

Inhalt

Vorwort	XIII
Beteiligte	XV
Einleitung	XIX
1 Grundlagen von jQuery	1
1.1 Einbinden der jQuery-Bibliothek in eine HTML-Seite	9
1.2 Ausführen von jQuery/JavaScript-Code nach dem Laden des DOM, aber noch vor dem vollständigen Laden der Seite	11
1.3 Selektieren von DOM-Elementen mit Selektoren und der jQuery-Funktion	13
1.4 Selektieren von DOM-Elementen in einem bestimmten Kontext	15
1.5 Ein Wrapper-Set mit DOM-Elementen filtern	16
1.6 Abhängige Elemente im aktuell selektierten Wrapper-Set finden	18
1.7 Vor einer destruktiven Änderung zur vorherigen Selektion zurückkehren	19
1.8 Die vorherige Selektion mit der aktuellen Selektion vereinigen	21
1.9 Das DOM basierend auf dem aktuellen Kontext durchlaufen, um ein neues Set mit DOM-Elementen zu erhalten	22
1.10 DOM-Elemente erstellen, bearbeiten und einfügen	23
1.11 Entfernen von DOM-Elementen	25
1.12 DOM-Elemente ersetzen	26
1.13 DOM-Elemente klonen	27
1.14 Attribute von DOM-Elementen lesen, setzen und entfernen	30
1.15 HTML-Inhalte lesen und setzen	31
1.16 Text-Inhalte lesen und setzen	32
1.17 Den \$-Alias verwenden, ohne globale Konflikte zu erzeugen	33

2	Elemente mit jQuery selektieren	35
2.1	Nur Kind-Elemente selektieren	36
2.2	Bestimmte Geschwister-Elemente selektieren	38
2.3	Elemente über die Index-Reihenfolge selektieren	39
2.4	Aktuell animierte Elemente selektieren	41
2.5	Elemente anhand ihres Inhalts selektieren	42
2.6	Elemente über eine negative Selektion selektieren	43
2.7	Elemente anhand ihrer Sichtbarkeit selektieren	44
2.8	Elemente anhand von Attributen selektieren	45
2.9	Form-Elemente anhand des Typs selektieren	46
2.10	Elemente mit bestimmten Eigenschaften selektieren	47
2.11	Den Kontext-Parameter verwenden	49
2.12	Einen eigenen Filter-Selektor erstellen	50
3	Fortgeschrittene Techniken	53
3.1	Ein Set mit selektierten Ergebnissen durchlaufen	53
3.2	Das Selektions-Set auf ein bestimmtes Element reduzieren	56
3.3	Ein selektiertes jQuery-Objekt in ein reines DOM-Objekt konvertieren	59
3.4	Den Index eines Elements in einer Selektion ermitteln	62
3.5	Aus einem bestehenden Array ein Array mit ausgewählten Werten erstellen	64
3.6	Eine Aktion auf einer Untermenge des selektierten Sets ausführen	66
3.7	Konfigurieren von jQuery, so dass es nicht mit anderen Bibliotheken kollidiert	69
3.8	Funktionalität durch Plugins hinzufügen	71
3.9	Die verwendete Selektion bestimmen	74
4	jQuery-Tools	77
4.1	Features mit jQuery.support erkennen	77
4.2	Mit jQuery.each über Arrays und Objekte iterieren	79
4.3	Arrays mit jQuery.grep filtern	80
4.4	Über Array-Elemente mit jQuery.map iterieren und sie verändern	81
4.5	Zwei Arrays durch jQuery.merge kombinieren	81
4.6	Doppelte Array-Einträge mit jQuery.unique ausfiltern	82
4.7	Callback-Funktionen mit jQuery.isFunction testen	83
4.8	Whitespace aus Strings oder Form-Werten mit jQuery.trim entfernen	84
4.9	Objekte und Daten per jQuery.data an DOM-Elemente anhängen	84
4.10	Objekte durch jQuery.extend erweitern	86

