

jQuery – Einstieg 2 – CSS manipulieren; toggle, hide und show

Inhalt:

1. CSS-Eigenschaften manipulieren
2. Beispiel 2: Elemente ein- oder ausblenden - Animationen
3. Theorie - Methode .toggle() bzw. hide() bzw. show()
4. Praxis Übung: Methode .toggle() bzw. hide() bzw. show()

1)CSS-Eigenschaften manipulieren

Die Methode “.css“ setzt beim <style>-Attribut eines DOM-Knotens an und bietet gleich zweierlei: zum einen kann sie beliebige, für ein Element gültige CSS-Eigenschaften auslesen und diese zum anderen bei Bedarf auch schreiben.

a)Zum Lesen einer Eigenschaft

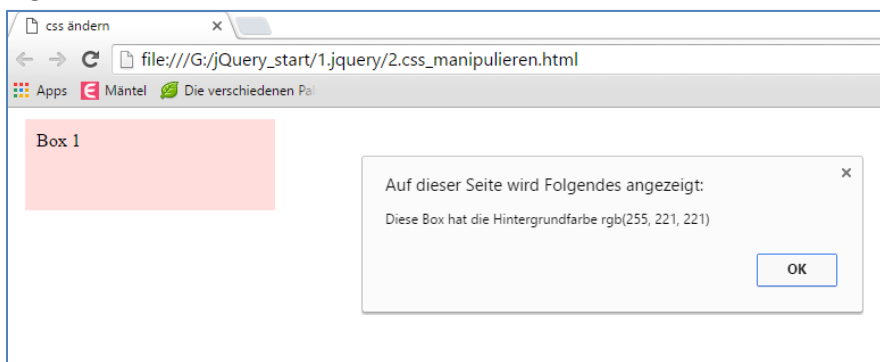
muss man lediglich den Namen des in Frage kommenden Properties (als String mit Anführungszeichen) übergeben:

```
$(“div”).click(function () {  
    //Hintergrundfarbe auslesen:  
    var color = $(this).css(“background-color”);  
    alert(“Dieser Div hat die Hintergrundfarbe “ +color);  
});
```

