

# ***jQuery und modern WebFrameworks***

Trainer: Georg Praßl BSc.

## ➤ jQuery - Grundlagen

- Was ist jQuery
- jQuery einbinden
- Allgemeine Syntax
- Selektoren

## ➤ jQuery – Event Methoden

## ➤ jQuery – Effects

## ➤ jQuery – DOM Manipulation

## ➤ Modern Webdevelopment

- Angular vs React
- Angular example

- Ist eine JavaScript Library
- Hat JS großartig vereinfacht
- „Write less, do more“
- Macht aus vielen Zeilen Code oft eine einzelne Zeile
- jQuery kann
  - HTML/DOM manipulation
  - CSS manipulation
  - HTML event methods
  - Effects and animations
  - AJAX

## ➤ Zwei Optionen

- Download Files → jQuery.com
  - Vorteil
    - o Keine Probleme mit Datenschutz und Tracking
  - Nachteil
    - o Wird bei jedem Seitenaufruf komplett geladen
- CDN → <https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js>
  - Vorteil
    - o Wenn User bereits irgendwo anders geladen hat, ist die Loading Time schneller (Caching)
  - Nachteil
    - o Google tracked alles

- Mit jQuery werden HTML Elemente abgefragt (query) und auf diesen Elementen etwas ausgeführt
- `$(selector).action()`
  - `$` → definiert jQuery
  - `(selector)` → Abfragewert um Element zu finden
  - `Action()` → jQuery action welche auf Elementen ausgeführt wird

- `$(document).ready(function() { //Hier jQuery methoden })`
- `$(function() { // Hier jQuery Methode}`
- Beide Methoden gleich
- Benötigt man damit jQuery erst auf Elemente zugreift wenn Document fertig geladen.
- Z.B.:
  - Versuche ein Element das noch nicht geladen ist zu verstecken.
  - Versuche die Größe eines Bildes abzufragen.

- Grundsätzlich verwendet jQuery CSS Selektoren
- Beispiele
  - `$(„p.description“)`
    - Holt alle p Elemente mit der Klasse description
  - `$(„:button“)`
    - Holt alle button Elemente und input Elemente mit dem Typ „button“
- Query Selector Tester
  - <https://www.w3schools.com/jquery/tryselect.asp>

- Alle Aktionen die ein User auslösen kann sind events
- Events werden abgefeuert
  - Das keyup Event wird abgefeuert, in dem Moment wenn man von der Taste wieder runter geht
- Vier Haupttypen
  - Mouse Events
  - Keyboard Events
  - Form Events
  - Document/Window Events



- Die meisten DOM Events haben auch eine passende jQuery Methode
  - `$(„div“).click();`
- Häufig verwendete jQuery Event Methoden
  - `$(document).ready()`
  - `Click()`, `dbclick()`
  - `Mouseenter()`, `mouseleave()`
  - `Mousedown()`, `mouseup()`
  - `Hover()` → Achtung ist eine Kombi aus `Mouseenter` und `mouseleave`, benötigt also auch zwei auszuführende Methoden

## ➤ Focus(), blur()

## ➤ On()

- Kann einen oder mehrer Eventhandler für dieses Element entgegennehmen

```
➤ $("p").on({  
  mouseenter: function(){  
    $(this).css("background-color", "lightgray");  
  },  
  mouseleave: function(){  
    $(this).css("background-color", "lightblue");  
  },  
  click: function(){  
    $(this).css("background-color", "yellow");  
  }  
});
```

- Um HTML Elemente mit einen Effekt auszustatten kann man die jQuery Methoden
  - Hide
  - Show
  - Toggle
  - Slide
  - Fade
  - Animate
- verwenden

## ➤ Get Content

- Text()
- Html()
- Val()

## ➤ Get Attributes

- Attr()

## ➤ Set Content (Callback)

- Text()
- Html()
- Val()

## ➤ Set Attributes (Callback)

- Attr()

## ➤ Add Elements

- Append()
- Prepend()
- After()
- Before()

## ➤ Remove Elements

- Remove(filterSelector?)
- Empty()

- Es ist möglich mit jQuery
  - `addClass()`
  - `removeClass()`
  - `toggleClass()`
  - `Css()`

- Mit AJAX kann man Daten zwischen Server und Webseite austauschen.
- AJAX → Asynchronous JavaScript and XML
- Mit AJAX kann man
  - Text, HTML, XML oder JSON
- über
  - GET und POST
- austauschen
- Neuere Alternative → Fetch API
  - [https://developer.mozilla.org/en-US/docs/Web/API/Fetch\\_API/Using\\_Fetch](https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API/Using_Fetch)

- `$(selector).Load(URL,data,callback)`
  - Lädt Daten von externer Quelle und übergibt sie dem ausgewählten Element
- `$.Post(url, data, callback)`
  - Kann Daten von Externer Quelle laden und schreiben
- `$.Get(url, callback)`
  - Kann Daten von Externer Quelle laden
- `$.ajax({name:value, name:value})`





