 45 |
| 2.4.10 | Radio | 46 |
| 2.4.11 | Reset | 47 |
| 2.4.12 | Select | 49 |
| 2.4.13 | Submit | 50 |
| 2.4.14 | Text | 51 |
| 2.4.15 | Textarea - Mehrzeiliges Editierfenster | 52 |
| 2.5 | Verschiedene | 53 |
| 2.5.1 | Layer - Dynamisch positionierbare Elemente | 53 |
| 2.5.2 | Screen - Informationen über den Bildschirm | 57 |
| 2.5.3 | Style | 58 |

Grundlagen:

Netscape: Client-Side JavaScript Reference.
<http://developer.netscape.com/docs/manuals/javascript.html>

Flanagan, David: JavaScript. The Definitive Guide.
O'Reilly 1998, 2. Auflage, ISBN 1-56592-392-8

Flanagan, David: JavaScript - kurz&gut.
O'Reilly 1998, ISBN 3-89721-208-0

Microsoft: Dynamic HTML. Reference and Software Development Kit.
Microsoft Press 1999, ISBN 0-7356-0638-2

Microsoft über die Sitemap
<http://msdn.microsoft.com/siteguide/sitemp.asp>

Vorbemerkungen

- Die folgende Sammlung von Browser/HTML-Objekten soll einen schnellen Überblick über die meisten Objekte und deren Eigenschaften und Methoden geben. Sie beschreibt nicht alle verfügbaren Merkmale, ist also keine Referenz.
- Bei den Eigenschaften, Methoden, Konstruktoren und Eventhandlern in den Objektbeschreibungen ist in der Regel jeweils die Javascript-Version angegeben, mit der diese eingeführt wurden, z.B.

Number (objekt) 1.2
 Konvertiert das angegebene Objekt in eine Zahl (wie `new Number(objekt)`), siehe unter `Number`).

Bei Sonderwegen von Netscape oder Microsoft ist entsprechend die Version dieser Produkte vermerkt, zusammen mit einem Kürzel, das das Produkt kennzeichnet, z.B.

shift() NN4
 Entfernt das *erste* Element aus dem Array und gibt es zurück.

Infinity IE4, 1.3
 Eine numerische Konstante, die den Wert Unendlich repräsentiert.

Die Version 1.0 von JavaScript wird oft nicht gesondert vermerkt.

- Bei der Angabe zweier Versionen, z.B. 1.0, 1.2, kennzeichnet die erste die Einführung und die zweite die Modifikation der betroffenen Methoden, Eventhandler, Konstruktoren bzw. Eigenschaften. Beispiel:

setDay(), setUTCDay() 1.0, 1.2
 Setzt den Tag innerhalb einer Woche (0..6; Ortszeit, UTC).

- Eigenschaften, Methoden etc., die besondere Privilegien erfordern (z.B. bei Netscape „`UniversalPreferencesRead`“ oder „`UniversalBrowserWrite`“), sind in der Regel nicht aufgeführt.
- Die Darstellung ist etwas Netscape-lastig, einfach deshalb, weil Netscape der „Erfinder“ von JavaScript ist. In Form z.B. des `HTMLElement`-Objektes haben aber auch der Internet Explorer und DOM deutliche Duftmarken hinterlassen.

Die Objektbeschreibungen möchte ich nach und nach weiter verfeinern und mit mehr Prosa versehen, sie vielleicht auch mit Beispielen und Bildern versehen, und sie vor allem einigermaßen aktuell halten.

Hohenheim, 3. Dezember 2000

Wolfgang Dehnhardt

2.9 Die Objekte des Sprachkerns

Jede der Objektbeschreibungen beginnt mit einem Kästchen, in dem kurz die Zugriffssyntax angegeben ist und alle Eigenschafts-, Methoden-, Konstruktor- und Ereignisnamen aufgezählt sind. Im Anschluss daran sind letztere jeweils kurz charakterisiert. Dazu gehört auch eine Notiz, ab welcher Version es sie gibt und ob sie spezialisiert sind auf einen der beiden Browser, dem Navigator (NN_x) oder dem Internet Explorer (IE_x).

Beachten Sie bitte, dass exotische Eigenschaften und Funktionen gelegentlich weggelassen sind, vor allem dann, wenn diese besondere Zugriffsprivilegien benötigen.

2.9.1 „Top-Level“

| | | |
|--|---|------------------------------|
| "Top-Level" | Keine Bindung an ein Objekt! | |
| Eigenschaften Infinity, NaN, undefined | Methoden escape(), eval(), isFinite(), isNaN(), Number(), parseFloat(), parseInt(), String(), unescape() | Konstruktoeren --- |

Unter „Top-Level“ sind alle Eigenschaften und Methoden gesammelt, die nicht an bestimmte Objekte gebunden, d.h. gültig für alle Objekte sind (globale Eigenschaften und Funktionen).

Infinity 1.3, IE4

Eine numerische Konstante, die den Wert Unendlich repräsentiert.

NaN 1.3, IE4

Konstante, die „Not-a-Number“, d.h. „keine Zahl“ symbolisiert.

Number (objekt) 1.2

Konvertiert das angegebene Objekt in eine Zahl (wie `new Number(objekt)`, siehe unter `Number`).

undefined 1.3

Der Wert „undefiniert“.

escape (string)

Konvertiert alle ASCII-Zeichen, die weder Buchstaben noch Zahlzeichen sind. Sie werden ihrem Hexadezimalwert `xx` entsprechend in den String `%xx` gewandelt und als Funktionswert zurückgegeben. Das Kodierungsverfahren hat ihr Vorbild in URLs und wird in JavaScript für *Cookies* verwendet (einziger Unterschied: Leerzeichen ist `+` bei URLs und `%20` bei `escape()`).

eval (codestring)

Führe den String in der Parameterklammer als JavaScript-Programm aus.

isFinite (zahl) 1.2

true, wenn die Zahl zahl endlich ist, sonst false.

isNaN (wert) 1.1

true, wenn wert keine Zahl ist, sonst false.

parseFloat (string)

String in eine *Fließkommazahl* konvertieren. Die Konvertierung wird mit Stringende oder beim ersten nicht zur Zahl gehörenden Zeichen abgeschlossen. (NaN, wenn der String nicht mit einem gültigen Zeichen beginnt.)

parseInt (string, basis)

String in eine *ganze Zahl* konvertieren. Die Konvertierung wird mit Stringende oder beim ersten nicht zur Zahl gehörenden Zeichen abgeschlossen. Der optionale Parameter basis ist die Zahlenbasis der als String kodierten Zahl. (NaN, wenn der String string nicht mit einem gültigen Zeichen beginnt.)

String (string)

wie new String(string), siehe String-Objekt

unescape (string)

Die Umkehrung von escape(string), d.h %20 wird wieder zum Leerzeichen und %25 zum Ausrufungszeichen.

2.9.2 Array

| Array | | 1.1 und höher |
|----------------------|--|---|
| Eigenschaften | Methoden | Konstruktoren |
| length | concat(), join(), pop(), push(), reverse(), shift(), slice(), sort(), splice(), toString(), unshift() | new Array() new Array(groesse) new Array(elem1, elem2, ...) |

Konstruktoren

new Array()

new Array(groesse)

new Array(element1, element2, ...) 1.1

Mit new Array() wird ein Array erzeugt. Optional kann dabei entweder seine Länge festgelegt oder es mit einer Elementenliste initialisiert werden.

Eigenschaften

length 1.1
 Enthält die Größe eines Arrays, die durch das Arrayelement mit dem größten Index festgelegt ist.

Methoden

concat(werteliste) 1.2
 Fügt dem Array die als Parameter übergebene Werteliste am Ende an.

join(trenner) 1.1
 Fügt die Elemente eines Arrays zu einem String, wobei die optionale Variable `trenner` als Trennzeichen (oder Trennstring) zwischen die Elemente geschoben wird, und gibt ihn zurück.

pop() NN4
 Entfernt das *letzte* Element aus dem Array und gibt es zurück.

push(werteliste) NN4
 Fügt einem Array die Werte der Liste `werteliste` als neue Elemente am Ende an. Zurückgegeben wird der letzte angehängte Wert.

reverse() 1.1
 Kehrt die Reihenfolge der Elemente eines Arrays um.

shift() NN4
 Entfernt das *erste* Element aus dem Array und gibt es zurück.

slice(startindex, stopindex) 1.2
 Gibt einen zusammenhängenden Teil des Arrays zurück, und zwar beginnend mit dem Index `startindex` und endend *eins vor* dem `stopindex`. Ist einer der Indizes negativ, so wird vom Ende des Array her gezählt.

sort(sortierfunktion) 1.1
Ohne Parameter sortiert die Funktion das Array in alphabetischer Reihenfolge. Bei Angabe einer Sortierfunktion, in der die Beziehungen zwischen jeweils zwei Arrayelementen festgelegt werden, bestimmt diese die Reihenfolge der Elemente.

splice(startindex, anzahl, werteliste) NN4
 Löscht `anzahl` Elemente ab dem Index `startindex` aus dem Array und fügt an deren Stelle optional die Elemente der `werteliste` ein.

toString() 1.1
 Konvertiert ein Array zu einem zusammenhängenden String (entspricht `join()`) und gibt es zurück.

unshift(werteliste) NN4
 Fügt einem Array ein oder mehrere Werte `wert` als neue Elemente am Anfang hinzu. Zurückgegeben wird die Größe des neuen Arrays.

2.9.3 Boolean

| Boolean | | |
|----------------------|-------------------------------|--|
| Eigenschaften | Methoden toString() | Konstruktoren new Boolean(<i>wert</i>) Boolean(<i>wert</i>) |

new Boolean(*wert*)

Erzeugt ein Boolean-Objekt entsprechend dem Wert von *wert*. Es hat den Wert `false` für 0, null, NaN und "", sonst ist der Wert `true`.

Boolean(*wert*)

Konvertiert den Wert *wert* in einen primitiven Booleschen Typ und gibt ihn zurück.

toString()

Resultat dieser Funktion ist entweder "true" oder "false".

2.9.4 Date - Datum

| Date | | |
|---|---|--|
| Methoden get/setDate(), get/setDay(), get/setFullYear(), getHours(), get/setMilliseconds(), get/setMinutes(), get/setMonth(), getSeconds(), get/setTime(), setTimezoneOffset(), get/setUTCDate(), get/setUTCDay(), get/setUTCFullYear(), | Methoden get/setUTCHours(), get/setUTCMilliseconds(), get/setUTCMinutes(), get/setUTCMonth(), get/setUTCSeconds(), toGMTString(), toLocaleString(), toString(), toUTCString(), valueOf() Statische Methoden: Date.parse(), Date.UTC() | Konstruktoren new Date() new Date(milliSekunden) new Date(datumstring) new Date(jahr, monat, tag, stunde, minute, sekunde, millisekunden) |

Konstruktoren

new Date()

Objekt mit aktuellem Datum

new Date(millisekunden)

Millisekunden seit dem 1. Januar 1970 0:00 Uhr UTC (Universal Time Coordinated).

new Date(datumzeitstring)

z.B. Date("June 28, 2000 08:30:00").

new Date(jahr, monat, tag, stunde, minute, sekunde, ms)

Datumsobjekt mit Datum und Uhrzeit wie angegeben.

Methoden

Date.parse(datumzeitstring)

Gibt die seit dem 01.01.70 0:00Uhr UTC verstrichene Zeit in Millisekunden zurück.