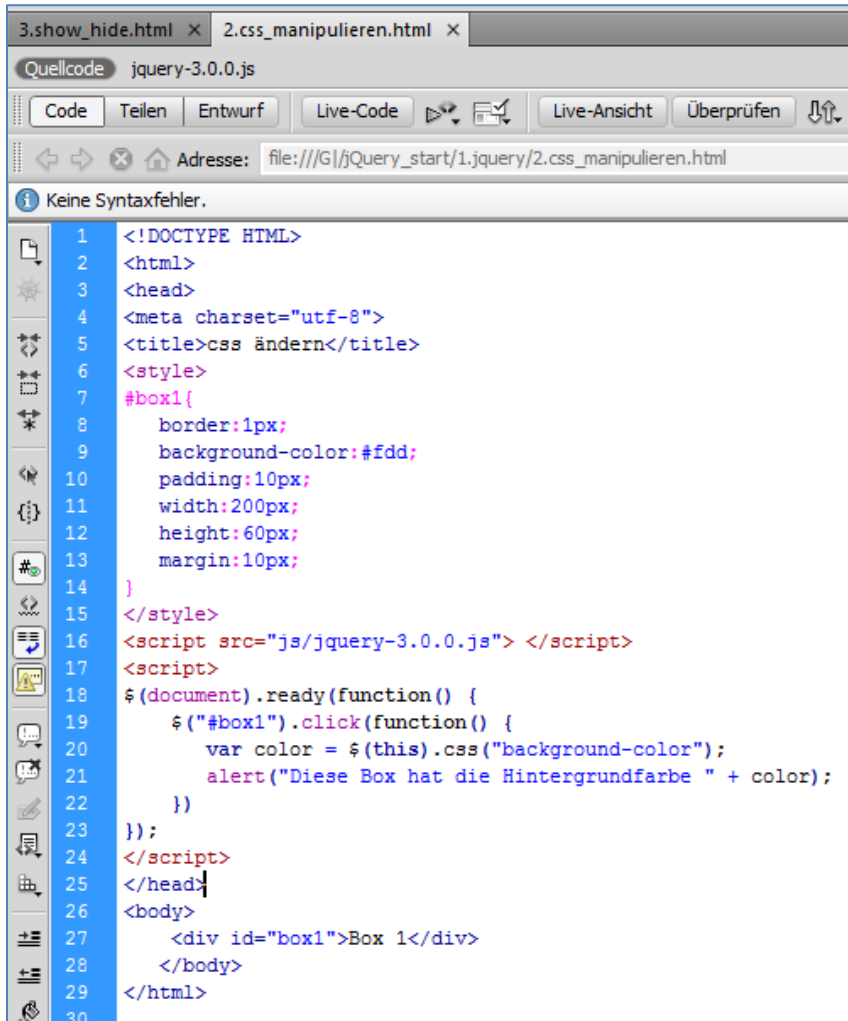
Übung: 2.css manipulieren.html

Erstelle eine neue HTML-Datei „2.css_manipulieren.html“ mit einer Box1, die mit <style> einen CSS-Stil erhält, der in folgender Abbildung angeführt ist. Darunter wird jQuery eingebunden und wie oben dargestellt, der Farbcode des Hintergrundes ausgelesen und als „alert“-Fenster ausgegeben:

Ergebnis:



Code:



```
1 <!DOCTYPE HTML>
2 <html>
3 <head>
4 <meta charset="utf-8">
5 <title>css ändern</title>
6 <style>
7 #box1{
8     border:1px;
9     background-color:#fdd;
10    padding:10px;
11    width:200px;
12    height:60px;
13    margin:10px;
14 }
15 </style>
16 <script src="js/jquery-3.0.0.js"> </script>
17 <script>
18 $(document).ready(function() {
19     $("#box1").click(function() {
20         var color = $(this).css("background-color");
21         alert("Diese Box hat die Hintergrundfarbe " + color);
22     })
23 });
24 </script>
25 </head>
26 <body>
27     <div id="box1">Box 1</div>
28 </body>
29 </html>
30
```

b)Erweiterung – Wert ändern

Um eine Eigenschaft zu setzen wird als zweites Argument am einfachsten direkt der gewünschte Wert übergeben:

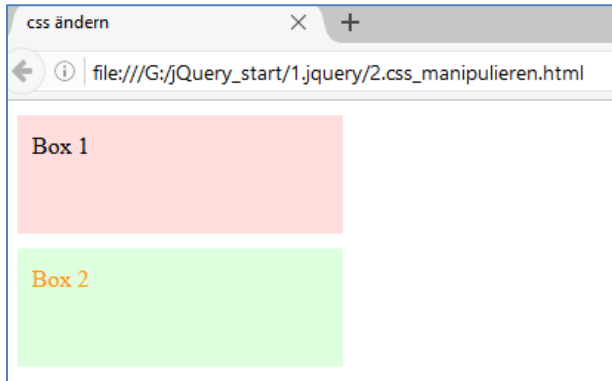
```
$(document).ready(function() {
    $("#box2").click(function() {
        //Schriftfarbe auf Orange setzen:
        $(this).css("color", "#ff9900");
    })
});
```

Übung: 2.css manipulieren.html

Kopiere den Stil der ersten Box und ändere nur die Hintergrundfarbe. Erstelle auch eine <div> dafür im <body>.

Dann soll im <script> die obige Änderung der <color> erfolgen. Speichern.

Ergebnis



Code:

```
3.show_hide.html x 2.css_manipulieren.html x
Quellcode jquery-3.0.0.js
Code Teilen Entwurf Live-Code Live-Ansicht Überprüfen
Adresse: file:///G:/jQuery_start/1.jquery/2.css_manipulieren.html
Keine Syntaxfehler.
1 <!DOCTYPE HTML>
2 <html>
3 <head>
4 <meta charset="utf-8">
5 <title>css ändern</title>
6 <style>
7 #box1{
8     border:1px;
9     background-color:#fdd;
10    padding:10px;
11    width:200px;
12    height:60px;
13    margin:10px;
14 }
15 #box2{
16    border:1px;
17    background-color:#dfd;
18    padding:10px;
19    width:200px;
20    height:60px;
21    margin:10px;
22 }
23 </style>
24 <script src="js/jquery-3.0.0.js"> </script>
25 <script>
26 $(document).ready(function() {
27     $("#box1").click(function() {
28         var color = $(this).css("background-color");
29         alert("Diese Box hat die Hintergrundfarbe " + color);
30     })
31 });
32 $(document).ready(function() {
33     $("#box2").click(function() {
34         //Schriftfarbe auf Orange setzen:
35         $(this).css("color", "#ff9900");
36     })
37 });
38 </script>
39 </head>
40 <body>
41     <div id="box1">Box 1</div>
42     <div id="box2">Box 2</div>
43 </body>
44 </html>
45
```

2) Beispiel 2: Elemente ein- oder ausblenden - Animationen

Möchte man ein lebendig wirkendes, interaktives User Interface gestalten, so empfiehlt sich der dezente Einsatz von Animationen.

