

JQuery



- Freier, umfangreicher JavaScript-Framework
- 2006 von John Resig in New York veröffentlicht
- Weltweit am meisten verwendeter JavaScript-Framework, ca. 40% aller JavaScript-unterstützten Webseiten nutzen jQuery
- Funktionen
 - Elementselektion im DOM
 - DOM-Manipulation
 - Erweitertes Event-System
 - Zahlreiche Hilfsfunktionen
 - Effekte und Animationen
 - Ajax-Funktionalität
 - erweiterbar

Einbindung

jQuery besteht aus einer JavaScript-Datei, die in die HTML-Seite eingebunden werden muss

```
<script src="jquery-1.7.2.js"></script>
```

Im folgenden steht dann die zentrale Funktion `jQuery()` zur Verfügung. Diese kann auch mit `$()` verkürzt geschrieben werden.
Beispiel:

```
jQuery(...);   oder   $(...);
```

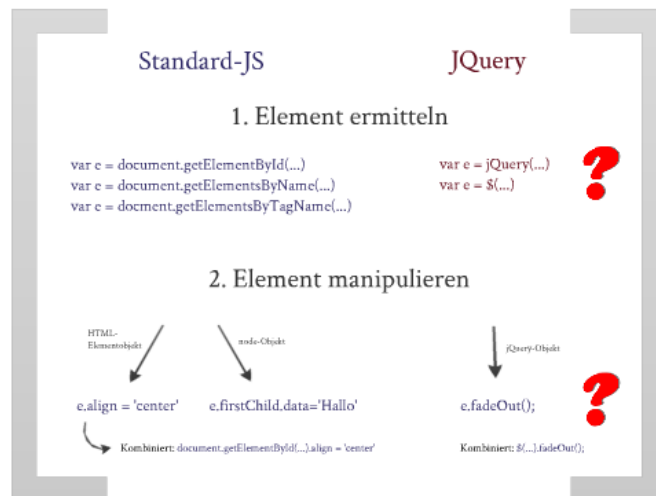


<http://jquery.com/download/>

Idee / Weg

Möchte man eine HTML-Seite manipulieren, so ändert sich an der bisher bekannten Vorgehensweise nichts. Im ersten Schritt ermittelt man das für den zu manipulierenden Bereich zuständige Javascript-Objekt. Im zweiten Schritt nutzt man die Fähigkeiten des in Punkt 1 ermittelten Javascript-Objekts, um durch Methodenaufruf und Manipulation der Eigenschaften den gewünschten Effekt zu erzeugen.

Vergleich



Standard-JS

jQuery

1. Element ermitteln

```
var e = document.getElementById(...)  
var e = document.getElementsByName(...)  
var e = document.getElementsByTagName(...)
```

```
var e = jQuery(...)  
var e = $(...)
```



2. Element manipulieren

HTML-
Elementobjekt



```
e.align = 'center'
```

node-Objekt



```
e.firstChild.data='Hallo'
```

jQuery-Objekt



```
e.fadeOut();
```



Kombiniert: `document.getElementById(...).align = 'center'`

Kombiniert: `$(...).fadeOut();`



Selector Engine

Schritt 1

Die im zweiten Schritt zu manipulierenden JQuery-Objekte werden über die JQuery-Funktion ermittelt. Die Funktion erwartet als Parameter einen CSS-Selektor. Der Selektor dient als Suchkriterium.

`$("*")` - Alle Elemente

`$("p")` - HTML-Tag als Suchkriterium

`$("#content")` - ID als Suchkriterium

`$(".wichtig")` - Klasse als Suchkriterium

`$("p.wichtig")` - Kombination

`$("p:first")` - Anwendung von Filtern

`$(document)` - aktuelle Dokument

Ein Treffer



`$("p").fadeOut();`

Hinweis
Sollte die "Auswahl" doch mehr als einen Treffer liefern, wird die Funktion nur auf den ersten Treffer angewendet!