5	Schneller, Einfacher, Spaßiger	89
5.1	Das ist nicht jQuery, sondern JavaScript!	89
5.2	Was ist an \$(this) falsch?	90
5.3	Überflüssige Wiederholungen vermeiden	93
5.4	Ihre verketteten jQuery-Methoden formatieren	95
5.5	Code aus anderen Bibliotheken übernehmen	96
5.6	Einen eigenen Iterator schreiben	99
5.7	Ein Attribut umschalten	101
5.8	Performance-Killer finden	103
5.9	Ihre jQuery-Objekte puffern	108
5.10	Schnellere Selektoren schreiben	109
5.11	Tabellen schneller laden	111
5.12	Schleifen programmieren	114
5.13	Name Lookups verringern	117
5.14	Das DOM mit <code>.innerHTML</code> schneller aktualisieren	120
5.15	Debuggen? Sprengen Sie die Ketten	121
5.16	Ist das ein Bug von jQuery?	123
5.17	In jQuery debuggen	124
5.18	Weniger Server-Anfragen erzeugen	126
5.19	Zurückhaltendes JavaScript schreiben	129
5.20	jQuery für die progressive Verbesserung nutzen	131
5.21	Machen Sie Ihre Seiten barrierefrei	133
6	Dimensionen	137
6.1	Die Dimensionen von Window und Document ermitteln	137
6.2	Ermitteln der Dimensionen eines Elements	138
6.3	Den Offset eines Elements ermitteln	141
6.4	Ein Element in den sichtbaren Bereich scrollen	143
6.5	Ermitteln, ob sich ein Element im sichtbaren Bereich befindet	145
6.6	Zentrieren eines Elements im sichtbaren Bereich	148
6.7	Absolute Positionierung eines Elements an seiner aktuellen Position	149
6.8	Ein Element relativ zu einem anderen Element positionieren	149
6.9	Stylesheets abhängig von der Browser-Breite wechseln	151
7	Effekte	153
7.1	Elemente per Sliding und Fading ein- und ausblenden	155
7.2	Elemente durch ein Sliding Up anzeigen	158
7.3	Erzeugen eines horizontalen Akkordeons	160
7.4	Elemente gleichzeitig Sliden und Faden	163

7.5	Sequenzielle Effekte anwenden	164
7.6	Erkennen, ob Elemente aktuell animiert werden	166
7.7	Animationen stoppen und zurücksetzen	167
7.8	Eigene Easing-Methoden für Effekte nutzen	169
7.9	Alle Effekte deaktivieren	170
7.10	Aufwändigere Effekte mit jQuery UI erzeugen	171
8	Events	173
8.1	Einen Handler mit vielen Events verbinden	174
8.2	Eine Handler-Funktion mit anderen Daten wiederverwenden	175
8.3	Ein ganzes Set mit Eventhandlern entfernen	177
8.4	Eventhandler auslösen	178
8.5	Dynamische Daten an Eventhandler übergeben	179
8.6	Sofortiger Zugriff auf ein Element (noch vor document.ready)	182
8.7	Die Ausführungsschleife für Handler stoppen	185
8.8	Beim Verwenden von event.target das richtige Element erhalten	187
8.9	Mehrere parallele hover()-Animationen vermeiden	189
8.10	Eventhandler für neu hinzugefügte Elemente nutzbar machen	191
9	Events für Fortgeschrittene	195
9.1	jQuery nutzen, wenn es dynamisch geladen wird	195
9.2	Das globale Auslösen von Events beschleunigen	196
9.3	Eigene Events erstellen	199
9.4	Eventhandler stellen benötigte Daten bereit	202
9.5	Event-gesteuerte Plugins erstellen	205
9.6	Benachrichtigungen erhalten, wenn jQuery-Methoden aufgerufen werden	209
9.7	Objekt-Methoden als Event Listener nutzen	213
10	HTML-Forms durch eigenen Code verbessern	217
10.1	Ein Texteingabefeld beim Laden der Seite fokussieren	218
10.2	Form-Elemente aktivieren und deaktivieren	219
10.3	Automatisch Radio Buttons auswählen	222
10.4	(De)selektieren aller Checkboxes durch Links	224
10.5	(De)selektieren aller Checkboxes über einen einzelnen Umschalter	225
10.6	Auswahl-Optionen hinzufügen und entfernen	227
10.7	Abhängig von der Anzahl der Zeichen ins nächste Feld springen	228
10.8	Anzahl der verbleibenden Zeichen anzeigen	230
10.9	Texteingabefelder auf bestimmte Zeichen beschränken	233