Einfache Animationsmethoden müssen an jQuery-Collections nicht über Plugins oder das jQuery-UI-Tool zur Verfügung gestellt werden, sondern gehören als „Grundbedürfnis“ zum jQuery-Core.

Beispiele: `.hide()`, `.hide(d)`, `.show()`, `.show(d)`, `.toggle()`

Wie die Namen bereits aussagen, dient `.show()` zum Einblenden, `.hide()` zum Ausblenden eines Element-Containers. Eine dritte Methode `.toggle()` wirkt als Kombination beider Grundmethoden, versteckt sichtbare Elemente, blendet dafür versteckte im Gegenzug ein. Die Methoden werden einfach auf eine Collection angewendet und bearbeiten alle darin enthaltenen Items gleichzeitig.

Übung:

Ziel: mittels *Mausklick* soll die Sichtbarkeit eines div-Bereichs verändert werden.

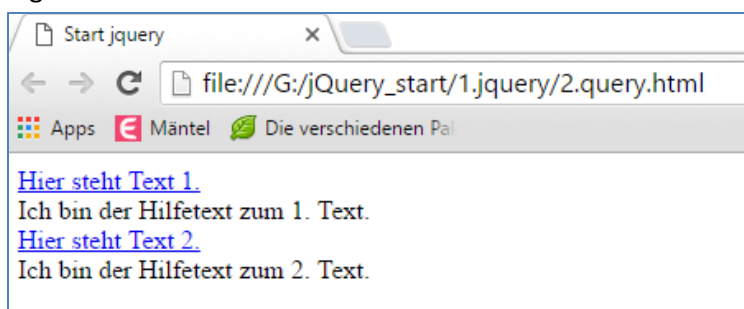
Das ist äußerst praktisch. Denn so kann man "störenden" Text oder Bilder ganz einfach vom Besucher ausblenden lassen. Interessant kann das aber beispielsweise auch sein, wenn man zusätzliche Informationen in Form von Hilfetexten einblenden möchte.

Erstelle eine neue Datei **„2.jquery.html“**. Binde wieder die „jquery“-Datei aus dem Ordner „js“ im `<head>` ein.

Die HTML-Datei soll folgenden Inhalt im `<body>` haben:

```
<body>
  <div>
    <a href="#" class="links">Hier steht Text 1.</a>
    <div>
      Ich bin der Hilfetext zum 1. Text.
    </div>
  </div>
  <div>
    <a href="#" class="links">Hier steht Text 2.</a>
    <div>
      Ich bin der Hilfetext zum 2. Text.
    </div>
  </div>
</body>
```

Ergebnis:

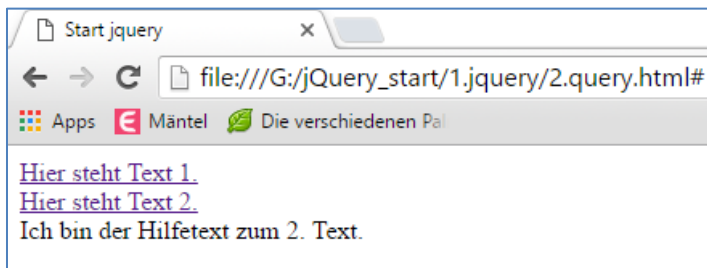


Das eigentliche Skript hält sich – wie das bei jQuery nun mal üblich ist – von seinem Umfang her in Grenzen.

```
<script>
$(document).ready(function() {
  $("a.links").click(function() {
    $(this).next("div").toggle();
  });
});
</script>
```

Der Event-Handler click sucht in diesem Fall nach dem div-Element, das dem gerade angeklickten *Hyperlink* folgt, und verändert dessen Sichtbarkeit. Ist der div-Bereich also ausgeblendet, wird er durch das Anklicken des Links wieder eingeblendet. Andersherum funktioniert das dann genauso.