Viele Treffer

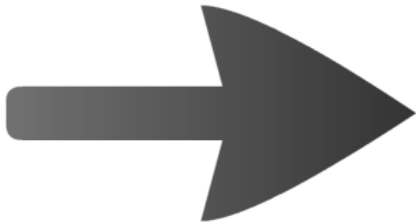
```
$("p").each(function(){  
    $(this).fadeOut();  
});
```

Erklärung:
Art "Schleife", um alle Treffer abzuarbeiten.
Innerhalb der Funktion kann mit dem Ausdruck `$(this)` auf das aktuelle Element zugegriffen werden.



<http://api.jquery.com/category/selectors>

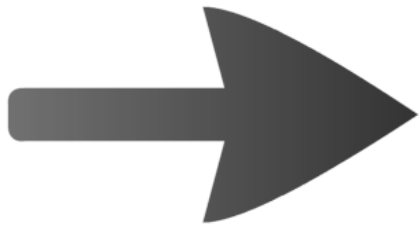
Ein Treffer



```
$("#p").fadeOut();
```

Hinweis

Sollte die "Auswahl" doch mehr als einen Treffer liefern, wird die Funktion nur auf den ersten Treffer angewendet!



Viele Treffer

```
$("#p").each(function()  
    $(this).fadeOut();  
});
```

Erklärung:

Art "Schleife", um alle Treffer abzuarbeiten.
Innerhalb der Funktion kann mit dem Ausdruck `$(this)` auf das aktuelle Element zugegriffen werden.

Anonyme Funktionen

Die Methode "each" erwartet als Parameter eine Funktion, die dann für alle Elemente der Auswahl ausgeführt wird.

```
$("#p").each(Callback-Funktion);
```



```
function show(){  
  alert($(this).fadeOut);  
}  
  
$("#p").each(show);
```



üblich

```
$("#p").each(  
  function(){  
    $(this).fadeOut();  
  }  
);
```



Anonyme
Funktion

Methoden des JQuery-Objekts

Schritt 2

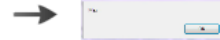
Zugriff auf Formularelementen

Name: `<input type="text" id="eingabe">`

Name:

Wert eines Eingabeelements lesen:

```
var n = $("#eingabe").val();  
alert(n);
```



Wert eines Eingabeelements schreiben (setzen):



```
$("#eingabe").val("Peter");
```

Name:

<http://api.jquery.com/jquery.val/>



Ereignisse / Events

In JQuery kann jedes Element mit einem Ereignis gekoppelt werden. Bei Auslösung des Ereignisses wird die zugehörige Callback-Funktion aufgerufen.

Beispiel

Name: `<input type="text" id="eingabe">`

Name:

Beispielskripten bei Mausklick in das Eingabefeld

```
$(document).ready(function(){  
  $("#eingabe").click(function(){  
    alert("Max eingabe");  
  });  
});
```

Beispielskripten bei Verlust des Eingabefeld

```
$(document).ready(function(){  
  $("#eingabe").blur(function(){  
    alert("Max eingabe");  
  });  
});
```

ready
focus
keyup
submit
change
mousemove

<http://api.jquery.com/category/events/>

DOM-Elemente einfügen, ersetzen und entfernen

Über das JavaScript-Objekt "node" können wir den der Seite zugrundeliegenden Elementbaum manipulieren. Wir können Knoten/Unterbäume hinzufügen, ändern oder löschen und damit die Struktur der Seite beeinflussen. Auch JQuery bietet entsprechende Methoden an, die deutlich einfacher zu verwenden sind.

1

append

Inhalte hinter das letzte Kind des
selektierten Elements einhängen

```
$("p:first").append("<h1>Hallo Welt</h1>");
```

2

replaceWith

Selektierten Elements wird
durch neue Inhalte ersetzt

```
$("p:first").replaceWith("<h1>Hallo Welt</h1>");
```

3

Inhalte löschen: `$("#p:first").empty();`

<http://api.jquery.com/category/manipulation/>



<http://api.jquery.com/>

Zugriff auf Fomularelementen

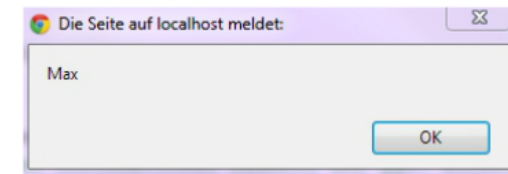
Name: `<input type="text" id="eingabe">`

Name:



Wert eines Eingabeelements lesen:

```
var n = $("#eingabe").val();  
alert( n );
```



Wert eines Eingabeelements schreiben (setzen):

```
$("#eingabe").val("Peter");
```



Name:



<http://api.jquery.com/category/forms/>

Ereignisse / Events



In JQuery kann jedes Element mit einem Ereignis gekoppelt werden. Bei Auslösung des Ereignisses wird die zugehörige Callback-Funktion aufgerufen.