Date.UTC(jahr, monat, tag, stunde, minute, sekunde, ms)

Seit dem 01.01.70 0:00Uhr UTC verfllossene Millisekunden.

getDate(), getUTCDate() 1.0, 1.2

Liefert den Tag innerhalb des Monats (1...31; Ortszeit, UTC).

getDay(), getUTCDay() 1.0, 1.2

Liefert den Tag innerhalb einer Woche (0...6; Ortszeit, UTC).

getFullYear(), getUTCFullYear() 1.2

Liefert das Jahr zurück (Ortszeit, UTC).

getHours(), getUTCHours() 1.0, 1.2

Gibt den Stundenteil des Date-Objektes zurück (Ortszeit, UTC).

getMilliseconds(), getUTCMilliseconds() 1.2

Gibt die Millisekundenzahl des Date-Objektes zurück (Ortszeit, UTC).

getMinutes(), getUTCMinutes() 1.0, 1.2

Der Minutenanteil des Objektes wird zurückgegeben (Ortszeit, UTC).

getMonth(), getUTCMonth() 1.0, 1.2

Der Monat des Date-Objektes (Ortszeit, UTC).

setDate(), setUTCDate() 1.0, 1.2

Setzt den Tag innerhalb des Monats (1...31; Ortszeit, UTC).

setDay(), setUTCDay() 1.0, 1.2

Setzt den Tag innerhalb einer Woche (0...6; Ortszeit, UTC).

setFullYear(), setUTCFullYear() 1.2

Setzt das Jahrfeld im Objekt (Ortszeit, UTC).

setHours(), setUTCHours() 1.0, 1.2

Setzt den Stundenteil des Date-Objektes (Ortszeit, UTC).

setMilliseconds(), setUTCMilliseconds() 1.2

Setzt die Millisekundenzahl des Date-Objektes (Ortszeit, UTC).

setMinutes(), setUTCMinutes() 1.0, 1.2

Der Minutenanteil des Objektes wird zurückgegeben (Ortszeit, UTC).

setMonth(), setUTCMonth() 1.0, 1.2

Der Monat des Date-Objektes (Ortszeit, UTC).

toLocaleString(), toGMTString()/toUTCString() 1.0, 1.0/1.2

Datumsobjekt in einen String wandeln (Ortszeit, UTC).

toString()

Datumsobjekt in einen String wandeln.

valueOf () 1.1
 Datumsobjekt in Millisekunden-Zeit wandeln (wie `getTime ()`).

2.9.5 Function

| Function | | |
|--|--|---|
| Eigenschaften arguments[], caller, length, prototype NN: arity | Methoden toString() NN: apply() | Konstruktoren new Function(<i>arg1</i> , <i>arg2</i> , <i>arg3</i> ,... <i>funktionskoerper</i>) |

Konstruktor

new Function(*arg1*, *arg2*, *arg3*, ... *funktionskoerper*) 1.1
 Funktionskonstruktor

Eigenschaften

- arguments []**
Array mit den der Funktion übergebenen Parametern.
- arity** NN4
Die Anzahl der definierten Parameter.
- caller**
Referenz zu der Funktion, die diese aufgerufen hat. (null, falls vom „Top-Level“ aufgerufen.)
- length** 1.1
Die Anzahl der definierten Parameter.
- prototype** 1.1
Das Prototypobjekt, das von allen Objekten, die mittels des `Function`-Konstruktors erzeugt wurden, geerbt wird.

Methoden

- apply(*objekt*, *arg1*, *arg2*, ...)** NN4
Mit `funktion.apply(objekt, ...)` wird die Funktion `funktion` so aufgerufen, als ob sie eine Methode des Objektes `objekt` wäre.
- toString ()**
gibt den Code des Funktionskörpers als String zurück.

2.9.6 Math - mathematische Funktionen und Konstanten

| | | |
|--|---|-----------------------------|
| Math | Math. <i>konstantenName</i> , Math. <i>methodenName</i> () | |
| Konstanten E, LN10, LN2, LOG10E, LOG2E, PI, SQRT1_2, SQRT2 | Statische Methoden abs(), acos(), asin(), atan(), atan2(), ceil(), cos(), exp(), floor(), log(), max(), min(), pow(), random(), round(), sin(), sqrt(), tan() | Konstruktoren --- |

Allen Konstanten und Methoden *muss* Math vorangestellt werden, z.B. Math.random(), Math.PI, Math.sin(1.3) etc.

E, LN10, LN2, LOG10E, LOG2E, PI, SQRT1_2, SQRT2

Die Konstanten e (Euler-Zahl, 2.72.7182818...), $\log_e(10)$, $\log_{10}(e)$, $\log_2(e)$, π , $\sqrt{1/2}$, $\sqrt{2}$.

sin(x), cos(x), tan(x), asin(x), acos(x), atan(x), atan2(x,y)

Trigonometrische Funktionen und ihre Umkehrungen; atan2(x, y) stimmt überein mit atan(x/y).

exp(x), log(x)

Exponential- und Logarithmusfunktion.

abs(x), round(x), ceil(x), floor(x)

Absolutbetrag bilden, gaußsche Rundung (ab .5 aufwärts, sonst abwärts), nächst kleinere bzw. nächst größere ganze Zahl.

min(x, y), max(x, y)

Die kleinste bzw. die größte von zwei Zahlen bestimmen.

sqrt(x), pow(x, y)

Quadratwurzel ziehen und Potenz aus zwei Zahlen bilden (x^y)

random()

Eine Zufallszahl zwischen 0.0 und 1.0 berechnen.

2.9.7 Number

| | | |
|---|-------------------------------|--|
| Number | Number. <i>konstantenName</i> | 1.1 und höher |
| Konstanten MAX_VALUE, MIN_VALUE, NaN, NEGATIVE_INFINITY, POSITIVE_INFINITY | Methoden toString() | Konstruktoren new Number(<i>wert</i>) Number(<i>wert</i>) |

Konstruktor

- new Number (wert)** 1.1
Gibt ein *Zahlenobjekt* zurück (Typ `object`).
- Number (wert)** 1.1
Gibt einen *gewöhnlichen* numerischen Wert zurück (Typ `number`).

Konstanten

- MAX_VALUE, MIN_VALUE** 1.1
Die größte und die zu 0 hin kleinste darstellbare Zahl.
- NaN** 1.1
„not-a-number“, Symbol dafür, dass z.B. eine Variable keine Zahl beinhaltet. Beispielsweise hat der Ausdruck `0/0` den Wert `NaN`.
- NEGATIVE_INFINITY, POSITIVE_INFINITY** 1.1
Ein Wert, der negative oder positive Unendlichkeit symbolisiert. Beim Ausdrucken wird `-Infinity` oder `Infinity` ausgegeben.

Methoden

- toString (zahlenbasis)** 1.1
Wandelt eine Zahl in einen String um. Der optionale Parameter `zahlenbasis` kann einen Wert von 2 bis 36 haben, 2 für duale Codierung, 8 für oktale, 10 für dezimale (Standardwert) etc.

2.9.8 Object - Die Superklasse aller Objekte

| Object | | |
|----------------------|--|---|
| Eigenschaften | Methoden | Konstruktoren |
| constructor | assign(), eval(), toString(), unwatch(), valueOf(), watch() | new Object() new Object(<i>wert</i>) |

Konstruktoren

- new Object ()**
Erzeugt ein Objekt ohne Eigenschaften.
- new Object (wert)** 1.1
Erzeugt eine Zahlenobjekt (Typ `number`, `string` oder `boolean`) entsprechend dem Typ von `wert`.

Methoden

assign(wert)

Den Zuweisungsoperator überladen: `obj = wert <-> obj.assign(wert)`

eval(code)

Den String `code` als JavaScript-Programm ausführen.

toString()

Einen String erzeugen, der das Objekt repräsentiert.

watch(propname, handler)

NN4

unwatch(propname)

Mit `watch()` einen Handler `handler` definieren, der dann aufgerufen wird, wenn in die Eigenschaft `propname` gespeichert wird. (`unwatch()` zum Beenden.)

valueOf()

1.1

Gibt einen dem Objekt entsprechenden Wert von elementarem Typ zurück.

2.9.9 RegExp - Reguläre Ausdrücke

| RegExp | | 1.2 und höher |
|---|---------------------------------|---|
| Eigenschaften | Methoden | Konstruktoren |
| source NN: global, ignoreCase, lastIndex | compile(), test() NN: exec() | new RegExp(<i>muster</i> , <i>attribute</i>) |
| Statische Eigenschaften | | |
| \$n NN: input, \$_, lastIndex, lastMatch, lastParen, leftContext, multiline, rightContext | | |

Siehe ab Seite 88 des Buches.

2.9.10 String - Zeichenketten

| String | | |
|----------------------|---|---------------------------|
| Eigenschaften | Methoden | Konstruktoren |
| length | anchor(), big(), blink(), bold(), charAt(), charCodeAt(), concat(), fixed(), fontcolor(), fontsize(), indexOf(), italics(), lastIndexOf(), link(), match(), replace(), search(), slice(), small(), split(), strike(), small(), split(), strike(), sub(), substring(), toLowerCase(), sup(), toUpperCase() Statische Methode fromCharCode() | new String(<i>wert</i>) |

Konstruktor

new String(*wert*) 1.1
Erzeugt ein neuere Stringobjekt.

Eigenschaften

length 1.0
Die Anzahl der Zeichen in einem String (`string.length`).

Methoden

String.fromCharCode(*zeichencodeweiste*) 1.2
zeichencodeweiste ist eine kommagetrennte Liste von unicodierten Zeichen. Zurückgegeben wird ein String, der aus den entsprechenden Zeichen besteht.

charAt(*position*) 1.0
Mit diese Funktion kann jedes Zeichen eines Strings individuell angesprochen werden. Das erste Zeichen hat den Index 0, und das letzte `string.length-1`. Rückgabewert ist das bei *position* positionierte Zeichen.

charCodeAt(*position*) 1.2
Gibt statt des Zeichen in der Position *position* dessen Unicode zurück (soweit vom Browser unterstützt).

concat(*werteweiste*) 1.2
`string.concat()` heftet die Werte der kommagetrennten Liste nacheinander an den String *string*, nachdem diese erforderlichenfalls zuvor in Strings gewandelt wurden.

- indexOf(substring)** 1.0
indexOf(substring, start)
Sucht in einem vorgegebenen String den Teilstring `substring` und gibt die Position des ersten übereinstimmenden Zeichens oder `-1` (falls keine Übereinstimmung besteht) zurück. Mittels des optionalen Positionsparameters `start` kann die Suche auf die Zeichen danach beschränkt werden. Die Suche endet nach der *ersten* Übereinstimmung.
- lastIndexOf(substring)** 1.0
lastIndexOf(substring, start)
`lastIndexOf()` funktioniert wie `indexOf()`, allerdings bestimmt die *letzte* Übereinstimmung von `substring` die Position des ersten übereinstimmenden Zeichens.
- match(regex)** 1.2
`match()` sucht einen String nach Übereinstimmung mit dem Muster `regex` ab. Die Ergebnisse werden in einem Array zurückgegeben. Bestehen keine Übereinstimmungen, so wird `null` zurückgegeben.
- replace(regex, ersetzung)** 1.2
`regex` bestimmt die Muster im String, die ersetzt werden sollen, und `ersetzung` ist der String, der an die Stelle der übereinstimmenden Muster gesetzt werden soll.
- search(regex)** 1.2
`search()` sucht einen String nach Übereinstimmung mit dem Muster `regex` ab. Zurückgegeben wird die erste Position der Übereinstimmung oder `-1` bei Misserfolg.
- slice(von, bis)** 1.2
`slice()` hat die gleiche Funktion wie `substring()`, es können aber zusätzlich auch negative Parameterwerte verwendet werden. Solche Werte haben zur Folge, dass die Positionen `von` und `bis` vom *Ende* des Strings nach links gezählt werden.
- split(begrenzer)** 1.1
`split()` zerlegt einen String in Teilstrings, die in ein Array gespeichert werden. Der String `begrenzer` bestimmt die Stellen im String, an denen die Teilstrings voneinander getrennt werden. Wird `begrenzer` weggelassen, so besteht das zurückgegebene Array aus einem Element mit dem Originalstring, und enthält `begrenzer` einen leeren String, so enthalten die Arrayelemente jeweils ein Zeichen des Strings. Ab der Version 1.2 können für `begrenzer` auch reguläre Ausdrücke verwendet werden.
- substr(von, anzahl)** 1.2
Wie `substring()`, statt der Zeichennummer `bis` wird die Zahl `anzahl` der auszuscheidenden Zeichen angegeben (ab Version 1.2).
- substring(von, bis)** 1.0
Schneidet den Teil aus einem String, der bei Zeichennummer `von` beginnt und einen Zähler vor der Zeichennummer `bis` endet, und gibt ihn zurück.
- toLowerCase()** 1.0
Konvertiert alle alphabetische Zeichen in Kleinbuchstaben.

toUpperCase()

1.0

Konvertiert alle alphabetische Zeichen in Großbuchstaben.

