jQuery - Grundlagen

Theoretisches Wissen

jQuery-Versionen

- jQuery 1.x
- jQuery 2.x → keine Unterstützung für IE 6, 7, 8
- jQuery 3.x → keine Unterstützung für IE 6, 7, 8

jQuery nutzen

- Lokal herunterladen und einbinden
- CDN nutzen
- Kombination: CDN mit Fallback

Chaining – Aneinanderreihen

```
$("#beispiel").hide().fadeIn(4000).fadeOut(4000);
```

übersichtlicher:

```
$("#beispiel")
   .hide()
   .fadeIn(4000)
   .fadeOut(4000);
```

Anzeigen und Ausblenden

- `show()`
- `hide()`
- `fadeIn(), fadeOut()`
- `toggle()`

Auswählen mit JavaScript

- `document.getElementById("id")`
- `document.getElementsByClassName("klasse")`
- `document.getElementsByTagName("element")`
- `document.querySelector("selektor")`
- `document.querySelectorAll("selektor")`

Auswählen mit jQuery

- `jQuery("CSS-Selektor")` bzw. `$("CSS-Selektor")`
- Dann Methode anhängen, z.B. css()
  `$("li").css("background", "red")`
- keine for-Schleife notwendig um alle zu durchlaufen

Elemente auswählen

Selektor	Beispiel	Erläuterung
`$("#eins")`	`<div id="eins"></div>`	Element aufgrund von id auswählen
`$(".eins")`	`<div class="eins"></div>`	Element aufgrund von Klasse auswählen
`$("p")`	`<p>…</p>`	Elemente bestimmten Typs wählen
`$("*")`		Alle Elemente auswählen (nicht empfohlen, da ineffektiv)

Attributselektoren I

Beispiel	Funktion
`$("a[title]")`	Attribut title vorhanden
`$("a[title^=Artikel]")`	Attribut title vorhanden, Wert beginnt mit "Artikel"
`$("a[title$=März]")`	Attribut title vorhanden, Wert endet mit "März"
`$("input[type=text]")`	Attribut type mit Wert "text"

Attributselektoren II

Beispiel	Funktion
`$("a[title]~=Extern")`	Attribut title vorhanden, im Wert kommt das Wort "Extern" vor
`$("a[title*=Extern]")`	Attribut title vorhanden, im Wert kommt "Extern" vor
`$("input[type!=text]")`	Attribut type vorhanden, Wert nicht gleich "text" (Ergänzung von jQuery)

Selektoren kombinieren

Beispiel	Name	Funktion
`$("nav li")`	Nachfahren-Kombinator	Alle `li`-Elemente, die Nachfahre von `nav` sind
`$("article > header")`	Kind-Element	Alle `header`, die direktes Kind von `article` sind
`$("p + nav")`	Unmittelbar folgende Geschwister-Knoten	Alle `nav`, die unmittelbar auf `p` folgen
`$("header ~ p")`	Folgende Geschwister-Elemente	Alle `p`, die auf `header` folgen

Kindelemente auswählen I

Beispiel	Erläuterung
`$("li:first-child")`	Alle ersten Kindelemente
`$("p:first-of-type")`	Erstes seiner Art
`$("li:nth-child(x)")`	x-tes li-Element
`$("p:only-child")`	Einziges Kindelement

Kindelemente auswählen II

Beispiel	Erläuterung
`$("aside:only-of-type")`	Einziges seiner Art
`$("li:last-child")`	Letztes Kindelement
`$("p:last-of-type")`	Letztes seiner Art
`$("li:nth-last-child(x)")`	X-tes Element von hinten gezählt

jQuery – Filter I

Beispiel	Erläuterung
`$("li:eq(n)")`	das n-te Element – Zählung beginnt bei 0!
`$("li:lt(n)")`	Kleiner als (less then)
`$("li:gt(n)")`	Größer als (greater then)

jQuery – Filter II

Beispiel	Erläuterung
`$("li:first")`	liefert nur den ersten Treffer
`$("li:last")`	liefert nur den letzten Treffer
`$("li:even")`	liefert nur gerade Treffer
`$("li:odd")`	liefert nur ungerade Treffer

jQuery – Filter III

Beispiel	Erläuterung
`$("li:contains('text')")`	beinhaltet
`$("li:empty")`	liefert leere Elemente des Typs
`$("li:has('a')")`	liefert Kindelemente
`$("li:parent")`	liefert Elternelemente

jQuery – Filter IV

Beispiel	Erläuterung
`$("li").filter(":even")`	filtern
`$("li").find("a")`	Kindelemente finden
`end()`	Filter beenden