Beispiel

Name:

Name:

Benachrichtigungsfenster bei
Mausklick in das Eingabefeld

```
$("#eingabe").click(  
  function(){  
    alert( $("#eingabe").val() );  
  }  
);
```

Benachrichtigungsfenster bei
Verlassen des Eingabefelds

```
$("#eingabe").blur(  
  function(){  
    alert( $("#eingabe").val() );  
  }  
);
```

ready

focus

keyup

mousemove

submit

change



<http://api.jquery.com/category/events/>

Name: <input type="text" id="eingabe">

Name:

Benachrichtigungsfenster bei
Mausklick in das Eingabefeld

```
$("#eingabe").click(  
  function(){  
    alert( $("#eingabe").val() );  
  }  
);
```

Benachrichtigungsfenster bei
Verlassen des Eingabefelds

```
$("#eingabe").blur(  
  function(){  
    alert( $("#eingabe").val() );  
  }  
);
```

ready-Event

Mit Hilfe der Methode "ready" lassen sich auf einer Seite Initialisierungsaufgaben erledigen, also Aufgaben, die direkt nach dem vollständigen Laden der Seite ausgeführt werden müssen. Zu solchen Aufgaben gehört beispielsweise das Verknüpfen von Elementen mit Ereignissen. Vor dem vollständigen Laden der Seite können mit der Selector-Engine keine Elemente gefunden werden.

```
<html>

<head>
  <script src="jquery-1.7.2.js"></script>
  <script type="text/javascript">
    $(document).ready(
      function () {
        //Initialisierungsaufgaben
        $("#eingabe").click(
          function() {
            [...]
          }
        );
      }
    );
  [...]
</script>
</head>

<body>
  [...]
</body>

</html>
```

Aufgabe 10

Erzeugen Sie eine HTML-Seite mit zwei Texteingabefeldern und einer Schaltfläche. Nach Klick auf die Schaltfläche soll der Text aus dem oberen Eingabefeld in das untere übertragen werden. Das obere Feld wird anschließend geleert. Nutzen Sie für die Ereignisbehandlung und das Lesen/Setzen der Textfelder ausschließlich JQuery-Funktionen.

Feld 1:

Feld 2:

DOM-Elemente einfügen, ersetzen und entfernen

Über das JavaScript-Objekt "node" können wir den der Seite zugrundeliegenden Elementbaum manipulieren. Wir können Knoten/Unterbäume hinzufügen, ändern oder löschen und damit die Struktur der Seite beeinflussen. Auch JQuery bietet entsprechende Methoden an, die deutlich einfacher zu verwenden sind.

1

append

Inhalte hinter das letzte Kind des
selektierten Elements einhängen



```
$("#p:first").append("<h1>Hallo Welt</h1>");
```

2

replaceWith

Selektierten Elements wird
durch neue Inhalte ersetzt



```
$("#p:first").replaceWith("<h1>Hallo Welt</h1>");
```

3

Inhalte löschen: `$("#p:first").empty();`



<http://api.jquery.com/category/manipulation/>

Aufgaben

Aufgabe

Laden Sie folgende Aufgaben ohne Änderungen am HTML-Code der Tabelle und ohne Änderungen im CSS-Block!!!
Nutzen Sie ausschließlich jQuery für die Umsetzung der Aufgaben.