Die folgende Gruppe von Methoden erzeugt HTML-Markenpaare:

```
string.anchor(name) -> <A Name="name">string</A>
string.big()        -> <BIG>string</BIG>
string.blink()     -> <BLINK>string</BLINK>
string.bold()      -> <B>string</B>
string.fixed()     -> <TT>string</TT>
string.fontcolor(col) -> <FONT Color="col">string</FONT>
string.fontSize(size) -> <FONT Color="size">string</FONT>
string.italics()   -> <I>string</I>
string.link(url)   -> <A Href="url">string</A>
string.small()     -> <SMALL>string</SMALL>
string.strike()    -> <STRIKE>string</STRIKE>
string.sub()       -> <SUB>string</SUB>
string.strike()    -> <STRIKE>string</STRIKE>
string.sup()       -> <SUP>string</SUP>
```

2.10 Die Window-Objekte des Browsers

Der nun folgenden Objektbeschreibung liegt die in Abb. 2-1 gezeigte hierarchische Beziehungsmodell für Browserobjekte zugrunde.

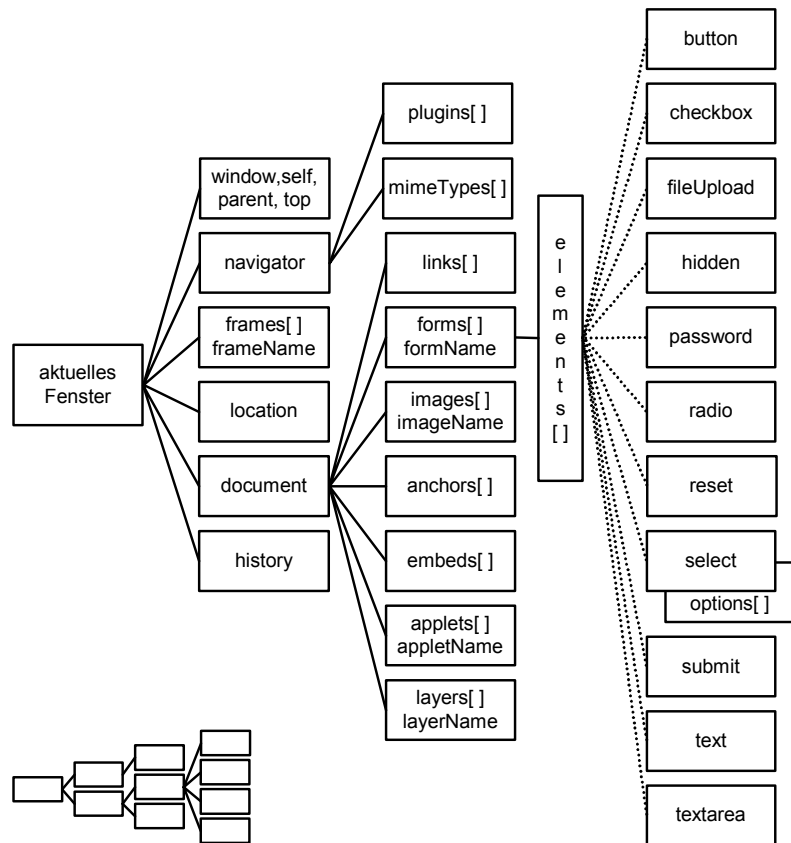
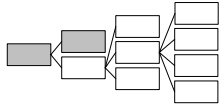


Abb. 2-1: Browser-Objekte und ihre Beziehungen zueinander

2.10.1 Window - Browserfenster oder Frame

| | | |
|--|--|---|
| Window | Zugriff self window window.frame[] |  |
| Eigenschaften closed, defaultStatus, document, frames[], history, length, location, name, navigator, offScreenBuffering, opener, parent, screen, self, status, top, window NN: crypto, innerHeight, innerWidth, java, locationbar, menubar, netscape, outerHeight, outerWidth, Packages, pageXOffset, pageYOffset, personalbar, screenX, screenY, scrollbars, statusbar, sun, toolbar IE: clientInformation, event | Methoden alert(), blur(), clearInterval(), clearTimeout(), close(), confirm(), focus(), moveBy(), moveTo(), open(), prompt(), resizeBy(), resizeTo(), scroll(), scrollBy(), scrollTo(), setInterval(), setTimeout() NN: atob(), back(), btoa(), captureEvents(), disableExternalCapture(), enableExternalCapture(), find(), forward(), handleEvent(), home(), print(), releaseEvents(), routeEvent(), setHotKeys(), setResizable(), setZOptions(), stop() IE: navigate() | Ereignisse onblur, onerror, onfocus, onload, onresize, onunload NN: ondragdrop, onmove |

Eigenschaften

| | |
|--|------------|
| clientInformation | IE4 |
| Siehe navigator-Eigenschaft | |
| closed | 1.1 |
| Gibt an, ob das Fenster geschlossen wurde. | |
| crypto | NN4 |
| Verweis auf das Crypto-Objektes eines Fensters. | |
| defaultStatus | 1.0 |
| Voreingestellter Text für die Statuszeile des Fensters. | |
| document | 1.0 |
| Verweis auf das Dokumentenobjekt document in diesem Fenster. | |
| event | IE4 |
| Beschreibt das zuletzt aufgetretene Ereignis im Objekt event. | |
| frames [] | |
| Ein Array aller Frames des Fensters. | |
| history | |
| Verweis auf das history-Objekt des Fensters (Aufzeichnungspfad). | |

| | |
|--|----------------|
| innerHeight, innerWidth | NN4 |
| Die Höhe und die Breite des Anzeigebereich für das Dokuments in Pixeln. | |
| java | NN3 |
| Das <code>java.*</code> -Package von LiveConnect. | |
| locationbar | NN4 |
| Gibt an, ob die Leiste mit dem URL-Eingabefeld sichtbar ist. | |
| length | |
| Die Anzahl der Frames in dem Fenster (<code>window.length</code>). | |
| location | |
| Die URL des Dokumentes, das im Fenster angezeigt wird. Beim Setzen dieser Eigenschaft wird als Folge das der gesetzten URL entsprechende Dokument beschafft und angezeigt (siehe <code>location</code> -Objekt). | |
| menubar | NN4 |
| Zeigt an, ob die Menuleiste angezeigt wird. | |
| name | 1.0/1.1 |
| Der Name des Fensters. | |
| navigator | |
| Ein Verweis auf das <code>navigator</code> -Objekt. | |
| netscape | NN3 |
| Das Java-LiveConnect-Package <code>netscape.*</code> . | |
| offScreenBuffering | 1.2 |
| Änderungen können zwischengepuffert werden, um z.B. Flackern bei der Vornahme dieser Änderungen zu vermeiden (Doppelpufferung). | |
| opener | 1.1 |
| Verweis auf dasjenige Fenster, das das angegebene geöffnet hat (<code>fenster.opener</code>). | |
| outerHeight, outerWidth | NN4 |
| Die äußere Höhe und die äußere Breite des Fensters in Pixeln. | |
| Packages | NN3 |
| LiveConnect-Packages der Java-Klassen. | |
| pageXOffset, pageYOffset | NN4 |
| Die aktuelle horizontale und vertikale Rollposition. | |
| parent | |
| Das unmittelbar übergeordnete Fenster bzw. der übergeordnete Frame. | |
| personalbar | NN4 |
| Gibt an, ob vom Browser der persönliche Toolbar angezeigt wird. | |
| screenX, screenY | NN4 |
| Die Koordinaten des Fensters auf dem Bildschirm (Ecke links oben). | |

| | |
|---|------------|
| scrollbars | NN4 |
| Gibt an, ob Rollbalken angezeigt werden. | |
| screen | 1.2 |
| Objekt mit Informationen über den Bildschirm (siehe <code>screen</code> -Objekt). | |
| self | |
| Verweis auf das Fenster selbst (siehe <code>window</code> -Eigenschaft). | |
| status | |
| Transienter Text der Statuszeile (dauerhafter ist <code>defaultStatus</code>). | |
| statusbar | NN4 |
| Gibt an, ob die Statuszeile angezeigt wird. | |
| sun | NN3 |
| Das Java-LiveConnect-Package <code>sun.*</code> . | |
| toolbar | NN4 |
| Gibt an, ob der Toolbar des Browsers angezeigt wird. | |
| top | |
| Verweis auf das „Top-Level“-Fenster. | |
| window | |
| Verweis auf das Fenster selbst (siehe <code>top</code>). | |

Methoden

| | |
|---|------------|
| alert (nachricht) | |
| Nachricht in Dialogbox anzeigen | |
| atob (string64) | NN4 |
| Dekodiere einen Base64-kodierten String. | |
| back () | NN4 |
| Zum vorangegangenen Dokument navigieren. | |
| blur () | 1.1 |
| Eingabefokus von Fenster nehmen. | |
| btoa (bindaten) | NN4 |
| Codiere Binärdatenstring mittels Base64 (siehe auch <code>atob ()</code>). | |
| captureEvents (ereignismaske) | NN4 |
| Gibt mittels Ereignismaske die Ereignisse an, die beim Auftreten in Dokumenten (<code>document</code> -Objekt) und Windows (<code>window</code> -Objekte) auftreten können. | |
| clearInterval (intervallID) | 1.2 |
| Bricht die periodisch wiederholte Ausführung eines Programmes ab (<code>intervallID</code> ist der Rückgabewert von <code>setInterval ()</code>). | |