Ergebnis bei einem Klick auf den ersten Link:



Code:

```
2.query.html* x
Quellcode jquery-3.0.0.js
Code Teilen Entwurf Live-Code Live-Ansicht Überprüfen
Adresse: file:///G:/jquery_start/1.jquery/2.query.html
1 <!DOCTYPE HTML>
2 <html>
3 <head>
4 <meta charset="utf-8">
5 <title>Start jquery</title>
6 <script src="js/jquery-3.0.0.js">
7 </script>
8 <script>
9 $(document).ready(function() {
10   $("a.links").click(function() {
11     $(this).next("div").toggle();
12   });
13 });
14 </script>
15 </head>
16 <body>
17   <div>
18     <a href="#" class="links">Hier steht Text 1.</a>
19     <div>
20       Ich bin der Hilfetext zum 1. Text.
21     </div>
22   </div>
23   <div>
24     <a href="#" class="links">Hier steht Text 2.</a>
25     <div>
26       Ich bin der Hilfetext zum 2. Text.
27     </div>
28   </div>
29 </body>
30 </html>
31
```

3)Theorie - Methode .toggle() bzw. hide() bzw. show()

Methode .show() und .hide():

Diese einfache Funktion ermöglicht es, den Container „box1“ aus- bzw. einzublenden:

```
function hide1() {  
    $(".toggle");  
}
```

Die Methode „.toggle()“

wertet die aktuelle Sichtbarkeit des bearbeiteten Elements aus und zeigt oder versteckt es entsprechend:

```
function toggle1() {  
    $(this).toggle();  
}
```

Werden keine Argumente übergeben, so wendet „.toggle()“ die Methoden „.show()“ und „.hide()“ auf die Elemente der Collection an.

Die Dauer der Animation kann man festlegen:

```
function toggle2() {  
    $(this).toggle("fast");  
}
```

Dauer für toggle(), hide() und (show())

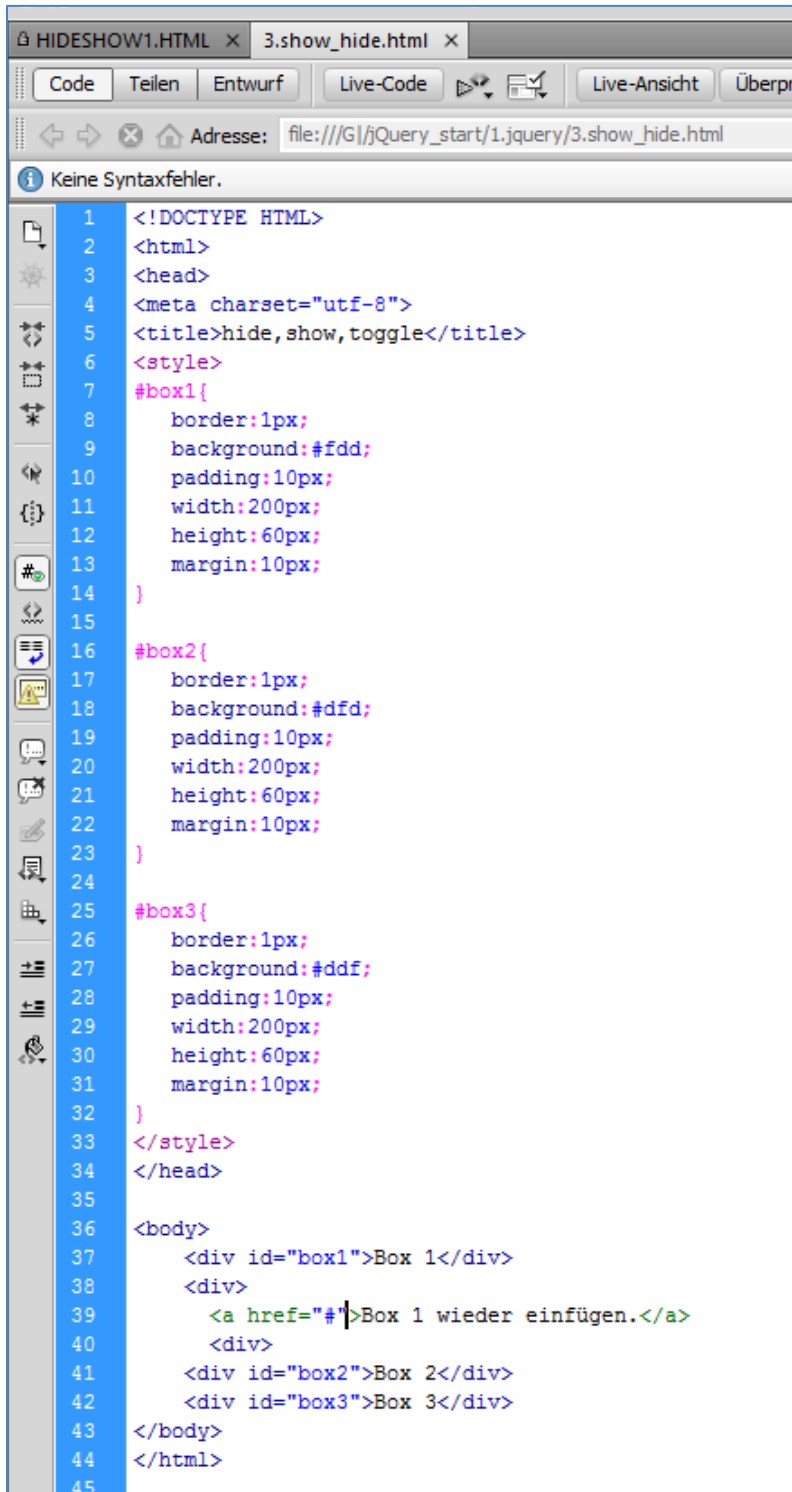
Zeitdauerangabe	Dauer in Millisekunden
(ohne)	0 (default)
fast	200
slow	600
Positive Ganzzahl oder gleich 0	Übergebener Wert ins ms

