

JavaScript

Objekte

Die nativen Objekte in JavaScript

- Arrays
- Datumsobjekte
- Wahrheitsobjekte
- Zeichenketten
- Bildobjekte
- Funktionsobjekte
- Reguläre Ausdrücke

Das DOM-Konzept

Objekte und Hierarchie des DOM

- Zugriff auf Elemente des HTML-Dokumentes
- DOM-Objekte gehören nicht zu JavaScript
- Objektschnittstelle DOM wird vom W3C standardisiert
- ist aus beliebigen Sprachen zugänglich (wenn diese im Browser zugänglich sind)

Entwicklung des DOM

- von Netscape entwickelt
- 1997 vom W3C übernommen
- verschiedene Level – unterschiedlich implementiert
- moderne Browser => DOM 3
- Mit Entwicklung von XML wurde DOM modifiziert und auf baumartig strukturierte Dokumente anwendbar gemacht.
- Abbildung der Struktur im Arbeitsspeicher
- Zugriff über API

Die DOM-Objekte

- all
- anchor
- applet
- document
- event
- form
- frame
- history
- image
- layer
- link
- location
- mimeType
- navigator
- node
- plugin
- screen
- window

window – wichtige Eigenschaften

- document
- location
- history
- status
- *frames*
- *self, parent, top*

Das node-Objekt

Nummer	Knotentyp	
1	Elementknoten	Für JavaScript interessant
2	Attributknoten	
3	Textknoten	
4	CDATA-Bereich	Im Wesentlichen für XML interessant, Ausnahme Typ 9
5	Entity-Referenz	
6	Entity	
7	Process Instruction (PI)	
8	<i>Kommentar</i>	
9	Dokument	
10	Dokumenttyp	
11	Dokumentfragment	
12	Notation	

DOM-Knoten

- Das aktuelle **HTML-Dokument** ist ein **Dokumentknoten**
- Jedes **HTML-Element** ist ein **Elementknoten**
- **Texte** in HTML-Elementen sind **Textknoten**
- Jedes **HTML-Attribut** ist ein **Attributknoten**
- **Kommentare** sind **Kommentarknoten**

Node-Objekte erzeugen

- `createElement`
erzeugt ein Element (`<p>`, `<h1>`, `<div>`, ...)
- `createTextNode`
erzeugt einen Textknoten (Inhalt eines Elementes)
- `createAttribute`
erzeugt ein HTML-Attribut (`src`, `width`, `style`, ...)

HTML-Elemente als Unterobjekte von document