- clearTimeout (timeoutID)**
Annulliert den mit `timeoutID` gekennzeichneten verzögerten Auftrag für die Ausführung eines Programmes (`setTimeout ()`).
- close ()**
Browserfenster schließen.
- confirm ()**
Stellt eine Ja/Nein-Frage in einem Dialogfenster zur Beantwortung.
- find (suchstring, grossklein, rueckwaerts)** NN4
Einen String `suchstring` im aktuellen Fenster suchen. `grossklein` gibt an, ob Groß-/Kleinschreibung unterschieden werden soll. `rueckwaerts` gibt die Suchrichtung an.
- forward ()** NN4
Im Aufzeichnungspfad (History) eins vorwärts, sofern möglich.
- focus ()** 1.1
Das Fenster erhält den Einabefokus.
- handleEvent (ereignis)** NN4
Übergibt ds Ereignisobjekt `ereignis` an einen passenden Eventhandler des aktuellen Objektes, falls vorhanden.
- home ()** NN4
Navigiert zur voreingestellten Startseite des Browsers.
- moveBy (dx, dy)** 1.2
Fenster um `dx` in X-Richtung und `dy` in Y-Richtung verschieben.
- moveTo (x, y)** 1.2
Versetzt das Fenster so, dass die linke obere Ecke des Fensters die Koordinatenposition `x, y` einnimmt.
- navigate ()** IE3
Eine neue URL laden.
- open (url, name, merkmale, ersetzen)** 1.0, 1.1
Öffnet ein neues Fenster. Das Fenster ist leer, wenn `url` nicht angegeben ist, sonst wird das der URL entsprechende Dokument angezeigt. `name` ist der optionale Name des neuen Fensters, und `replace` besagt (optional), ob im Aufzeichnungspfad (History) ein neuer Eintrag erfolgen soll (`false`) oder der alte ersetzt wird (`true`). `merkmale` ist ein String mit einer kommagetrennten Liste, mittels derer das neue Fenster konfiguriert wird. (Der String darf keine Leerzeichen enthalten!)
Die einzelnen Listeneinträge haben die Form
`merkmal=wert` oder `merkmal`
Merkmale sind insbesondere (nur die allgemein verwendbaren sind aufgeführt):
`directories=yes|no` Anzeige der Knöpfe
`height=pixel` Höhe der Anzeigefläche im Fenster in Pixeln

| | | |
|--|--|------------|
| <code>location=yes no</code> | Das Eingabefeld für URLs wird angezeigt | |
| <code>menubar=yes no</code> | Menüleiste wird angezeigt | |
| <code>resizable=yes no</code> | Die Größe des Fensters kann verändert werden | |
| <code>scrollbars=yes no</code> | Das Fenster hat nötigenfalls Rolllisten | |
| <code>status=yes no</code> | Die Statusanzeige ist sichtbar | |
| <code>toolbar=yes no</code> | Die Navigationsleiste (Toolbar) wird angezeigt | |
| <code>width=pixel</code> | Breite der Anzeigefläche im Fenster in Pixeln | |
| <code>print()</code> | | NN4 |
| | Das aktuell angezeigte Dokument ausdrucken, d.h. der Druckmanager wird gestartet. | |
| <code>prompt()</code> | | |
| | Fordert in einem Dialogfenster zur Eingabe eines Strings auf. | |
| <code>releaseEvents(ereignismaske)</code> | | NN4 |
| | Beendet das Abfangen von Ereignissen. | |
| <code>resizeBy(dbreite, dhoehe)</code> | | 1.2 |
| | Verändert die Größe um <code>dbreite</code> horizontal und <code>dhoehe</code> vertikal (Werte in Pixeln). | |
| <code>resizeTo(breite, hoehe)</code> | | 1.2 |
| | Verändert die Größe auf <code>breite</code> horizontal und <code>hoehe</code> vertikal (Werte in Pixeln). | |
| <code>routeEvent(ereignis)</code> | | NN4 |
| | Gibt das Ereignis <code>ereignis</code> an den nächsten Eventhandler weiter. | |
| <code>scroll(x, y)</code> | | 1.1 |
| | Verschiebe das Dokument im Anzeigefenster auf die angegebene x,y-Position. (Verworfen in 1.2.) | |
| <code>scrollBy(dx, dy)</code> | | 1.2 |
| | Verschiebe das Dokument um <code>dx</code> Pixel horizontal und um <code>dy</code> Pixel vertikal. | |
| <code>scrollTo()</code> | | 1.2 |
| | Ersetzt <code>scroll()</code> , siehe dort. | |
| <code>setInterval(javascriptcode, zeitintervall)</code> | | 1.2 |
| | Führt das JavaScript-Programm <code>javascriptcode</code> periodisch aus. | |
| <code>setResizable(veraenderbar)</code> | | NN4 |
| | Je nach Wert von <code>veraenderbar</code> kann die Fenstergröße verändert (<code>true</code>) oder fixiert (<code>false</code>) werden. | |
| <code>setTimeout(javascriptcode, wartezeit)</code> | | |
| | Starte das Programm <code>javascriptcode</code> nach Ablauf der Wartezeit <code>wartezeit</code> . | |
| <code>stop()</code> | | NN4 |
| | Stoppt das Laden des Seiteninhaltes. | |
| Ereignisse | | |
| <code>onblur</code> | | 1.1 |
| | Eventhandler für Fokusverlust. | |

| | |
|---|------------|
| ondragdrop | NN4 |
| Eventhandler, der aufgerufen wird, wenn mit der Maus Elemente in das Browserfenster gezogen werden. | |
| onerror | 1.1 |
| Eventhandler für den Fall von syntaktischen Fehlern. | |
| onfocus | 1.1 |
| Eventhandler für Fokuserhalt. | |
| onload | |
| Eventhandler, mit dem das Laden eines Dokumentes abgeschlossen wird. | |
| onmove | NN4 |
| Wenn das Browserfenster als Ganzes verschoben wird. | |
| onresize | 1.2 |
| Eventhandler für das Ändern der Fenstergröße. | |
| onunload | |
| Wird aufgerufen, wenn der Browser eine Seite verlässt. | |

2.10.2 Event - Einzelheiten zu Ereignissen

| Event | Zugriff function <i>handler(event)</i> NN4 <i>window.event</i> IE4 | Unverträgliche Varianten beim Netscape Navigator und beim MS Internet Explorer |
|--|---|---|
| Eigenschaften NN NN: data, height, layerX, layerY, modifiers, pageX, pageY, screenX, screenY, target, type, TYPE, which, width, x, y | Eigenschaften IE IE: altKey, button, cancelBubble, clientX, clientY, ctrlKey, fromElement, keyCode, offsetX, offsetY, reason, returnValue, screenX, screenY, shiftKey, srcElement, srcFilter, toElement, type, x, y | |

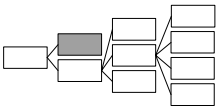
Eigenschaften

| | |
|--|------------|
| altKey | IE4 |
| Gibt an, ob die Alt-Taste gedrückt war. | |
| button | IE4 |
| Gibt an, welche Maustaste gedrückt war. | |
| cancelBubble | IE4 |
| Beendet die Weitergabe des Ereignisses. | |
| clientX, clientY | IE4 |
| Die Koordinaten eines Events in Bezug auf die Seite. | |

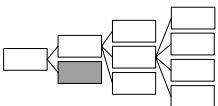
| | |
|--|-----------------|
| ctrlKey | IE4 |
| Gibt an, ob die Strg-Taste gedrückt war. | |
| fromElement | IE4 |
| Bei Mouseover- und Mouseout-Ereignissen das Objekt, auf das sich die Maus bezieht. | |
| height | NN4 |
| Die neue Höhe eines Fensters oder Frames, dessen Größe verändert wurde. | |
| keyCode | IE4 |
| Gibt den Unicode der gedrückten Taste zurück. | |
| layerX, layerY | NN4 |
| Die Koordinaten eines Events in einem Layer. | |
| modifiers | NN4 |
| Die Bitmaske für gedrückte Modifikatortasten wie Umschalte-, Alt- und Strg-Taste. | |
| offsetX, offsetY | IE4 |
| Die Koordinaten des Ereignissen relativ zum Container, in dem es aufgetreten ist (insbesondere HTML-Elemente). | |
| pageX, pageY | NN4 |
| Die Koordinaten eines Events in Bezug auf die Seite. | |
| reason | IE4 |
| Status der Datenübertragung. | |
| returnValue | IE4 |
| Definiert den Rückgabewert eines Eventhandlers. | |
| screenX, screenY | NN4, IE4 |
| Die Event-Koordinaten relativ zum Bildschirm. | |
| srcElement | IE4 |
| Das Objekt, in dem das Ereignis ausgelöst wurde. | |
| shiftKey | IE4 |
| Gibt an, ob die Umschalttaste gedrückt war. | |
| srcFilter | IE4 |
| Das Event, das bei einer Filteränderung erzeugt wurde. | |
| target | NN4 |
| Das Objekt, in dem das Ereignis ausgelöst wurde. | |
| toElement | IE4 |
| Das Element, auf das sich die Maus bewegt hat (Mouseover, Mouseout). | |
| type | NN4, IE4 |
| Der Typ des Ereignisses. | |
| TYPE | NN4 |
| Ereignistypkonstante für Ereignismasken. | |

| | | |
|--------------|---|------------|
| which | | NN4 |
| | Gibt an, welche Maustaste oder welche Taste auf der Tastatur gedrückt wurde. | |
| width | | NN4 |
| | Die neue Breite eines Fensters oder Frames, dessen Größe verändert wurde. | |
| x, y | | NN4 |
| | Synonym mit <code>screenX</code> , <code>screenY</code> | |
| x, y | | IE4 |
| | Die Koordinaten des Events auf dem Element (Bild z.B.), in dem es verursacht wurde. | |

2.10.3 Frames

| | | |
|--------------------------------------|--|---|
| Frame | Zugriff <i>window.frames[index]</i> <i>window.frames.length</i> <i>frames[index]</i> <i>frames.length</i> |  |
| Eigenschaften siehe Window | Methoden siehe Window | Ereignisse siehe Window |

2.10.4 History - URL-Aufzeichnungspfad des Browsers

| | | |
|----------------------|---|---|
| History | Zugriff <i>window.history</i> <i>frame.history</i> <i>history</i> |  |
| Eigenschaften | Methoden <i>back()</i> , <i>forward()</i> , <i>go()</i> | Ereignisse |

Methoden

back()

Im Verzeichnispfad zur vorangehenden URL zurückgehen.

forward()

Im Verzeichnispfad vorwärts zur nächsten URL gehen.

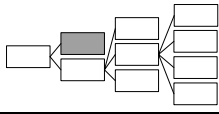
go(schritte)

Die Anzahl *schritte* an Schritten im Verzeichnispfad vorwärts (positiver Wert) oder rückwärts (negativer Wert) bewegen.

go(url)

Aktiviert diejenige URL im Verzeichnispfad, die den String *url* enthält.

2.10.5 Location - die URL-Eingabezeile des Browsers

| | | |
|--|--|--|
| Location | Zugriff location, window.location |  |
| Eigenschaften hash, host, hostname, href, pathname, port, protocol, search | Methoden reload(), replace() | Ereignisse |

Eigenschaften

Fast alle Eigenschaften extrahieren Bestandteile aus der aktuellen URL. Als Vorlage für die Erläuterung der Eigenschaften dient die folgende URL:

```
protocol://hostname:protocol/pathname?searchstring#hash
```

hash

Ankername.

host

Kombination aus hostname und protocol

hostname

Hostanteil der URL.

href

Einlesen der gesamten aktuellen URL. Setzen dieser Eigenschaft mit einer URL bewirkt deren Aktivierung, d.h. das entsprechende Dokument wird geladen und angezeigt.

pathname

Verzeichnispfad und Dateiname der URL.

port

Portnummer

protocol

URL-Typ, meist http.

search

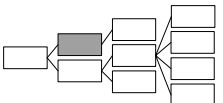
Query-String, meist aus Formularen. Inhalt ist eine fragezeichengetrennte Liste mit Einträgen der Form

```
name=wert
```

Methoden

- reload** (erzwingen) 1.1
 Das aktuelle Dokument wird erneut geladen. Ist der Parameter `erzwingen` auf `true` gesetzt, so wird das Original, also keine Cache- oder Proxy-Kopie, geladen.
- replace** () 1.1
 Ein neues Dokument ersetzt das aktuelle, ohne einen neuen Eintrag im URL-Aufzeichnungspfad (History) zu bewirken.