# *jQuery vs JS*

```
//jQuery
$('#elementID');

// VanillaJS
// Alternative #1
document.getElementById("elementID");
// Alternative #2
document.querySelector("#elementID");
```

```
// jQuery
element.addClass('className');
element.removeClass('className');

// VanillaJS
element.classList.add('className');
element.classList.remove('className');
```

```
// jQuery
element.show();
element.hide();
```

```
//VanillaJS
element.style.display="none";
element.style.display="";
```

```
// jQuery
$(document).ready(() => {
    // code code code
})

// VanillaJS
document.addEventListener('DOMContentLoaded',function(){
    // code code code
})
```

```
// jQuery
element.remove();

// VanillaJS
element.remove();
```

```
// jQuery
$.ajax({
    url: '/path',
    type: 'GET'
    success: (data) => {
        console.log(data)
    }
})

// VanillaJS
// Alternative #1
fetch('path')
    .then(response => response.text())
    .then(data => console.log(data))
// Alternative #2 (in an async function)
const response = await fetch('path');
const data = response.text()
console.log(data)
```

- Plain JS
- <https://animate.style/>
- Moderne Frameworks
- Bootstrap
- Interessanter Artikel
  - <https://thelicato.medium.com/jquery-is-useless-in-2022-65f5bab3177>

Vergleich von modernen Frameworks

# ***MODERNE FRAMEWORKS***

- Webentwicklung schreitet sehr schnell voran
- Wie erstellt man schnell und einfach schöne Webseiten ohne CMS-Systeme
- Packages kann man gut updaten
- Projekte somit immer auf neuesten Stand

- Architektur → Komponentenbasierte Architektur
- Open Source
- Beliebt
- Leicht für Responsive Design
- Rendering → Rendern sehr effizient auf C/S
- Leistung → Beide bieten eine ähnliche Leistung
- Leicht zu aktualisieren

## Angular

- Framework
- Real DOM
- Bidirektionale Datenbindung
- Typescript
- MVC-Modell
- Testen automatisch möglich

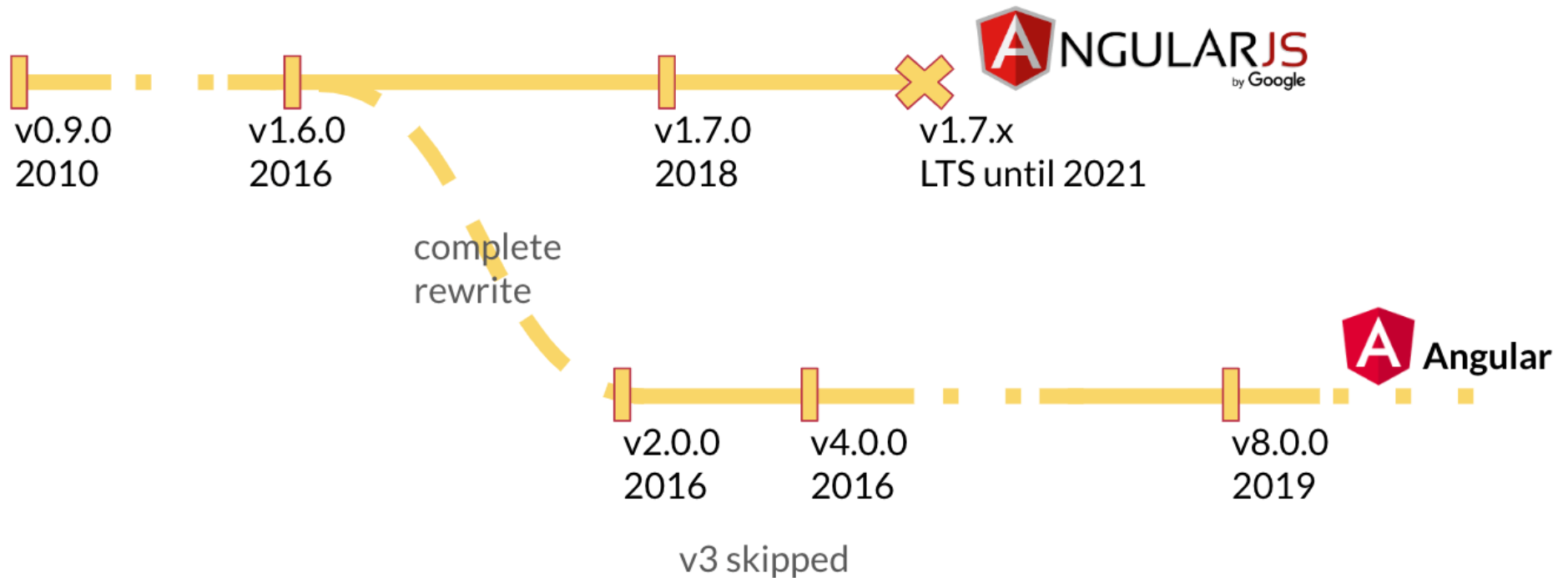
## React

- Library
- Virtuelles DOM
- Unidirektionale Verbindung
- Javascript
- Virtuelles DOM
- Zum Testen benötigt man zusätzliche Tools

<https://kinsta.com/de/blog/angular-vs-react/>

- `npm install -g @angular/cli`
- `Ng new App-Name`
- `Cd App-Name`
- `Ng add @angular/material`
- `Ng serve --open`

# Angular mehr im Detail





In general, you can expect the following release cycle:

- A major release every 6 months
- 1-3 minor releases for each major release
- A patch release almost every week

”

- Tag 1
  - CSS
  - CSS Frameworks
- Tag2
  - JavaScript
  - Typescript
- Tag3
  - jQuery
  - Angular

- Apps.egos.co.at
- Hier auch die Zertifikate downloadbar