

Auf einen Blick

| | |
|--|-----|
| Vorwort | 17 |
| Kapitel 1 Einleitung | 19 |
| Kapitel 2 Grundlagen und erste Beispiele | 37 |
| Kapitel 3 Vorbereitende Basics | 53 |
| Kapitel 4 Die Versionen von JavaScript und die Einbindung in Webseiten | 105 |
| Kapitel 5 Elementare JavaScript-Grundstrukturen | 119 |
| Kapitel 6 Fehlervermeidung, Fehlersuche und Testverfahren | 207 |
| Kapitel 7 JavaScript und Objekte | 209 |
| Kapitel 8 Ausnahmebehandlung, Ereignisbehandlung und das event-Objekt | 327 |
| Kapitel 9 DHTML und Animationen | 357 |
| Kapitel 10 AJAX und das Web 2.0 | 381 |
| Kapitel 11 Schutzmaßnahmen und Optimierung | 423 |
| Kapitel 12 Zugriff auf Webformulare | 431 |
| Kapitel 13 Erweiterte JavaScript-Techniken | 453 |
| Anhang A | 531 |
| Stichwortverzeichnis | 533 |

Inhaltsverzeichnis

| | |
|---|----|
| Vorwort | 17 |
| Kapitel 1 Einleitung | 19 |
| 1.1 Über das Buch | 20 |
| 1.1.1 An wen sich das Buch wendet | 20 |
| 1.1.2 Wie dieses Buch organisiert ist | 22 |
| 1.1.3 Schreibkonventionen | 22 |
| 1.1.4 Was Sie in dem Buch lernen | 23 |
| 1.2 Was Sie unbedingt haben sollten | 25 |
| 1.2.1 Die Hardware | 25 |
| 1.2.2 Die Software | 25 |
| 1.3 Firefox als Zentrale der Webentwicklung | 32 |
| 1.3.1 Interessante Add-ons für Firefox | 33 |
| 1.4 Andere Browser zur Entwicklungszentrale aufbauen? | 34 |
| 1.4.1 Internet Explorer | 34 |
| 1.4.2 Chrome/Iron | 34 |
| 1.4.3 Opera | 35 |
| 1.5 Zusammenfassung | 35 |
| Kapitel 2 Grundlagen und erste Beispiele | 37 |
| 2.1 Erste JavaScript-Beispiele | 37 |
| 2.1.1 Ein einfaches Mitteilungsfenster | 37 |
| 2.1.2 Schreiben eines angepassten Aktualisierungsdatums | 40 |
| 2.1.3 Entgegennahme einer Benutzereingabe | 41 |
| 2.2 Die Besonderheit bei der Webprogrammierung | 42 |
| 2.2.1 Kompilierung versus Interpretation ... | 42 |
| 2.2.2 Unterschiedliche Plattformen und Interpretation | 43 |
| 2.3 Ein erster Überblick über wichtige Techniken der Webprogrammierung | 44 |
| 2.3.1 HTML & XHTML | 45 |
| 2.3.2 Skriptsprachen im Web | 45 |
| 2.3.3 Java | 47 |
| 2.3.4 ActiveX | 48 |
| 2.3.5 Style Sheets | 48 |

| | | |
|------------------|---|-----------|
| 2.3.6 | DHTML | 49 |
| 2.3.7 | XML | 50 |
| 2.3.8 | AJAX und das Web 2.0 | 50 |
| 2.3.9 | Cookies | 51 |
| 2.3.10 | Grafik und Multimedia | 52 |
| 2.4 | Zusammenfassung | 52 |
| Kapitel 3 | Vorbereitende Basics | 53 |
| 3.1 | Der Aufbau von (X)HTML-Dateien | 54 |
| 3.1.1 | Das Prinzip der Fehlertoleranz | 54 |
| 3.1.2 | Steueranweisungen | 57 |
| 3.1.3 | Attribute | 58 |
| 3.1.4 | Strukturierung und Gestaltung mit HTML | 59 |
| 3.1.5 | Das Grundgerüst einer Webseite | 60 |
| 3.1.6 | Der Header | 61 |
| 3.1.7 | Der Körper von Webseiten | 63 |
| 3.2 | XML-Grundlagen | 63 |
| 3.2.1 | XML-Elemente | 64 |
| 3.2.2 | Die Syntax eines XML-Dokuments | 65 |
| 3.2.3 | Gültige XML-Dokumente | 66 |
| 3.3 | HTML versus XHTML | 67 |
| 3.4 | Ein erster Blick auf das DOM-Konzept | 69 |
| 3.5 | HTML5 und DOM5 | 70 |
| 3.5.1 | Das neue Vokabular | 72 |
| 3.5.2 | Das neue DOM5 und die MIME-Typen von HTML5 | 92 |
| 3.5.3 | Das Namensraumkonzept von HTML5 | 93 |
| 3.5.4 | Die erwartete Darstellung | 93 |
| 3.5.5 | Der Browserkontext und die Kommunikation zwischen verschiedenen Kontexten | 93 |
| 3.5.6 | HTML Microdata | 94 |
| 3.5.7 | Web Workers | 94 |
| 3.5.8 | Canvas-Objekte und die Einbindung von SVG | 95 |
| 3.6 | Die Gültigkeit von Webseiten testen | 96 |
| 3.7 | Grundlagen zu Style Sheets | 97 |
| 3.7.1 | Die Verwendung von Style Sheets in einer Webseite | 97 |
| 3.7.2 | Die konkrete Syntax von CSS-DeklARATIONEN | 99 |
| 3.7.3 | CSS3 | 99 |
| 3.8 | Zusammenfassung | 104 |