DOM durchlaufen

Beispiel	Erläuterung
`$("header").next(),` `$("header").nextAll()`	Das / die nächste(n) Element(e) auswählen
`$("#posts").children`	Kindelemente auswählen
`$("#page_header").parent(),` `$("#page_header").parents()`	Das / die Elternelement(e) auswählen
`$("#posts").closest()`	Das nächstgelegene Element auswählen

Richtige Strategie für Selektoren I

- Mehrmals Elemente nutzen → Variable einsetzen

```
$(".beispiel").show();
$(".beispiel").css("color", "orange");
```

- Besser:

```
var beispiel = $(".beispiel");
beispiel.show();
beispiel.css("color", "orange");
```

Richtige Strategie für Selektoren II

- Chaining nutzen

```
beispiel.show();
beispiel.css("color", "orange");
```

- Besser:

```
beispiel.show().css(" color",
"orange");
```

Richtige Strategie für Selektoren III

- ID-Selektoren sind am effektivsten (dann auch keine weiteren Angaben notwendig). Statt:

```
$("#outer #inner");
$("div#inner");
$(".outer-container #inner");
```

- Einfach:

```
$("#inner");
```

- Universellen Selektor (*) vermeiden

Richtige Strategie für Selektoren IV

- Statt verschachtelter Selektoren:

```
var productIds = $("#products div.id");
```

- Besser find() nutzen:

```
var productIds =
$("#products").find("div.id");
```

Richtige Strategie für Selektoren V

CSS-Selektoren mit Filter kombinieren

- Statt:

```
var p = $("p:eq(2)");
```

- Besser:

```
var p = $("p").eq(1);
```

Elemente einfügen I

Methode	Bedeutung
`append(inhalt)`	Der Inhalt wird innerhalb des Elementes am Ende angefügt.
`after(inhalt)`	Der Inhalt wird nach dem Element angefügt.
`prepend(inhalt)`	Der Inhalt wird innerhalb des Elementes am Anfang angefügt.
`before(inhalt)`	Der Inhalt wird vor dem Element angefügt.

Elemente einfügen II

append()	appendTo()
`$("nav").append("<p>a</p>")` Zuerst **Wo**, dann **Was**.	`$("<p>a</p>").appendTo("nav")` Zuerst **Was**, dann **Wo**.

Entsprechungen

- append() → appendTo()
- prepend → prependTo()
- before → insertBefore()
- after → insertAfter()

Ausmaße

- height(), width()

  Höhe/Breite ohne padding und border

- innerHeight(), innerWidth()

  Höhe/Breite mit padding

- outerHeight(), outerWidth()

  Höhe/Breite mit padding und border

Position

- `position()` – Position im Verhältnis zum Elternelement ermitteln (nur lesen)
- `offset()` – Position im Verhältnis zum Dokumentroot
- `scrollLeft(), scrollTop()` – horizontale / vertikale Scrollposition

Events

Kurzform	ausführlich

`$("#bsp").click(function(){`
`  alert("klick");`
`});`
	`$("#bsp").on("click",function(){`
	`  alert("klick");`
	`});`

Ereignistypen I

- **Browser-Ereignisse**
  error(), resize(), scroll()
- **Dokument laden**
  load(), ready(), unload()
- **Tastaturereignisse**
  keydown(), keypress(), keyup()

Ereignistypen II

- **Mausereignisse**
  click(), contextmenu(), dblclick(), hover(), mousedown(), mouseenter(), mouseleave(), mousemove(), mouseout(), mouseover(), mouseup(), toggle()*

*deprecated

Ereignistypen III

- Formularereignisse
  blur(), change(), focus(), focusin(), focusout(), select(), submit()