2.10.6 Navigator - Informationen über den Browser

| | | |
|--|--|--|
| Navigator | Zugriff navigator |  |
| Eigenschaften appCodeName, appName, appVersion, platform, userAgent NN: language, mimeTypees[], plugins[] IE: userLanguage, systemLanguage | Methoden javaEnabled(), taintEnabled() NN: plugins.refresh() | Ereignisse |

Eigenschaften

- appCodeName**
 Der Codename des Browsers, z.B. „Mozilla“ für den Netscape Navigator.
- appName**
 Der Name des Browsers, z.B. „Navigator“ oder „Internet Explorer“
- appVersion**
 Version des Browsers sowie seine Betriebssystemplattform.
- language** NN4
 Die eingestellte Sprache.
- mimeTypees []** 1.1
 Ein Array mit den unterstützten MIME-Typen (leer beim IE).
- platform** 1.2
 Das Betriebssystem, unter dem der Browser läuft.
- plugins []** 1.1
 Ein Array mit den installierten Plugins (leer beim IE).
- systemLanguage** IE4
 Die Sprache des zugrundeliegenden Betriebssystems.

userAgent

Der Wert des „user-agent“ im HTTP-Datenkopf.

userLanguage

Die Sprache des Benutzers

IE4

Methoden

javaEnabled()

true, wenn Java verfügbar und aktiviert ist.

1.1

taintEnabled()

Veraltet.

plugins.refresh(neuladen)

Neu installierte Plugins dem Browser verfügbar machen. Ist neuladen auf true gesetzt, so wird außerdem das aktuelle Dokument neu geladen.

NN4

2.11 document und darunter (außer forms)

In diesem Abschnitt sind das document-Objekt und die HTML-Objekte, die es enthalten kann, zusammengefasst. Ausgenommen sind Formularobjekte, denen ein eigener Abschnitt gewidmet ist.

2.11.1 HTML-Element

Der Internet Explorer verwendet ein Dokumentenmodell, bei dem alle HTML-Elemente eine gemeinsame Wurzel haben, aus der diese alle erben. Die hier aufgeführten Eigenschaften und Methoden sind also nur für den Internet Explorer von Belang, d.h. sie haben keine Relevanz für HTML-Objekte des Netscape Navigators.

| | | |
|--|---|--|
| <p>HTML-Element Internet Explorer</p> | <p>Zugriff document.all[index] document.elementName document.images[index] document.links[index] document.anchors[index] document.forms[index]</p> | <p>Vererbt an images[], links[], anchors[], forms[] und elements[]</p> |
| <p>Eigenschaften IE: all[], children[], className, document, id, innerHTML, innerText, lang, offsetHeight, offsetLeft, offsetTop, offsetWidth, outerHTML, outerText, parentElement, sourceIndex, style, tagName, title</p> | <p>Methoden IE: contains(), getAttribute(), insertAdjacentHTML(), insertAdjacentText(), removeAttribute(), scrollIntoView(), setAttribute()</p> | <p>Ereignisse onclick, ondblclick, onhelp, onkeydown, onkeypress, onkeyup, onmousedown, onmousemove, onmouseout, onmouseover, onmouseup</p> |

Eigenschaften

all []

Alle Elemente, also auch die Elemente, die in anderen Elementen beinhaltet sind.

children []

Alle Elemente, die dieses Element unmittelbar beinhaltet.

className

Der Wert des CLASS-Attributs, falls vorhanden.

document

Verweis auf das Dokument, das das Element enthält.

id

Wert des ID-Attributs.

innerHTML

Text in einem HTML-Element, ausgenommen die spitzen Klammern.

innerText

Der Text zwischen Start- und Ende-Markup, die Markups exklusive.

lang

Der Wert des LANG-Attributs.

offsetHeight, offsetWidth

Die Höhe bzw. die Breite des Elementes in Pixeln.

offsetLeft, offsetTop

Die X- bzw. Y-Koordinate des Elementes (Ecke links oben), und zwar relativ zu dem Element bzw. Container, der es enthält.

outerHTML

Text in einem HTML-Element, einschließlich der spitzen Klammern.

outerText

Der Text zwischen Start- und Ende-Markup, diese Markups einschließlich.

parentElement

Dasjenige Element, das direkt dieses Element beinhaltet.

sourceIndex

Gibt den Index zurück, das dieses Element im Array bzw. Container `all []` hat.

style

Der Inline-CSS-Stil dieses Elements.

tagName

Gibt den Namen des Markups (des „Tags“) zurück.

title

Gibt das TITLE-Attribut dieses Elements zurück (`htmlElement.title`).

Methoden

contains (zielelement)

Gibt an, ob das Element sich im Element `zielelement` befindet.

getAttribute (name)

Gibt den Wert des Attributes `name` zurück.

insertAdjacentHTML (wo, text)

Der HTML-Text `text` wird an der durch `wo` bezeichneten Stelle eingefügt.

insertAdjacentText (wo, text)

Der Text `text` wird an der durch `wo` bezeichneten Stelle eingefügt.

removeAttribute (name)

Entfernt das Attribut `name`.

scrollIntoView (oben)

Das Element `element` in den sichtbaren Bereich rollen (`true`: Element am oberen Rand; `false`: am Rand unten).

setAttribute (name, wert)

Ein neues Attribut mit dem Namen `name` und dem Attributwert `wert` definieren.

Ereignisse

| | |
|--------------------------|--|
| <code>onclick</code> | Klicken auf ein Element. |
| <code>ondblclick</code> | Doppelklick auf ein Element. |
| <code>onhelp</code> | Bei Anforderung von Hilfe per F1-Taste. |
| <code>onkeydown</code> | Taste wurde niedergedrückt. |
| <code>onkeypress</code> | Taste wurde gedrückt und losgelassen. |
| <code>onkeyup</code> | Taste wurde losgelassen. |
| <code>onmousedown</code> | Maustaste wurde niedergedrückt. |
| <code>onmousemove</code> | Maus wurde bewegt. |
| <code>onmouseout</code> | Maus wurde von einem Element weggezogen. |
| <code>onmouseover</code> | Maus wurde über ein Element bewegt. |
| <code>onmouseup</code> | Maustaste wurde losgelassen. |

2.11.2 Anchor - Ziel von Hypertextlinks

| | | |
|--|--|-------------------|
| Anchor <code><A ...></code> Version 1.2 | Zugriff <code>document.anchors[index]</code> <code>document.anchors.length</code> | |
| Eigenschaften: name NN: text, x, y | Methoden | Ereignisse |

HTML-Syntax

```
<A Href=url
    Name=#ankername ...
    Target=fenstername>
Text
</A>
```

Eigenschaften

- name** 1.2
Der Name des Ankers.
- text** NN4
Der Text zwischen den Ankermarken <A...> und . (Beim Internet Explorer die Eigenschaft innerText von HTML-Element.)
- x, y** NN4
Die Koordinaten des Ankers relativ zum aktuellen Dokument, und zwar zu dessen Ecke links oben. (Beim IE die Eigenschaft offsetLeft und offsetTop von HTML-Element als Superklasse aller HTML-Elemente.)

2.11.3 Applet - Einbettung eines Java-Applets

| | | |
|---|---|-------------------|
| Applet <APPLET ...> Version 1.1 | Zugriff document.applets[index] document.applets.length document.appletName | |
| Eigenschaften → als public gekennzeichnete Java-Objekteigenschaften des Applets | Methoden → als public gekennzeichnete Java-Objektmethoden des Applets | Ereignisse |

HTML-Syntax

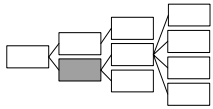
```
<APPLET Code=javaklasse
        Codebase=url
        Width=pixel    Height=pixel
        Alt=text
        Align=left|right|top|middle|bottom
        Hspace=pixel   Vspace=pixel
>
...
</APPLET>
```

Zwei weitere Attribute, Mayscript und Archive, sind Netscape-spezifisch und daher weggelassen.

2.11.4 Area - "Hotspot" in einer Bildkarte

Siehe Link

2.11.5 Document - Das HTML-Dokument

| | | |
|---|---|--|
| Document <BODY ...> | Zugriff window.document, document |  |
| Eigenschaften alinkColor, anchors[], applets[], bgColor, cookie, domain, embeds[], fgColor, forms[], images[], lastModified, linkColor, links[], location, plugins[], referrer, title, URL, vlinkColor NN: classes, height, ids, layers[], tags, width IE: activeElement, all[], charset, children[], defaultCharset, expando, parentWindow, readyState | Methoden clear(), close(), open(), write(), writeln() NN: captureEvents(), contextual(), getSelection(), releaseEvents(), routeEvents() IE: elementFromPoint() | Ereignisse onload, onunload → Window |

Eigenschaften

- activeElement** **IE4**
 Zeigt auf das Eingabeelement, das den Eingabefokus hat.
- alinkColor**
 Die Farbe einer aktivierten Link.
- all []** **IE4**
 Alle HTML-Elemente eines Dokumentes.
- anchors []** **1.0, 1.2**
 Array mit allen Ankern im Dokument.
- applets []** **1.1**
 Array mit allen Applets des Dokumentes.
- bgColor**
 Hintergrundfarbe des Dokumentes
- charset** **IE4**
 Der Zeichensatz, der im Dokument aktuell verwendet wird.

| | |
|--|-----------------|
| children [] | IE4 |
| Alle Elemente, die unmittelbar in dem Dokument enthalten sind (Kindelemente). Bitte beachten: In <code>all []</code> sind auch die Kindelemente von Kindelementen enthalten. | |
| classes | NN4 |
| Definiert Stilklassen (CSS). | |
| cookie | |
| Die Cookies eines Dokumentes. Sie werden entsprechend dem folgenden Muster gesetzt: | |
| <pre>document.cookie = "name=wert;expires=datum;path=pfad;domain=domaene;secure"</pre> | |
| defaultCharset | IE4 |
| Standardzeichensatz des Dokumentes. | |
| domain | 1.1 |
| Sicherheitsdomäne des Dokumentes. | |
| embeds [] | 1.1 |
| Die eingebetteten Objekte des Dokumentes. | |
| expando | IE4 |
| Verbietet die Definition neuer Eigenschaften in einem Dokumentenobjekt. | |
| fgColor | |
| Farbe des Textes im Dokument. | |
| forms [] | |
| Alle Formulare im Dokument. | |
| height, width | NN4 |
| Die vertikale und horizontale Größe des Dokuments in Pixeln. | |
| ids | NN4 |
| Stildefinition für individuelle HTML-Markups. | |
| images [] | 1.1 |
| Alle eingebetteten Bilder des Dokumentes. | |
| lastModified | |
| Datum und Uhrzeit der letzten Änderung im Dokument. | |
| layers [] | NN4 |
| Alle <code>LAYER</code> -Elemente im Dokument. | |
| linkColor | |
| Die Farbe von (unbesuchten) Textankern. | |
| links [] | |
| Alle Link-Objekte im Dokument. | |
| location siehe URL | 1.0, 1.1 |
| Die URL des aktuellen Dokumentes. | |

| | |
|---|------------|
| parentWindow | IE4 |
| Verweis auf das Fenster, in dem das Dokument angezeigt wird. | |
| plugins [] | |
| Array mit allen eingebetteten Objekten (synonym zu <code>embed</code>). | |
| readyState | IE4 |
| Ladestatus des Dokumentes; "complete", wenn vollständig. | |
| referrer | |
| Enthält die URL des Dokumentes, von dem aus dieses Dokument geladen wurde. | |
| tags | NN4 |
| Definiere den Stil von HTML-Markups. | |
| title | |
| Der Titel des Dokumentes, der mittels <code>TITLE</code> -Markup definiert wurde. | |
| URL | 1.1 |
| Die URL des aktuellen Dokumentes. | |
| vlinkColor | |
| Die Farbe einer besuchten Link. | |
| Methoden | |
| captureEvents () | NN4 |
| Siehe <code>window.captureEvents ()</code> | |
| clear () | |
| Veraltet, <code>open ()</code> verwenden. | |
| close () | |
| Den Ausgabestrom zu einem Dokument abschließen. | |
| contextual (stil1, stil2, ...) | NN4 |
| Definiere einen kontextabhängigen Stil. | |
| elementFromPoint (x, y) | IE4 |
| Bestimmt das Element, das an der Koordinate <code>x, y</code> zu finden ist. | |
| open (mimetype) | |
| Ein neues Dokument beginnen. | |
| releaseEvents () | NN4 |
| Siehe <code>window.releaseEvents ()</code> | |
| routeEvents () | NN4 |
| Siehe <code>window.routeEvents ()</code> | |
| write (wert1, wert2, ...) | |
| Mit <code>document.write ()</code> in einem Dokument ausgeben. | |

writeln(wert1, wert2, ...)