| | | |
|------------------|--|-----|
| Kapitel 4 | Die Versionen von JavaScript und die Einbindung in Webseiten | 105 |
| 4.1 | Die Versionszyklen von JavaScript und ECMAScript | 105 |
| 4.2 | JavaScript in Webseiten einbinden | 107 |
| 4.2.1 | Verbindung 1 – die Notation eines Skript-Containers in der Webseite | 108 |
| 4.2.2 | Verbindung 2 – die Verwendung von externen JavaScript-Dateien | 110 |
| 4.2.3 | Verbindung 3 – die Inline-Referenz | 112 |
| 4.3 | Der gemeinsame Namensraum | 113 |
| 4.4 | Der <noscript>-Container | 113 |
| 4.5 | Kann man testen, ob bei einem Browser JavaScript aktiviert ist? | 113 |
| 4.6 | Kann man testen, welche JavaScript-Version von einem Browser unterstützt wird? | 116 |
| 4.7 | Zusammenfassung | 117 |
| Kapitel 5 | Elementare JavaScript-Grundstrukturen | 119 |
| 5.1 | Token und Parser | 120 |
| 5.1.1 | Token | 120 |
| 5.1.2 | Parser | 120 |
| 5.2 | Kommentare | 120 |
| 5.2.1 | Mehrzeilige Kommentare in JavaScript | 121 |
| 5.2.2 | Einzeilige Kommentare in JavaScript | 121 |
| 5.3 | Schlüsselwörter in JavaScript | 121 |
| 5.4 | Datentypen, Variablen und Literale | 123 |
| 5.4.1 | Datentypen | 123 |
| 5.4.2 | Literale und Steuerzeichen bei Zeichenketten | 126 |
| 5.4.3 | Variablen anlegen | 128 |
| 5.4.4 | Namensregeln für Variablen | 130 |
| 5.5 | Anweisungen | 131 |
| 5.5.1 | Ausdrücke | 131 |
| 5.5.2 | Blöcke | 132 |
| 5.5.3 | Operatoren und Operanden | 132 |
| 5.6 | Kontrollstrukturen in JavaScript | 147 |
| 5.6.1 | Die if-Bedingung | 148 |
| 5.6.2 | Die switch-Fallunterscheidung | 149 |
| 5.6.3 | Die while-Schleife | 153 |
| 5.6.4 | Die do-while-Schleife | 155 |
| 5.6.5 | Die for-Schleife | 156 |
| 5.6.6 | Sprunganweisungen | 157 |

| | | |
|--------|--|-----|
| 5.7 | Datenfelder | 160 |
| 5.7.1 | Ein Array erzeugen | 160 |
| 5.7.2 | Zugriff auf Array-Elemente | 162 |
| 5.7.3 | Mit Schleifen auf Array-Elemente zugreifen | 165 |
| 5.8 | Funktionen, Prozeduren und Methoden | 168 |
| 5.8.1 | Funktion, Prozedur, Methode | 169 |
| 5.8.2 | Verwendung von vorgefertigten Funktionen | 169 |
| 5.8.3 | Die Definition eigener Funktionen | 173 |
| 5.8.4 | Der Aufruf einer Funktion | 174 |
| 5.8.5 | Lokale Variablen in Funktionen | 178 |
| 5.8.6 | Flexible Anzahl an Funktionsargu- menten und das Token arguments | 181 |
| 5.8.7 | Rekursive Funktionsaufrufe | 186 |
| 5.8.8 | Funktionsreferenzen | 187 |
| 5.8.9 | Benannte versus anonyme Funktionen | 189 |
| 5.8.10 | Innere Funktionen – Closures | 190 |
| 5.8.11 | Callbacks | 192 |
| 5.9 | Aufruf einer JavaScript-Funktion per Eventhandler | 194 |
| 5.9.1 | Was ist ein Ereignis? | 194 |
| 5.9.2 | Ein Mechanismus zur Reaktion auf Ereignisse | 195 |
| 5.9.3 | Verschiedene Ereignisse | 196 |
| 5.10 | Zusammenfassung | 205 |