1. Öffnen Sie die HTML-Seite „jQBeispiel.html“ im Browser und in einem Editor.
2. Binden Sie jQuery ein.
3. Ändern Sie die Seite so ab, dass die Tabelle nach dem Laden der Seite nicht mehr sichtbar ist. (Siehe <http://api.jquery.com/category/effects/> und <http://api.jquery.com/category/effects/>)
4. Platzieren Sie am Ende der Seite eine Schaltfläche, mit welcher die Tabelle eingeblendet werden kann. Lassen Sie die Tabelle langsam erscheinen. Verziehen Sie auf die klassischen HTML-Eventhandler! (Siehe <http://api.jquery.com/category/effects/> und <http://api.jquery.com/category/effects/>)
5. Fügen Sie einen zweiten Knopf hinzu, mit welchem die Tabelle wieder langsam ausgeblendet werden kann. Verziehen Sie auf die klassischen HTML-Eventhandler! (Siehe <http://api.jquery.com/category/effects/> und <http://api.jquery.com/category/effects/>)
6. Fügen Sie eine weitere Schaltfläche ein, mit welcher die Seite neu geladen werden kann. Verziehen Sie auf die klassischen HTML-Eventhandler! (Siehe <http://api.jquery.com/category/effects/>)
7. Ergänzen Sie die Seite so, dass die Punktzahl in der Spalte „Punkte“ zentriert dargestellt wird. Änderungen im CSS-Block sind nicht erlaubt. Nutzen Sie ausschließlich jQuery! (Siehe <http://api.jquery.com/category/css/>)
8. Färben Sie Zeilen wie folgt ein. Änderungen im CSS-Block sind nicht erlaubt. Nutzen Sie ausschließlich jQuery! (Siehe <http://api.jquery.com/category/css/>)

Platz	Farbe	CSS-Klasse
1-3	Orange	CL1
4	Hellorange	CL2
5-6	Grün	UI
7-15	Grau	standard
16	Hellrot	relegation
17-18	Rot	absteiger

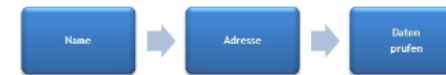
9. Fügen Sie am Ende der Seite eine Checkbox ein. Bei gesetztem Haken soll die Tabelle um eine zusätzliche Spalte „Platz“ ergänzt werden. Die Spalte befindet sich ganz vorne und zeigt die Platzierung (1, 2, 3, ...). Die Überschrift der Spalte lautet „Platz“. Direkte Änderungen in der Tabelle sind nicht erlaubt. Nutzen Sie ausschließlich jQuery! Verziehen Sie auf die klassischen HTML-Eventhandler! (Siehe <http://api.jquery.com/category/effects/> und <http://api.jquery.com/category/manipulation/>)
10. Wie 9., verwenden Sie jedoch einen Knopf mit der Aufschrift „Spalte 'Platz' hinzufügen / entfernen“. Bei Klick auf diesen Knopf wird die Spalte ein- bzw. ausgeblendet. Verziehen Sie auf die klassischen HTML-Eventhandler! (Tipp: <http://api.jquery.com/show/>)

Meine jQuery-Testseite

Mannschaft	Punkte
Borussia Dortmund	81
Bayern München	73
FC Schalke 04	64
Borussia M'gladbach	60
Bayer Leverkusen	54
VB Stuttgart	53
Hannover 96	48
VfL Wolfsburg	44
Werder Bremen	42
1. FC Nürnberg	42
1899 Hoffenheim	41
SC Freiburg	40
1. FC Köln	39
Hamburger SV	38
1. FC Köln	31
1. FC Köln	30
1. FC Köln	23

☐

Onlineanmeldung



Name eingeben

Name

Vorname

Aufgabe

Lösen Sie folgende Aufgaben ohne Änderungen am HTML-Code der Tabelle und ohne Änderungen im CSS-Block!!! Nutzen Sie ausschließlich jQuery für die Umsetzung der Aufgaben.