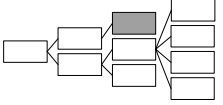
Wie `write()`, aber mit einem Zeilenumbruch als Abschluss.

Ereignisse

Siehe Window-Objekte:

onload, onunload

2.11.6 Image

| | | |
|--|--|--|
| <p>Image </p> | <p>Zugriff <i>document.images[], document.images.length, document.imageName</i></p> |  |
| <p>Eigenschaften border, complete, height, hspace, lowsrc, name, src, vspace, width</p> | <p>Methoden</p> | <p>Ereignisse onabort, onerror, onload</p> |

Eigenschaften

border

Dicke des Bilderrahmens (0: kein Rahmen)

complete

Gibt an, ob das Bild komplett geladen ist.

height, width

Breite und Höhe des Bildes in Pixeln.

hspace, vspace

Horizontaler und vertikaler Abstand des Bildes zu seiner Umgebung.

lowsrc

Zwischenbild mit geringerer Auflösung, das bis zum vollständigen Laden des Bildes angezeigt wird.

name

Der Name des Bildes.

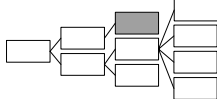
src

Die URL der eingebetteten Bilddatei.

Ereignisse

| | |
|----------------|---|
| onabort | Bei benutzerverursachtem Abbruch des Ladevorganges. |
| onerror | Bei Übertragungsfehlern während des Ladens. |
| onload | Nach erfolgreichem Laden. |

2.11.7 Link - Hypertext-Link

| | |
|--|---|
| Link , , <AREA Href=...> | Zugriff <i>document.links[], document.links.length</i>  |
| Eigenschaften hash, host, hostname, href, pathname, port, protocol, search, target, text, x, y | Methoden Ereignisse onclick, onmouseout, onmouseover |

Eigenschaften

Fast alle Eigenschaften dieser Objekte extrahieren Bestandteile aus der aktuellen URL. Als Vorlage für die Erläuterung der Eigenschaften dient die folgende Muster-URL:

`protocol://hostname:protocol/pathname?searchstring#hash`

hash

Ankername.

host

Kombination aus hostname und protocol

hostname

Hostanteil der URL.

href

Einlesen der gesamten aktuellen URL. Setzen dieser Eigenschaft mit einer URL bewirkt deren Aktivierung, d.h. ein neues Dokument wird geladen und angezeigt.

pathname

Verzeichnispfad und Dateiname der URL.

port

Portnummer.

protocol

URL-Typ, meist http.

search

Query-String, meist aus Formularen. Inhalt ist eine fragezeichengetrennte Liste mit Einträgen der Form

`name=wert` .

target

Das Zielfenster der Link.

text

Der Text zwischen den A-Markups.

NN4**x, y**

Die Koordinaten der Hypertext-Link.

NN4**Ereignisse****onclick**

Beim Anklicken eines Elements.

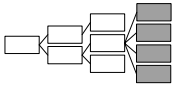
onmouseout

Beim Hinziehen des Mauszeigers zu einem Element.

onmouseover

Beim Wegziehen des Mauszeigers von einem Element.

2.12 Formular und Formularobjekte (forms)**2.12.1 Input - Die Eigenschaften von Formularobjekten**

| | | |
|--|--|---|
| Input <INPUT Type=...> | Zugriff <i>form.elements[],</i> <i>form.inputFeldName</i> |  |
| Eigenschaften: checked, defaultChecked, defaultValue, form, length, name, options[], selectedIndex, type, value | Methoden blur(), click(), focus(), select() | Ereignisse onblur, onchange, onclick, onfocus |

HTML-Syntax

```

<FORM ... >
  <INPUT Type=button ... >
  <INPUT Type=submit ... >
  <TEXTAREA ...>...</TEXTAREA>
  <SELECT ...>...</SELECT>
  <INPUT Type=hidden ... >
  ...
</FORM>

```

| Elementtyp (Type= <i>elementType</i>) | Eigenschaften | | | | | | | | Methoden | | | | | Ereignisse | | | |
|---|---------------|----------------|--------------|------|--------|------|-----------|---------------|----------|-------|--------|---------|---------|------------|--------|----------|---------|
| | checked | defaultChecked | defaultValue | form | length | name | options[] | selectedIndex | type | value | blur() | click() | focus() | select() | onblur | onchange | onclick |
| button | | | | X | | X | | | X | X | X | X | X | | | X | X |
| checkbox | X | X | | X | | X | | | X | X | X | X | X | | X | | X |
| radio | X | X | | X | | X | | | X | X | X | X | X | | X | | X |
| reset | | | | X | | X | | | X | X | X | X | X | | X | | X |
| submit | | | | X | | X | | | X | X | X | X | X | | X | | X |
| text | | | X | X | | X | | | X | X | X | | X | X | X | X | X |
| textarea | | | X | X | | X | | | X | X | X | | X | X | X | X | X |
| password | | | X | X | | X | | | X | X | X | | X | X | X | X | X |
| fileupload | | | | X | | X | | | X | X | X | | X | X | X | X | X |
| select | | | | X | X | X | X | X | X | | X | | X | | X | X | |
| hidden | | | | X | | X | | | X | X | | | | | | | |

Eigenschaften

checked

Kästchen ist angekreuzt (checkbox, radio)

defaultChecked

Kästchen ist anfangs angekreuzt.

defaultValue

Was in Texteingabefeldern anfangs angezeigt wird (text, textarea, password).

form

Referenz auf das Formular, in dem das INPUT-Element enthalten ist.

length

Gibt an, wieviele Optionen das SELECT-Element enthält.

name

Der Name eines Formularelementes.

options []

Optionen-Array in einem SELECT-Element.

selectedIndex

Nummer einer gewählten Option in einem SELECT-Element.

type

Der Typ des Formularelementes als String, z.B. "button" für das Element
<INPUT Type=button ...> oder "textarea" für das TEXTAREA-Element.

value

Wert des Elementes. Bei Button, Submit und Reset die Beschriftung des Knopfes, bei Eingabeelementen der eingegebene bzw. angezeigte Text etc.

Methoden**blur()**

Funktion, um einem Element den Eingabefokus zu nehmen.

click()

Funktion, um ein Kästchen (radio, checkbox) abzuhaken oder einen Knopf anzuklicken.

focus()

Funktion, um einem Element den Eingabefokus zu geben.

select()

Funktion, um Text in Eingabefeldern zu markieren.

Ereignisse

- onblur** Eventhandler, der beim Verlust des Fokus auf das Element aktiviert wird.
- onchange** Wird aktiviert bei Textelementen, wenn der Text geändert wird.
- onclick** Wird aktiviert, wenn auf das Element per Maus geklickt wird (1.0,1.1).
- onfocus** Eventhandler, wird beim Einstellen des Fokus auf das Element aktiviert.

2.12.2 Button - Taster

| | | |
|---|---|---|
| Button <INPUT Type=Button ...> | Zugriff <i>form.elements[index]</i> <i>form.buttonName</i> | |
| Eigenschaften form, name, type, value | Methoden blur(), click(), focus() | Ereignisse onblur, onclick, onfocus |

HTML-Syntax

```

<FORM ...>...
  <INPUT Type=button
    Name=elementname
    Value=aufschrift
    onblur=eventhandler
    onclick=eventhandler
    onfocus=eventhandler
  >
...

```

</FORM>

Eigenschaften

- form**
Verweis auf das Formular, in dem dieses Element enthalten ist.
- name**
Name des Formularelementes.
- type**
Spezifiziert den Typ des Formularelementes.
- value**
Beschriftung des Tasters.

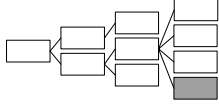
Methoden

- blur ()**
Funktion, um einem Element den Eingabefokus zu nehmen.
- click ()**
Funktion, um ein Kästchen (radio, checkbox) abzuhaken oder einen Knopf anzuklicken.
- focus ()**
Funktion, um einem Element den Eingabefokus zu geben.

Ereignisse

- onblur** Eventhandler, der beim Verlust des Fokus auf das Element aktiviert wird.
- onclick** Wird aktiviert, wenn auf das Element per Maus geklickt wird (1.0,1.1).
- onfocus** Eventhandler, wird beim Einstellen des Fokus auf das Element aktiviert.

2.12.3 Checkbox - Kästchen zum Ankreuzen

| | | |
|---|---|--|
| Checkbox <INPUT Type=Checkbox ...> | Zugriff <i>form.elements[index]</i> <i>form.checkboxName /</i> <i>form.checkboxName[index]</i> <i>form.checkboxName.length</i> |  |
| Eigenschaften checked, defaultChecked, form, name, type, value | Methoden blur(), click(), focus() | Ereignisse onblur, onclick, onfocus |

HTML-Syntax

```
<FORM ... >
  <INPUT Type=checkbox
    Name=elementodergruppenname
    Value=aufschrift
    Checked
    onBlur=eventhandler
    onClick=eventhandler
    onFocus=eventhandler
  >
  ...
</FORM>
```

Eigenschaften

checked

Das Kästchen ist angekreuzt (abgehakt).

defaultChecked

Dieses Kästchen ist per HTML-Attribut `Checked` angekreuzt..

form

Verweis auf das Formular, in dem dieses Element enthalten ist.

name

Name des Formularelementes.

type

Spezifiziert den Typ des Formularelementes.

value

Dieser Wert wird mit dem Formular verschickt.

Methoden

blur()

Funktion, um einem Element den Eingabefokus zu nehmen.

click()

Funktion, um ein Kästchen (`radio`, `checkbox`) abzuhaken oder einen Knopf anzuklicken.

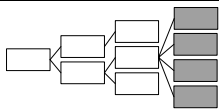
focus()

Funktion, um einem Element den Eingabefokus zu geben.

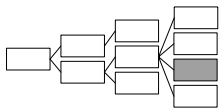
Ereignisse

onblur Eventhandler, der beim Verlust des Fokus auf das Element aktiviert wird.
onclick Wird aktiviert, wenn auf das Element per Maus geklickt wird (1.0,1.1).
onfocus Eventhandler, wird beim Einstellen des Fokus auf das Element aktiviert.