| | | |
|------------------|--|------------|
| Kapitel 6 | Fehlervermeidung, Fehlersuche und Testverfahren | 207 |
|------------------|--|------------|

| | | |
|------------------|-------------------------------------|------------|
| Kapitel 7 | JavaScript und Objekte | 209 |
|------------------|-------------------------------------|------------|

| | | |
|-------|---|-----|
| 7.1 | Was sind Objekte? | 209 |
| 7.1.1 | Wie werden Methoden und Eigen- schaften in JavaScript verwendet? | 210 |
| 7.1.2 | Die grundsätzliche Definition der ob- jektorientierten Programmierung | 210 |
| 7.2 | Klassen, Instanzen & Vererbung | 211 |
| 7.2.1 | Vererbung | 211 |
| 7.3 | JavaScript-Objekte erzeugen | 212 |
| 7.3.1 | Test auf die Existenz eines Objekts | 215 |
| 7.4 | Klassenmethoden und Klasseigenschaften verwenden | 215 |
| 7.4.1 | Nutzen von Klasselementen aus Math | 216 |
| 7.4.2 | Objekte und Klasselemente von Number | 218 |

| | | |
|--------|---|-----|
| 7.5 | Native Objekte in JavaScript | 220 |
| 7.5.1 | Datenfelder bzw. Arrays | 221 |
| 7.5.2 | Objekte vom Typ Boolean | 226 |
| 7.5.3 | Datumsoperationen mit Date | 226 |
| 7.5.4 | Stringoperationen | 235 |
| 7.5.5 | Reguläre Ausdrücke | 239 |
| 7.5.6 | Objekte vom Typ Function | 245 |
| 7.5.7 | Native Bildobjekte | 246 |
| 7.6 | Das DOM – das Mysterium der JavaScript-Welt | 246 |
| 7.6.1 | Was ist das DOM-Konzept? | 246 |
| 7.6.2 | Grundsätzliches zum Weg durch das DOM | 247 |
| 7.7 | Die verfügbaren DOM-Objekte | 250 |
| 7.7.1 | Das Konzept der Objektfelder | 252 |
| 7.8 | Generelle Zugriffsmöglichkeiten auf DOM-Objekte | 253 |
| 7.8.1 | Der Zugriff über einen Namen – getElementByName() | 253 |
| 7.8.2 | Zugriff über eine ID – getElementById() | 255 |
| 7.8.3 | Zugriff über den Elementnamen – getElementsByTagName() | 256 |
| 7.8.4 | Zugriff über die Klasse | 257 |
| 7.8.5 | Der kurze Dienstweg – this | 257 |
| 7.9 | Zugriff auf Inhalte von Elementen in der Webseite | 259 |
| 7.9.1 | Zugriff auf Textinhalte | 259 |
| 7.9.2 | Zugriff auf Formularinhalte | 260 |
| 7.9.3 | Zugriff auf klassische HTML-Attribute | 260 |
| 7.10 | Das Objekt document | 263 |
| 7.10.1 | Die Eigenschaften | 263 |
| 7.10.2 | Die Methoden von document | 268 |
| 7.11 | Das node-Objekt | 273 |
| 7.11.1 | Knotenarten | 273 |
| 7.11.2 | Die Eigenschaften eines node-Objekts | 275 |
| 7.11.3 | Die Methoden eines node-Objekts | 280 |
| 7.12 | Fenstertechniken | 284 |
| 7.12.1 | Der Toplevel und die reservierten Fensteramen | 285 |
| 7.12.2 | Die Eigenschaften und Methoden des window-Objekts | 285 |
| 7.12.3 | Anwendungen mit Fenstern | 290 |
| 7.12.4 | Frames, IFrames und das Objektfeld frames | 309 |
| 7.13 | Browsersauswertung | 317 |
| 7.14 | Das Objekt navigator | 317 |