1. Öffnen Sie die HTML-Seite „jqBeispiel.html“ im Browser und in einem Editor.
2. Binden Sie jQuery ein.
3. Ändern Sie die Seite so ab, dass die Tabelle nach dem Laden der Seite nicht mehr sichtbar ist. (Siehe <http://api.jquery.com/category/events/> und <http://api.jquery.com/category/Effects/>)
4. Platzieren Sie am Ende der Seite eine Schaltfläche, mit welcher die Tabelle eingeblendet werden kann. Lassen Sie die Tabelle langsam erscheinen. Verzichten Sie auf die klassischen HTML-Eventhandler! (Siehe <http://api.jquery.com/category/events/> und <http://api.jquery.com/category/Effects/>)
5. Fügen Sie einen zweiten Knopf hinzu, mit welchem die Tabelle wieder langsam ausgeblendet werden kann. Verzichten Sie auf die klassischen HTML-Eventhandler! (Siehe <http://api.jquery.com/category/events/> und <http://api.jquery.com/category/Effects/>)
6. Fügen Sie eine weitere Schaltfläche ein, mit welcher die Seite neu geladen werden kann. Verzichten Sie auf die klassischen HTML-Eventhandler! (Siehe <http://api.jquery.com/category/events/>)
7. Ergänzen Sie die Seite so, dass die Punktzahl in der Spalte „Punkte“ zentriert dargestellt wird. Änderungen im CSS-Block sind nicht erlaubt. Nutzen Sie ausschließlich jQuery! (Siehe <http://api.jquery.com/category/css/>)
8. Färben Sie Zeilen wie folgt ein. Änderungen im CSS-Block sind nicht erlaubt. Nutzen Sie ausschließlich jQuery! (Siehe <http://api.jquery.com/category/css/>)

Platz	Farbe	CSS-Klasse
1-3	Orange	CL1
4	Hellorange	CL2
5-6	Grün	UI
7-15	Grau	standard
16	Hellrot	relegation
17-18	Rot	absteiger

9. Fügen Sie am Ende der Seite eine Checkbox ein. Bei gesetztem Haken soll die Tabelle um eine zusätzliche Spalte „Platz“ ergänzt werden. Die Spalte befindet sich ganz vorne und zeigt die Platzierung (1, 2, 3, ...). Die Überschrift der Spalte lautet „Platz“. Direkte Änderungen an der Tabelle sind nicht erlaubt. Nutzen Sie ausschließlich jQuery! Verzichten Sie auf die klassischen HTML-Eventhandler! (Siehe <http://api.jquery.com/category/events/> und <http://api.jquery.com/category/manipulation/>)
10. Wie 9., verwenden Sie jedoch einen Knopf mit der Aufschrift „Spalte 'Platz' hinzufügen / entfernen“. Bei Klick auf diesen Knopf wird die Spalte ein- bzw. ausgeblendet. Verzichten Sie auf die klassischen HTML-Eventhandler! (Tipp: <http://api.jquery.com/data/>)

Meine jQuery-Testseite

Mannschaft	Punkte
Borussia Dortmund	81
Bayern München	73
FC Schalke 04	64
Borussia M'gladbach	60
Bayer Leverkusen	54
VB Stuttgart	53
Hannover 96	48
VfL Wolfsburg	44
Werder Bremen	42
1. FC Nürnberg	42
1899 Hoffenheim	41
SC Freiburg	40
1. FSV Mainz 05	39
FC Augsburg	38
Hamburger SV	36
Hertha BSC	31
1. FC Köln	30
1. FC Kaiserslautern	23

Tabelle anzeigen

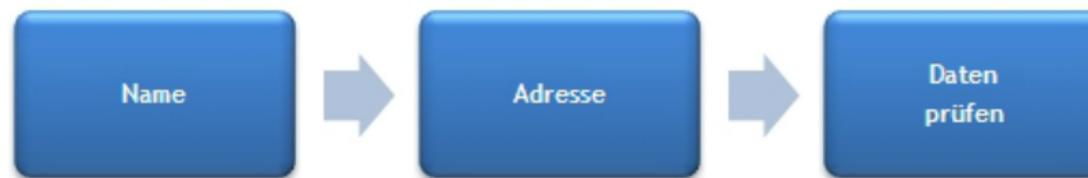
Tabelle ausblenden

Spalte 'Platz' hinzufügen / entfernen

☐ Spalte 'Platz' hinzufügen / entfernen

Seite neu laden

Onlineanmeldung



Name eingeben

Name	<input type="text"/>
Vorname	<input type="text"/>