

Multimedia im Netz

Wintersemester 2013/14

Übung 05 (Hauptfach)

jQuery

- <http://jquery.com>
- JavaScript-Bibliothek, aktuell Version 1.10.2
- Features:
 - Einfacher DOM-Zugriff
 - Event-Handling
 - Animationen
 - Einfache AJAX-Requests
- „jQuery is designed to change the way that you write JavaScript“
- Achtung: jQuery **IST** JavaScript!



Einbinden von jQuery

- Einbinden einer einzigen JavaScript-Datei
- 2 Möglichkeiten:
 - Herunterladen und Verweis auf Datei (offline):

```
<script type="text/javascript"  
src="jquery.min.js"></script>
```

- Hotlink zur Datei (empfohlen):

```
<script type="text/javascript"  
src="http://code.jquery.com/jquery-  
1.10.2.min.js"></script>
```

Grundsätzliches

- Globale jQuery-Funktion: `$ ()`
- Parameter: beliebige CSS-Selektoren
- Rückgabe: jQuery-Objekt, das Methoden bereitstellt

- Beispiele:
 - `$ ("#id")`
 - `$ ("div.container")`
 - `$ ("input[type='text']")`

- Außerdem: `$` stellt weitere Funktionen bereit, z.B.
`$.isArray(value , array)`

DOM-Manipulation (1)

```
<div id="myDiv">Container</div>
```

- HTML-Inhalt eines Elements auslesen:

```
var inhalt = $("#myDiv").html();
```

- HTML-Inhalt eines Elements setzen:

```
$("#myDiv").html("<b>Neuer Inhalt</b>");
```

- **Beachte:** `.html()` als Getter- und Setter-Methode

DOM-Manipulation (2)

```
<div id="myDiv">Container</div>
```

- HTML-Elemente erzeugen und in DOM einfügen:

```
$( "#myDiv" ).after( "<span>Inhalt</span>" );
```

- Attribute auslesen/ändern:

```
$( "#myDiv" ).addClass( "container" );
```

```
$( "#myDiv" ).attr( "id", "neueID" );
```

- Weitere Beispiele:

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

Chaining

- Verkettung von Methodenaufrufen auf jQuery-Objekten
- Code effizienter und besser lesbar
- Beispiele:

```
$( "#myDiv" ).removeClass( "off" ).addClass( "on" );
```

```
$( "#myDiv" )  
    .css( "color", "#cccccc" )  
    .removeClass( "container" )  
    .attr( "id", "neueID" )  
    .append( "<span>Text</span>" );
```

DOM-Traversing

- Durchlaufen der DOM-Struktur mithilfe von Selektoren
- Selektieren/Filtern von Elementen
- Beispiele:
 - `$("#myDiv").next("div")`
Findet das erste `div`-Element nach dem Element mit der ID `myDiv`
 - `$("ul").find("li.item")`
Findet in allen `ul`-Elementen alle `li`-Elemente mit der Klasse `item`
 - Weitere Beispiele: <http://api.jquery.com/category/Traversing/>

Callback-Funktionen

- Wichtiges Konzept bei jQuery: Callback-Funktionen
- Anonyme Funktion wird übergeben und von jQuery aufgerufen

```
$( "li" ).each( function() {  
    alert( $( this ).text() );  
} );
```

`$(document).ready()`

- `$(document).ready()` garantiert, dass DOM-Dokument vollständig geladen ist und darauf zugegriffen werden kann
- Erst dann wird Callback-Funktion ausgeführt

```
$(document).ready(function() {  
    //JavaScript-Code  
});
```

Event-Handling (1)

- Event-Handler: Erkennen Ereignisse (z.B. Klick) und führen daraufhin eine festgelegte Aktion (= Callback-Funktion) aus
- Klick-Handler mit jQuery:

```
$( "#myID" ).click(function(e) { ... } );
```

Alternative:

```
$( "#myID" ).on("click", function(e) { ... } );
```
- **Achtung:** Event-Handler funktionieren nur für Elemente, die bereits im DOM existieren, wenn der Handler gesetzt wird!

Event-Handling (2)

- Handler für viele Events möglich:
click, change, focus, submit, keypress, ...
<http://api.jquery.com/category/Events/>
- Standard-Ereignisse verhindern mit `preventDefault()`:

```
$( "a" ).click(function(event) {  
    event.preventDefault();  
    alert("Link: " + $(this).attr("href"));  
});
```

AJAX mit jQuery

- jQuery bietet Funktionen für AJAX-Requests:
 - `$.ajax()`
 - `$.get()`, `$.post()`, ...
- Vorteile: Einfachheit, Lesbarkeit, Browserkompatibilität
- Beispiel:

```
$.ajax({  
    url: "...", //URL für Request, z.B. PHP-Script  
    success: function(data) { } //Aufruf bei Erfolg  
});
```
- **Beachte:** Erst in der Callback-Funktion (`success`) können empfangene Daten genutzt werden!

Beispiel

```
$.ajax({
  url: "example.xml",
  type: "GET",
  dataType: "xml",
  success: function(d){
    $(d).find("example").each(function(){
      var id = $(this).attr("id");
      var text = $(this).text();
      $("#result").append(
        "<div>" + id + ": " + text +
" </div>"
      );
    });
  }
});
```

Beispiel: Kurzform

```
$.get("example.xml", function(d){
    $(d).find("example").each(function(){
        var id = $(this).attr("id");
        var text = $(this).text();

        $("#result").append(
            "<div>" + id + ": " + text + "</div>"
        );
    });
});
```

Übungsblatt 5

- **Thema: Filmsuche mit JQuery**
- **Bearbeitungszeit: 1 Woche**
- **Abgabe: 11.12.2013**

Danke! Fragen?