10.10	Eine Form mit Ajax abschicken	234
10.11	Forms überprüfen	236
11	Verbesserungen von HTML-Forms durch Plugins	243
11.1	Forms überprüfen	244
11.2	Eingabemasken für Felder erstellen	254
11.3	Textfelder automatisch vervollständigen	256
11.4	Einen Wertebereich selektieren	257
11.5	Einen Wert eingeben, der innerhalb bestimmter Grenzen liegt.	260
11.6	Dateien im Hintergrund hochladen	262
11.7	Die Länge von Texteingabefeldern begrenzen.	264
11.8	Label oberhalb von Eingabefeldern anzeigen	265
11.9	Ein Eingabeelement mit seinem Inhalt wachsen lassen	266
11.10	Ein Datum wählen	267
12	jQuery-Plugins	271
12.1	Wo finden Sie jQuery-Plugins?	271
12.2	Wann sollten Sie ein jQuery-Plugin schreiben?	273
12.3	Schreiben Sie Ihr erstes jQuery-Plugin	275
12.4	Ihrem Plugin Optionen mitgeben	277
12.5	Die Kurzform \$ in Ihrem Plugin verwenden	278
12.6	Private Funktionen in Ihr Plugin aufnehmen	280
12.7	Das Metadata-Plugin unterstützen.	282
12.8	Ihrem Plugin eine statische Funktion hinzufügen	284
12.9	Unit Tests für Ihr Plugin mit QUnit	286
13	Selbst geschriebene Benutzerschnittstellen	289
13.1	Eigene Tooltips erstellen.	291
13.2	In einem Baum navigieren.	295
13.3	Ein Akkordeon aufziehen	298
13.4	Registerkarten in einem Dokument	303
13.5	Ein einfaches modales Fenster anzeigen	306
13.6	Dropdown-Menüs erstellen.	313
13.7	Bilder zyklisch einblenden.	315
13.8	Sliding Panels.	320
14	Benutzerschnittstellen mit jQuery UI	325
14.1	Die komplette jQuery UI-Suite einbinden.	327
14.2	Ein oder zwei einzelne jQuery UI-Plugins einbinden	328

14.3	Ein jQuery UI-Plugin mit den Standard-Optionen initialisieren	329
14.4	Ein jQuery UI-Plugin mit eigenen Optionen initialisieren	330
14.5	Eigene jQuery UI-Plugin-Standardwerte erstellen	331
14.6	Optionen für jQuery UI-Plugins lesen und setzen	333
14.7	Plugin-Methoden von jQuery UI aufrufen	334
14.8	Mit Events von jQuery UI-Plugins umgehen	335
14.9	Ein jQuery UI-Plugin zerstören	337
14.10	Einen Musikplayer mit jQuery UI erstellen	337
15	Themes in jQuery UI	351
15.1	Themes für jQuery UI-Widgets mit ThemeRoller erstellen	356
15.2	Layout und Theme-Styles von jQuery UI überschreiben	370
15.3	Ein Theme auf Komponenten anwenden, die nicht zum jQuery UI gehören	381
15.4	Mehrere Themes auf einer einzelnen Seite verwenden	391
15.5	Anhang: Weitere Informationen zu CSS	400
16	jQuery, Ajax, Datenformate: HTML, XML, JSON, JSONP	401
16.1	jQuery und Ajax	401
16.2	Ajax auf der gesamten Site verwenden	404
16.3	Einfache Ajax-Anwendung mit Rückmeldungen an den Benutzer	406
16.4	Ajax-Hilfsfunktionen und Datentypen	410
16.5	HTML-Fragmente mit jQuery einsetzen	412
16.6	XML-Code in ein DOM konvertieren	413
16.7	JSON erzeugen	414
16.8	JSON parsen	415
16.9	jQuery und JSONP verwenden	416
17	jQuery in großen Projekten verwenden	419
17.1	Auf dem Client speichern	419
17.2	Den Anwendungs-Status für eine einzelne Session speichern	422
17.3	Den Anwendungs-Status über eine Session hinaus speichern	424
17.4	Eine JavaScript Template Engine nutzen	425
17.5	Ajax-Anfragen queuen	428
17.6	Ajax und der Zurück-Button	430
17.7	JavaScript am Seitenende unterbringen	432
18	Unit Tests	435
18.1	Unit Tests automatisieren	435
18.2	Ergebnisse sicherstellen	437

18.3	Synchrone Callbacks testen	438
18.4	Asynchrone Callbacks testen	439
18.5	Benutzeraktionen testen	440
18.6	Tests atomar halten	441
18.7	Tests gruppieren	443
18.8	Durchzuführende Tests auswählen	444
Index	446