Beispiel:

```
function toggle3() {  
    //keine Anführungszeichen um den ms-Wert  
    $(this).toggle(3000);  
}
```

4)Praxis Übung: Methode .toggle() bzw. hide() bzw. show()

Erstelle eine „3.show-hide.html“ mit folgenden Style-Vorschriften im <head> und den drei <div> im <body> plus einem Link <a href> zum Wieder-Einblenden. Als Basis kann die Übung „2.css_manipulieren.html“ herangezogen werden, da dort die ersten beiden „Boxen“ schon vorhanden sind:



```
1 <!DOCTYPE HTML>
2 <html>
3 <head>
4 <meta charset="utf-8">
5 <title>hide, show, toggle</title>
6 <style>
7 #box1{
8     border:1px;
9     background:#fdd;
10    padding:10px;
11    width:200px;
12    height:60px;
13    margin:10px;
14 }
15
16 #box2{
17     border:1px;
18     background:#dfd;
19     padding:10px;
20     width:200px;
21     height:60px;
22     margin:10px;
23 }
24
25 #box3{
26     border:1px;
27     background:#ddf;
28     padding:10px;
29     width:200px;
30     height:60px;
31     margin:10px;
32 }
33 </style>
34 </head>
35
36 <body>
37     <div id="box1">Box 1</div>
38     <div>
39         <a href="#">Box 1 wieder einfügen.</a>
40     </div>
41     <div id="box2">Box 2</div>
42     <div id="box3">Box 3</div>
43 </body>
44 </html>
45
```

Füge nach dem Ende vom `</style>` den Code für jQuery ein:

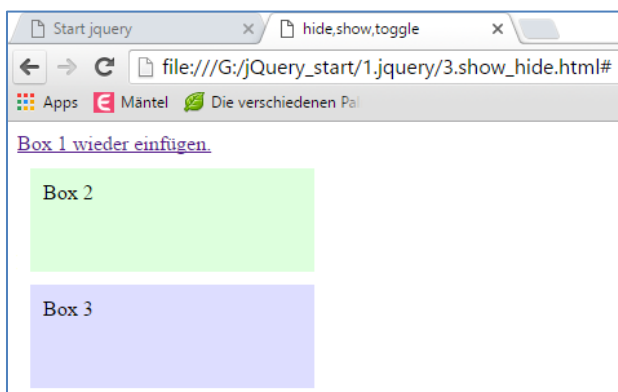
1. Die Verlinkung zur jQuery-Datei im Ordner „js“
2. Die Methode „.hide()“
3. Die Methode „.show()“, die sich auf die Einblendung spezialisiert, wenn man den Hyperlink anklickt.

```
33 </style>
34 <script src="js/jquery-3.0.0.js"> </script>
35 <script>
36 $(document).ready(function() {
37     $("#box1").click(function() {
38         $(this).hide();
39     })
40 });
41 $(document).ready(function() {
42     $("a").click(function() {
43         $("#box1").show();
44     });
45 });
46
47 </script>
48 </head>
```