| | | |
|------------------|--|------------|
| 7.15 | Die Browserkennung | 318 |
| 7.15.1 | Suche nach spezifischen Informationen in navigator.appName, navigator.appVersion und navigator.userAgent | 319 |
| 7.15.2 | Indirekte Identifizierung eines Browsers unter Ausnutzung eines Mangels .. | 320 |
| 7.15.3 | Die Spracheinstellung eines Browsers .. | 321 |
| 7.15.4 | Eine Browserweiche erstellen | 321 |
| 7.16 | Zugriff auf Bildschirminformationen eines Besuchers | 322 |
| 7.17 | Zusammenfassung | 325 |
| Kapitel 8 | Ausnahmebehandlung, Ereignisbehandlung und das event-Objekt | 327 |
| 8.1 | Eventhandler in JavaScript | 328 |
| 8.1.1 | (X)HTML- versus JavaScript-Eventhandler | 328 |
| 8.1.2 | Verschiedene Ereignisse | 330 |
| 8.2 | Grundlagen zu Ereignisobjekten | 332 |
| 8.2.1 | Blubbern | 332 |
| 8.2.2 | Die Informationen im Ereignisobjekt .. | 333 |
| 8.3 | Das Ereignisobjekt verwenden | 333 |
| 8.3.1 | Das Netscape-Ereignismodell | 334 |
| 8.3.2 | Das Microsoft-Ereignismodell | 339 |
| 8.4 | Ausnahmebehandlung | 341 |
| 8.4.1 | Was ist eine Ausnahme? | 342 |
| 8.4.2 | Wozu ein Ausnahmekonzept? | 343 |
| 8.4.3 | Die verschiedene Ausnahmetypen | 345 |
| 8.4.4 | Das Auffangen einer Ausnahme | 345 |
| 8.4.5 | Mehrere Ausnahmen behandeln | 348 |
| 8.5 | Eine universelle Ereignisbehandlung mit Ausnahmebehandlung | 349 |
| 8.6 | Globale Ereignisbehandlung | 351 |
| 8.6.1 | Eine universelle Funktion zur Registrierung eines Listener | 353 |
| 8.7 | Zusammenfassung | 355 |
| Kapitel 9 | DHTML und Animationen | 357 |
| 9.1 | DHTML und Animation mit reinem JavaScript .. | 357 |
| 9.2 | DHTML und Animation mit reinem CSS | 358 |
| 9.3 | Verbinden von JavaScript und Style Sheets | 358 |
| 9.3.1 | Mit style eine Webseite formatieren ... | 359 |
| 9.3.2 | Eine Animation mit style | 361 |
| 9.3.3 | Ein Farbwähler für die Hintergrundfarbe | 362 |

| | | |
|-------------------|--|------------|
| 9.3.4 | Eine Animation durch Manipulation von Grafiken (style-Version) | 367 |
| 9.3.5 | Eine Interaktionskomponente dynamisieren | 368 |
| 9.4 | Verändern des Inhalts der Webseite | 370 |
| 9.4.1 | Ein aufklappbares Navigationsmenü .. | 371 |
| 9.4.2 | Eine Menüleiste | 375 |
| 9.5 | Zusammenfassung | 380 |
| Kapitel 10 | AJAX und das Web 2.0 | 381 |
| 10.1 | Die Beziehung zwischen Webserver und Browser | 382 |
| 10.1.1 | Der grundsätzliche Ablauf einer Kommunikation zwischen Herr und Diener – HTTP | 382 |
| 10.1.2 | Verbindungsorientierte Datenkommunikation | 384 |
| 10.1.3 | Zustandslose Datenkommunikation ... | 384 |
| 10.1.4 | Die HTTP-Details und die HTTP-Header | 384 |
| 10.1.5 | Der HTTP-Request | 385 |
| 10.1.6 | Der HTTP-Response | 386 |
| 10.1.7 | Die Methoden der Datenübertragung per HTTP | 386 |
| 10.2 | Die Meldungen eines Webserver | 388 |
| 10.3 | Das Datenformat | 389 |
| 10.3.1 | Klartextdatenformate | 390 |
| 10.4 | Was ist AJAX und was bezeichnet Web 2.0? ... | 392 |
| 10.4.1 | AJAX als technische Basis des Web 2.0 | 392 |
| 10.5 | Der grundsätzliche Ablauf einer AJAX-Anfrage und das XMLHttpRequest-Objekt (XHR) | 394 |
| 10.5.1 | Der grundsätzliche Ablauf | 394 |
| 10.5.2 | Ein XMLHttpRequest-Objekt erzeugen | 395 |
| 10.6 | Die Anforderung von Daten | 398 |
| 10.6.1 | Die Methoden eines XHR-Objekts | 398 |
| 10.6.2 | Die Eigenschaften eines XMLHttpRequest-Objekts | 400 |
| 10.7 | Verschiedene Datentypen vom Server per AJAX nachfordern | 401 |
| 10.7.1 | Eine reine Textdatei mit AJAX nachfordern | 402 |
| 10.7.2 | Übergabe von Werten an den Webserver | 408 |
| 10.7.3 | Nachfordern von XML-Daten | 411 |