2.12.4 Element *siehe Input*

| | |
|-----------------------------------|--|
| <p>Element → Input</p> | <p>Zugriff <i>form.elements[index], form.inputFeldName</i></p>  |
|-----------------------------------|--|

2.12.5 FileUpload

| | | |
|---|---|--|
| <p>FileUpload <INPUT Type=File ..></p> | <p>Zugriff <i>form.elements[index] form.fileUploadName</i></p>  | |
| <p>Eigenschaften form, name, type, value</p> | <p>Methoden blur(), focus(), select()</p> | <p>Ereignisse onblur, onchange, onfocus</p> |

HTML-Syntax

```
<FORM Enctype="multipart/form-data" Method=POST ...>
  <INPUT Type=file
    Name=elementname
    Size=elementbreite
    Maxlength=maximalezeichenzahl
    onblur=eventhandler
    onchange=eventhandler
    onfocus=eventhandler
  >
  ...
</FORM>
```

Eigenschaften

form

Verweis auf das Formular, in dem dieses Element enthalten ist.

name

Name des Formularelementes.

type

Spezifiziert den Typ des Formularelementes.

value

Wert des Eingabefeldes. Der eingegebene Text legt einen Dateinamen fest. Der Wert kann weder in HTML noch mittels JavaScript (oder mit anderen Scriptsprachen wie VBScript) vorbesetzt werden.

Methoden

blur()

Funktion, um einem Element den Eingabefokus zu nehmen.

focus()

Funktion, um einem Element den Eingabefokus zu geben.

select()

Funktion, um Text in Eingabefeldern zu markieren.

Ereignisse

onblur Eventhandler, der beim Verlust des Fokus auf das Element aktiviert wird.

onchange Wird aktiviert bei Textelementen, wenn der Text geändert wird.

onfocus Eventhandler, wird beim Einstellen des Fokus auf das Element aktiviert.

2.12.6 Form

| | | |
|--|---|--|
| Form <FORM ...> | Zugriff <i>document.formName,</i> <i>document.forms[formIndex]</i> | |
| Eigenschaften: action, elements[], elements.length, encoding, length, method, name, target | Methoden reset(), submit() | Ereignisse onreset, onsubmit |

HTML-Syntax

```

<FORM Name=formularname
      Method=GET|POST
      Action=url
      Target>windowname
      Enctype=verschlüsselungsart
      onsubmit=eventhandler
      onreset=eventhandler
>
  Texte und Formularelemente
</FORM>

```

Eigenschaften

action

Aktiviert die angegebene URL bei Ausführung des Formulars (Submit).

elements []

Array mit den Elementen eines Formulars.

elements.length

Anzahl der Elemente eines Formulars.

encoding

Bezieht sich auf das `enctype`-Attribut und ist bei Weglassen auf „application/x-www-form-urlencoded“ gesetzt. Die Eigenschaft legt fest, wie der Anforderungsdatenblock bzw. der Query-String kodiert sind.

length

Die Anzahl der Formularelemente, siehe `elements.length`.

method

Legt die Übertragungsmethode für die Formulardaten fest. Die Daten werden im Anforderungsdatenblock bei `POST` übertragen und als Query-String in der URL bei `GET`.

name

Der Name des Formulars.

target

Legt fest, in welchem Window oder welchen Frame das Resultat der Formularaktion angezeigt wird.

Methoden**reset()**

Setzt alle Eingaben auf den Anfangszustand zurück.

submit()

Löst die Formularaktion aus.

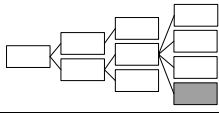
Ereignisse**onreset**

Wird bei Drücken des Reset-Knopfes ausgelöst.

onsubmit

Wird bei Drücken des Submit-Knopfes ausgelöst.

2.12.7 Hidden

| | | |
|---|-----------------|--|
| Hidden <INPUT Type=Hidden ..> | Zugriff |  |
| Eigenschaften form, name, type, value | Methoden | Ereignisse |

HTML-Syntax

```
<FORM ...>
  <INPUT Type=hidden
        Name=elementname
        Value=wert
  >
  ...
</FORM>
```

Eigenschaften

form

Verweis auf das Formular, in dem dieses Element enthalten ist.

name

Name des Formularelementes.

type

Spezifiziert den Typ des Formularelementes.

value

Dieser Wert wird mit dem Formular verschickt.

2.12.8 Option

| | | |
|--|---|--|
| Option → Select | Zugriff <i>select.options[index]</i> | |
| Eigenschaften defaultSelected, index, selected, text, value | Konstruktor new Option(<i>text, wert,</i> <i>standardwahl, gewaehlt</i>) | |

HTML-Syntax

```
<SELECT ...>
  <OPTION Value=wert Selected> text
  ...
</SELECT>
```

Wird das value-Attribut weggelassen, so tritt text an die Stelle von wert.

Konstruktor

new Option(text, wert, standardwahl, gewaehlt)

Dynamische Erzeugung eines Optionsobjektes (siehe auch Select-Objekt). Zu den Parametern siehe die folgenden Eigenschaften.

Eigenschaften

defaultSelected

Dies ist die Wahlvorgabe für den Aufbau des Optionsmenüs.

index

Index der Option in diesem Optionsarray.

selected

Diese Option wurde vom Nutzer ausgewählt.

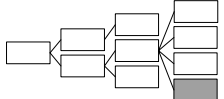
text

Der Text, der diese Option beschreibt. Wenn das value-Attribut nicht verwendet wurde, ist dies zugleich dessen Wert.

value

Der Wert des value-Attributs.

2.12.9 Password

| | | |
|---|--|--|
| Password Zugriff form.name form.elements[index] |  | |
| Eigenschaften form, name, type, value | Methoden blur(), focus(), select() | Ereignisse onblur, onchange, onfocus |

HTML-Syntax

```

<FORM ...>
  <INPUT Type=password
        Name=elementname
        Value=wert
        Size=elementbreite
  >
  ...
</FORM>
    
```

Eigenschaften

form

Verweis auf das Formular, in dem dieses Element enthalten ist.

name

Name des Formularelementes.

type

Spezifiziert den Typ des Formularelementes.

value

Der Wert des Eingabefeldes ist das vom Benutzer eingegebene Passwort.

Methoden**blur ()**

Funktion, um einem Element den Eingabefokus zu nehmen.

focus ()

Funktion, um einem Element den Eingabefokus zu geben.

select ()

Funktion, um Text in Eingabefeldern zu markieren.

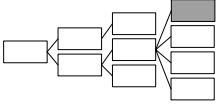
Ereignisse

onblur Eventhandler, der beim Verlust des Fokus auf das Element aktiviert wird.

onchange Wird aktiviert bei Textelementen, wenn der Text geändert wird.

onfocus Eventhandler, wird beim Einstellen des Fokus auf das Element aktiviert.

2.12.10 Radio

| | | |
|---|--|---|
| Radio <INPUT Type=radio ...> | Zugriff <i>form.elements[index]</i> <i>form.radioName /</i> <i>form.radio[index]</i> <i>form.radio.length</i> |  |
| Eigenschaften checked, defaultChecked, form, name, type, value | Methoden blur(), click(), focus() | Ereignisse onblur, onclick, onfocus |

HTML-Syntax

```

<FORM ... >
  <INPUT Type=radio
    Name=elementodergruppenname
    Value=wert
    Checked
    onblur=eventhandler
    onclick=eventhandler
    onfocus=eventhandler
  >
  ...
</FORM>

```

Eigenschaften

checked

Dieser Radioknopf ist gedrückt.

defaultChecked

Dieser Radioknopf ist per HTML-Attribut gedrückt.

form

Verweis auf das Formular, in dem dieses Element enthalten ist.

name

Name des Formularelementes.

type

Spezifiziert den Typ des Formularelementes.

value

Dieser Wert wird mit dem Formular verschickt.

Methoden

blur()

Funktion, um einem Element den Eingabefokus zu nehmen.

click()

Funktion, um ein Kästchen (`radio`, `checkbox`) abzuhaken oder einen Knopf anzuklicken.

focus()

Funktion, um einem Element den Eingabefokus zu geben.

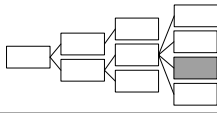
Ereignisse

onblur Eventhandler, der beim Verlust des Fokus auf das Element aktiviert wird.

onclick Wird aktiviert, wenn auf das Element per Maus geklickt wird (1.0,1.1).

onfocus Eventhandler, wird beim Einstellen des Fokus auf das Element aktiviert.

2.12.11 Reset

| | | |
|---|---|--|
| Reset <INPUT Type=reset ...> | Zugriff <i>form.name</i> <i>form.elements[index]</i> |  |
| Eigenschaften form, name, type, value | Methoden blur(), click(), focus() | Ereignisse onblur, onclick, onfocus |

HTML-Syntax

```
<FORM ... >
  <INPUT Type=reset
        Name=elementname
        Value=aufschrift
        onblur=eventhandler
        onclick=eventhandler
        onfocus=eventhandler
  >
  ...
</FORM>
```

Eigenschaften

form

Verweis auf das Formular, in dem dieses Element enthalten ist.

name

Name des Formularelementes.

type

Spezifiziert den Typ des Formularelementes.

value

Der Wert des Eingabefeldes ist das vom Benutzer eingegebene Passwort.

Methoden

blur ()

Funktion, um einem Element den Eingabefokus zu nehmen.

click ()

Funktion, um ein Kästchen (radio, checkbox) abzuhaken oder einen Knopf anzuklicken.

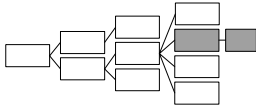
focus ()

Funktion, um einem Element den Eingabefokus zu geben.

Ereignisse

| | |
|----------------|--|
| onblur | Eventhandler, der beim Verlust des Fokus auf das Element aktiviert wird. |
| onclick | Wird aktiviert, wenn auf das Element per Maus geklickt wird (1.0,1.1). |
| onfocus | Eventhandler, wird beim Einstellen des Fokus auf das Element aktiviert. |

2.12.12 Select

| | | |
|--|---|--|
| Select <code><SELECT ...></code> <code><OPTION ...>...</code> <code>...</code> <code></SELECT></code> | Zugriff <code>form.elementName,</code> <code>form.elements[index]</code> |  |
| Eigenschaften <code>form, length, name, options[],</code> <code>selectedIndex, type</code> | Methoden <code>blur(), focus()</code> | Ereignisse <code>onblur, onchange, onfocus</code> |

HTML-Syntax

```

<FORM ... >
  <SELECT Name=elementname
    Size=sichtbareoptionen
    Multiple
    onblur=eventhandler
    onchange=eventhandler
    onfocus=eventhandler
  >
    <OPTION Value=wert Selected>beschriftung
    weitere Optionen
  </SELECT>
  ...
</FORM>

```

Eigenschaften

form

Verweis auf das Formular, in dem dieses Element enthalten ist.

length

Anzahl der Optionen des Auswahlnüs.

name

Name des Formularelementes.

options []

Array von Optionen des Auswahlmenüs.

selectedIndex

Aktuell ausgewählte Option.

type

Spezifiziert den Typ des Formularelementes.

Methoden

blur()

Funktion, um einem Element den Eingabefokus zu nehmen.

focus()

Funktion, um einem Element den Eingabefokus zu geben.

Ereignisse

onblur Eventhandler, der beim Verlust des Fokus auf das Element aktiviert wird.

onchange Wird aktiviert, wenn eine neue Option ausgewählt wurde.

onfocus Eventhandler, wird beim Einstellen des Fokus auf das Element aktiviert.

2.12.13 Submit

| | | |
|---|--|---|
| Submit <INPUT Type=submit ...> | Zugriff <i>form.name,</i> <i>form.elements[index]</i> | |
| Eigenschaften form, name, type, value | Methoden blur(), click(), focus() | Ereignisse onblur, onclick, onfocus |

HTML-Syntax

```

<FORM ... >
  <INPUT Type=submit
    Name=elementname
    Value=aufschrift
    onblur=eventhandler
    onclick=eventhandler
    onfocus=eventhandler
  >
  ...
</FORM>

```

form

Verweis auf das Formular, in dem dieses Element enthalten ist.

name

Name des Formularelementes.

type

Spezifiziert den Typ des Formularelementes.

value

Der Wert des Eingabefeldes ist das vom Benutzer eingegebene Passwort.