Ergebnis:



Nach dem Ausblenden:

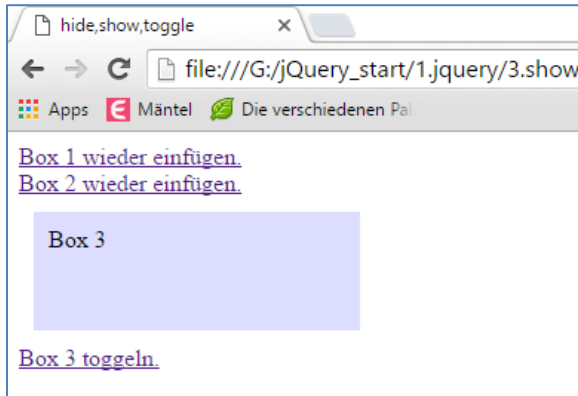


Aufgabe:

a)Erstelle für die Box2 ein Ausblenden mit 5000 Millisekunden und einem neuen Link unter der Box2, damit diese wieder zurückgeholt werden kann.

b)Erstelle einen Link über der Box3. Dieser soll die Methode „.toggle()“ verwenden, um die Box3 ein- bzw. ausblenden zu können.

Mögliche Lösung:



Ein Problem wurde jedoch nicht gelöst: es werden alle Boxen eingeblendet, wenn man Box3 toggled.

Übung:

Diese Funktion schreibt beim Ausblenden eine Nachricht in eine benachbarte Box und beim Einblenden in die animierte Box, wobei in der zweiten Box der ursprüngliche Inhalt wiedergegeben wird.

```
function show1() {  
    $("#box1").show("slow", function() {  
        $(this).html(„Bin wieder da.“);  
        $("box2").html("Box 2");  
    }  
}
```

Übung:

Bild nach einem Klick „einfaden“ lassen

Erstelle ein Dokument „1.3.jquery_fadein.html“ mit dem Bild „img“ und einer Verlinkung zu jQuery.

Statt eines Links, der in „a“ eingebettet ist, nutzt dieses Mal eine ID, z.B. „klickMich“ auf die dann mit dem jQuery Selektor zugegriffen wird.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Kampagne: Pelzige Freunde</title>
    <link rel="stylesheet" type="text/css" href="styles/mein_stil.
css">
  </head>
  <body>
    <div id="clickMe">Pelzigen Freund des Tages anzeigen</div>
    <div id="bilderrahmen">
      
    </div>
    <script src="scripts/jquery-1.6.2.min.js"></script>
    <script>
      $(document).ready(function() {
        $("#clickMe").click(function() {
          $("img").fadeIn(1000);
          $("#bilderrahmen").slideToggle("slow");
        });
      });
    </script>
  </body>
</html>
```

In jQuery ist es wichtig, die Effekte so nacheinander ablaufen zu lassen, dass sie sich dabei nicht in die Quere kommen. Wir werden uns im Laufe des Buches noch weiter mit diesem Thema befassen.

Zuerst wird der Einblendeffekt für das Bild ausgeführt

Wir haben noch etwas in die Klammern geschrieben, um die Effekte ein bisschen aufzumotzen. Mehr dazu erfahren Sie in Kapitel 5.



index.html

Quellen 1.1.jquery-Einstieg.docx und 1.2.jquery.docx

Maximilian Vollendorf, Frank Bongers in: jQuery – Das umfassende Handbuch, Galileo Press, Bonn 2014, S. 81—112 (Basics), S.238-240 (CSS manipulieren) , S.249-274 (Animationen)

Sascha Schoppengerd, Amalia Schoppengerd in: jQuery Praxiseinstieg, mitp-Verlag, 2010

Florence Maurice in: PHP 5.6 und MySQL 5.7, dpunkt, 2015, jQuery, Ajax und PHP, S. 513-532

Ralph Steyer in: jQuery, Das neue JavaScript-Framework für interaktives Design; Addison-Wesley Verlag, 2010, S.59-72, S. 199

Jon Duckett, in: JavaScript & jQuery; Wiley-Vch Verlag, 2015, S.296-305