| | | |
|-------------------|--|------------|
| 10.7.4 | JSON-Daten anfordern und verarbeiten | 417 |
| 10.8 | Statusinformationen für den Anwender | 420 |
| 10.8.1 | Die Auswertung des Statuscodes des Servers | 420 |
| 10.9 | Zusammenfassung | 421 |
| Kapitel 11 | Schutzmaßnahmen und Optimierung | 423 |
| 11.1 | Optimierung von JavaScripts | 423 |
| 11.1.1 | Interpreterprinzip = langsam – JIT = schnell | 424 |
| 11.1.2 | Rezepte | 425 |
| 11.2 | Schutzmaßnahmen | 428 |
| 11.3 | Zusammenfassung | 430 |
| Kapitel 12 | Zugriff auf Webformulare | 431 |
| 12.1 | Das Formularobjekt | 431 |
| 12.1.1 | Die Eigenschaften eines Formularobjekts | 432 |
| 12.1.2 | Die Methoden | 432 |
| 12.1.3 | Die Elemente in dem Webformular | 433 |
| 12.1.4 | Das Objektfeld options | 435 |
| 12.2 | Zugriff auf ein Webformular per JavaScript ... | 435 |
| 12.2.1 | Zugriff auf den Wert eines Formularelements | 436 |
| 12.3 | Verschicken und Zurücksetzen von Formulardaten | 444 |
| 12.3.1 | Verschicken von Formulardaten mit AJAX | 445 |
| 12.4 | Plausibilisieren von Formularen | 447 |
| 12.4.1 | Abbruch von Aktionen | 448 |
| 12.4.2 | Fehlerreaktion | 449 |
| 12.4.3 | Ein Beispiel mit Plausibilisierung | 449 |
| 12.5 | Zusammenfassung | 452 |
| Kapitel 13 | Erweiterte JavaScript-Techniken | 453 |
| 13.1 | Erweiterte Objekttechniken | 453 |
| 13.1.1 | Was zeichnet in JavaScript eine Methode gegenüber einer normalen Funktion aus? | 454 |
| 13.1.2 | Ein einzelnes Objekt erweitern | 454 |
| 13.1.3 | Was ist Prototyping? | 456 |
| 13.1.4 | Erstellen von Konstruktormethoden ... | 464 |
| 13.1.5 | Vererbung mit Prototyping simulieren | 468 |
| 13.1.6 | Datenkapselung | 470 |
| 13.2 | HTML5 – Multithreading über Web Workers ... | 473 |

| | | |
|-----------------------------------|---|-----|
| 13.3 | HTML5 – Zeichnen mit Canvas-Objekten | 476 |
| 13.3.1 | Das Canvas-Element | 477 |
| 13.3.2 | Der Grafikkontext | 477 |
| 13.3.3 | Farben und Farbverläufe | 479 |
| 13.3.4 | Rechtecke | 488 |
| 13.3.5 | Bilder und Bildausschnitte | 488 |
| 13.3.6 | Pfade ziehen – komplexe Formen | 492 |
| 13.3.7 | Texte zeichnen | 497 |
| 13.3.8 | Pattern und Schatten | 498 |
| 13.3.9 | Den Zustand eines Canvas speichern und restaurieren | 501 |
| 13.3.10 | Transformationen | 503 |
| 13.3.11 | Kompositionen und Clipping | 506 |
| 13.3.12 | Animationen mit Canvas-Objekten ... | 507 |
| 13.4 | Geodating mit HTML5 und mehr | 508 |
| 13.4.1 | Die gemeinsame Basis | 510 |
| 13.4.2 | Das Google Maps API | 511 |
| 13.4.3 | Die Geolokalisierung mit HTML5 | 516 |
| 13.5 | Daten im Browser speichern mit HTML5 und der Technik des Local Data Storage bzw. Session Data Storage | 519 |
| 13.5.1 | Offline Web-Applikationen mit dem Application Cache | 520 |
| 13.5.2 | Die Objekte localStorage und sessionStorage sowie deren Methoden und Eigenschaft | 521 |
| 13.6 | Zusammenfassung | 529 |
| Anhang | | 531 |
| Quellen | | 531 |
| Stichwortverzeichnis | | 533 |