Methoden

blur ()

Funktion, um einem Element den Eingabefokus zu nehmen.

click ()

Funktion, um ein Kästchen (`radio`, `checkbox`) abzuheben oder einen Knopf anzuklicken.

focus ()

Funktion, um einem Element den Eingabefokus zu geben.

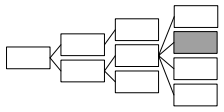
Ereignisse

onblur Eventhandler, der beim Verlust des Fokus auf das Element aktiviert wird.

onclick Wird aktiviert, wenn auf das Element per Maus geklickt wird (1.0,1.1).

onfocus Eventhandler, wird beim Einstellen des Fokus auf das Element aktiviert.

2.12.14 Text

| | | |
|--|--|--|
| Text <code><INPUT Type=text ...></code> | Zugriff <code>form.name,</code> <code>form.elements[index]</code> |  |
| Eigenschaften <code>defaultValue, form, name,</code> <code>type, value</code> | Methoden <code>blur(), focus(), select()</code> | Ereignisse <code>onblur, onchange, onfocus</code> |

HTML-Syntax

```

<FORM ... >
  <INPUT Type=text
    Name=elementname
    Value=wert
    Size=elementbreite
    Maxlength=erlaubtezeichnzahl
    onblur=eventhandler
    onchange=eventhandler
    onfocus=eventhandler
  >
  ...
</FORM>

```

Eigenschaften

defaultValue

Der mittels des Attributs Value definierte Vorgabewert des Eingabefeldes.

form

Verweis auf das Formular, in dem dieses Element enthalten ist.

name

Name des Formularelementes.

type

Spezifiziert den Typ des Formularelementes.

value

Der Wert des Eingabefeldes ist der vom Benutzer eingegebenen Text.

Methoden**blur()**

Funktion, um einem Element den Eingabefokus zu nehmen.

focus()

Funktion, um einem Element den Eingabefokus zu geben.

select()

Funktion, um Text in Eingabefeldern zu markieren.

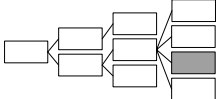
Ereignisse

onblur Eventhandler, der beim Verlust des Fokus auf das Element aktiviert wird.

onclick Wird aktiviert, wenn auf das Element per Maus geklickt wird (1.0,1.1).

onfocus Eventhandler, wird beim Einstellen des Fokus auf das Element aktiviert.

2.12.15 Textarea - Mehrzeiliges Editierfenster

| | | |
|--|--|---|
| Textarea <TEXTAREA ...> ... </TEXTAREA> | Zugriff <i>form.name,</i> <i>form.elements[index]</i> |  |
| Eigenschaften defaultValue, form, name, type, value | Methoden blur(), focus(), select() | Ereignisse onblur, onchange, onfocus |

HTML-Syntax

```
<FORM ... >
  <TEXTAREA Name=elementname
    Rows=elementbreite
    Cols=erlaubtezeichnzahl
    Wrap=off|physical|virtual
    Readonly
    onblur=eventhandler
    onchange=eventhandler
```

```
        onfocus=eventhandler
    >
    textvorbelegung
</TEXTAREA>
...

```

Eigenschaften

defaultValue

Der zwischen den Markups stehende Text als Vorgabewert des Editierfensters.

form

Referenz auf das Formular, in dem das INPUT-Element enthalten ist.

name

Der Name eines Formularelementes.

type

Der Typ des Formularelementes als String.

value

Wert des Elementes. Bei Button, submit und Reset die Beschriftung des Knopfes, bei Eingabeelementen der eingegebene bzw. angezeigte Text etc.

Methoden

blur()

Funktion, um einem Element den Eingabefokus zu nehmen.

focus()

Funktion, um einem Element den Eingabefokus zu geben.

select()

Funktion, um Text in Eingabefeldern zu markieren.

Ereignisse

onblur Eventhandler, der beim Verlust des Fokus auf das Element aktiviert wird.

onchange Wird aktiviert bei Textelementen, wenn der Text geändert wird.

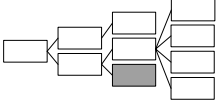
onfocus Eventhandler, wird beim Einstellen des Fokus auf das Element aktiviert.

2.13 Verschiedene

2.13.1 Layer - Dynamisch positionierbare Elemente

Layer-Objekte sind Netscape-spezifisch, kommen also nur beim Navigator vor. Sie können mit dem LAYER-Markup und dem Layer-Konstruktor `Layer()` definiert bzw. erzeugt

werden. Sie sind aber auch mittelbar durch die Verwendung von Stilvorlagen (CSS-P) und von Stilattributen in HTML-Elementen erzeugbar.

| | | |
|---|---|---|
| Layer (nur Netscape 4) | Zugriff <i>document.layers[index]</i> |  |
| Eigenschaften NN: above, background, below, bgColor, clip.bottom, clip.height, clip.left, clip.right, clip.top, clip.width, document, hidden, layers[], left, name, pageX, pageY, parentLayer, siblingAbove, siblingBelow, src, top, visibility, window, x, y, zIndex | Methoden NN: captureEvents(), handleEvent(), load(), moveAbove(), moveBelow(), moveBy(), moveTo(), moveToAbsolute(), offset(), releaseEvents(), resizeBy(), resizeTo(), routeEvent() Konstruktor new Layer(breite, parent) | Ereignisse onblur, onfocus, onload, onmouseout, onmouseover |

HTML-Syntax

| | |
|---|---|
| <pre><LAYER ID=layername Left=wert Top=wert PageX=wert PageY=wert Width=wert Height=wert Src=URL Clip="x,y,w,h" Zindex=wert Visibility=show hide inherit Bgcolor=farbe Background=url onblur=eventhandler onfocus=eventhandler onload=eventhandler onmouseout=eventhandler onmouseover=eventhandler ></pre> | <p>Position relativ zum linken Rand Position relativ zum oberen Rand Position relativ zum Dokument</p> <p>Größe des Layers</p> <p>URL für den Layer-Inhalt Clipping-Rechteck Layer-Nummerierung</p> <p>Hintergrundfarbe Hintergrundbild</p> |
|---|---|

Konstruktor

new Layer(breite, parent)

Ein neuer Layer mit einer Breite von *breite* Pixeln und an das Objekt *parent* gebunden.

Eigenschaften

| | | |
|--|--|-----|
| above | Das Layer-Objekt über diesem. | |
| background | Das Hintergrundbild dieses Layers. | NN4 |
| below | Das Layer-Objekt unterhalb von diesem. | NN4 |
| bgColor | Die Hintergrundfarbe. | NN4 |
| clip.bottom, clip.height, clip.left, clip.right, clip.top, clip.width | Die Dimensionen der Clipping-Fläche des Layers. | |
| document | Das Dokumentenobjekt dieses Layers. | NN4 |
| hidden | Zeigt an, ob der Layer verborgen ist. | NN4 |
| layers [] | Ein Array ineinander geschachtelter Layer. | NN4 |
| left, top | Koordinaten relativ zu dem Layer bzw. Dokument, der diesen Layer enthält. | NN4 |
| name | Name des Layer-Objektes. | NN4 |
| pageX, pageY | Koordinaten des Layers relativ zum Dokument des Browserfensters (Wurzeldokument). | NN4 |
| parentLayer | Zeiger auf das übergeordnete Layer- oder Dokumentenobjekt. | NN4 |
| siblingAbove | Verweis auf den Layer der gleichen Ebene, der diesen überlagert. | NN4 |
| siblingBelow | Verweis auf den Layer der gleichen Ebene, der diesem untergeordnet ist. | NN4 |
| src | Die URL für den Inhalt des Layers. | NN4 |
| visibility | Mit „show“ wird der Layer angezeigt, mit „hide“ verborgen und mit „inherit“ so angezeigt, wie es der übergeordnete Layer wird. | NN4 |

| | | |
|---------------|---|------------|
| window | Verweis auf das Window-Objekt, das diesen Layer enthält. | NN4 |
| x, y | Koordinaten des Layers relativ zum übergeordneten Layer oder Dokument. | NN4 |
| zIndex | Laufende Nummer des Layers in der Abfolge ineinander <i>geschachtelter</i> Layer. | NN4 |

Methoden

| | | |
|--------------------------------------|---|-------------|
| captureEvents (ereignismaske) | Gibt mittels Ereignismaske die Ereignisse an, die beim Auftreten in Dokumenten (document-Objekt) und Windows (window-Objekte) auftreten können. | NN4 |
| handleEvent (ereignis) | Übergibt ds Ereignisobjekt <i>ereignis</i> an einen passenden Eventhandler des aktuellen Objektes, falls vorhanden. | NN4 |
| load(inhalt, breite) | Ändere Inhalt und Breite des Layers. | NN\$ |
| moveAbove (layer) | Diesen Layer <i>über</i> den Layer <i>layer</i> legen | NN4 |
| moveBelow (layer) | Diesen Layer <i>unter</i> den Layer <i>layer</i> legen | NN4 |
| moveBy (dx, dy) | Fenster um <i>dx</i> in X-Richtung und <i>dy</i> in Y-Richtung verschieben. | NN4 |
| moveTo (x, y) | Versetzt das Fenster so, dass die linke obere Ecke des Fensters die Koordinatenposition <i>x, y</i> einnimmt. | NN4 |
| moveToAbsolute (x, y) | Verschiebt Layer zu den Koordinaten <i>x, y</i> relativ zum Wurzeldokument. | NN4 |
| offset (dx, dy) | Siehe <code>moveBy()</code> . | NN4 |
| releaseEvents (ereignismaske) | Beendet das Abfangen von Ereignissen. | NN4 |
| resizeBy (dbreite, dhoehe) | Verändert die Größe des Layers um <i>dbreite</i> horizontal und <i>dhoehe</i> vertikal (Werte in Pixeln). | NN4 |
| resizeTo (breite, hoehe) | Verändert die Größe des Layers auf <i>breite</i> horizontal und <i>hoehe</i> vertikal (Werte in Pixeln). | NN4 |

routeEvent (ereignis)

NN4

Gibt das Ereignis `ereignis` an den nächsten Eventhandler weiter.**2.13.2 Screen - Informationen über den Bildschirm**

| Screen | Zugriff <i>screen.eigenschaft</i> | |
|--|--------------------------------------|-------------------|
| Eigenschaften availHeight, availLeft, availTop, availWidth, colorDepth, height, pixelDepth, width | Methoden | Ereignisse |

Eigenschaften**availHeight**

Verfügbare Bildschirmhöhe in Pixeln.

availLeft

Erstes verfügbares Pixel horizontal.

NN4

availTop

Erstes verfügbares Pixel vertikal.

NN4

availWidth

Verfügbare Bildschirmbreite in Pixeln.

colorDepth

Tiefe der Farbpalette.

height

Bildschirmhöhe in Pixeln.

pixelDepth

Farbtiefe des Bildschirms.

NN4

width

Bildschirmbreite in Pixeln.

2.13.3 Style

| | | |
|-------------------------------------|---|-------------------|
| Style | Zugriff NN: <i>document.classes.class.markup</i> <i>document.ids.elementname</i> <i>document.tags.mrkupname</i> <i>document.contextual()</i> IE: <i>htmlElement.style</i> | |
| Eigenschaften siehe unten | Methoden NN: <i>borderWidth(), mrgins(),</i> <i>padding()</i> | Ereignisse |

Siehe ab Seite 121 im Buch.