

# Loading jQuery

jQuery is ready! Go to the next slide please..

# Dienstmededeling

# Skillstaak 4 is hier!

- 2 punten op taak AnySurfer
- 3 punten op Photoshop

# Skillstaak 4 is hier!

- 2 punten op taak AnySurfer
- 3 punten op Photoshop

## AnySurfer

- 13 December: Auditorium 1
- beoordeel twee site op 15 toegankelijkheidscriteria
- **Help AnySurfer toegankelijkheid te verbeteren!**

# Skillstaak 4 is hier!

## Photoshop

### Deel 1

Ga op reis met Photoshop!



# Skillstaak 4 is hier!

## Photoshop

### Deel 2

Maak je eigen startup!  
*(PS: ik zoek een job, moest je echt een startup hebben)*





# Wat is jQuery

- JavaScript **framework**
- Gebruikt op **80%** van alle websites
- Doel: JS **korter** en **makkelijker** maken





# JavaScript

**Zeg nooit zomaar Java tegen JavaScript**

# JavaScript

- 1995: Brendan Eich begint aan programmeertaal voor Netscape Navigator
- Doel: **interactief** web

# JavaScript

- 1995: Brendan Eich begint aan programmeertaal voor Netscape Navigator
- Doel: **interactief** web
- Sun's Java kreeg populariteit
- Netscape: volg de trend

# JavaScript

- 1995: Brendan Eich begint aan programmeertaal voor Netscape Navigator
- Doel: **interactief** web
- Sun's Java kreeg populariteit
- Netscape: volg de trend
- Mei 1995: 10 dagen later: eerste prototype "**Mocha**"
- Publieke naam: LiveScript

# JavaScript

- 1995: Brendan Eich begint aan programmeertaal voor Netscape Navigator
- Doel: **interactief** web
- Sun's Java kreeg populariteit
- Netscape: volg de trend
- Mei 1995: 10 dagen later: eerste prototype "**Mocha**"
- Publieke naam: LiveScript
- September 1995: *JavaScript*

# Java?

## Java

```
String[] testStrings = {"een", "twee"};  
Array.sort(testStrings, (s1,s2) -> s1-s2);
```

## JavaScript

```
var testStrings = ["een", "twee"];  
testStrings.sort(function(s1, s2){return s1-s2})
```

# De naam "JavaScript"

- Pure marketing van Netscape
- Veel kritiek...

# De naam "JavaScript"

- Pure marketing van Netscape
- Veel kritiek...
- Internet Explorer: **JScript**



# De naam "JavaScript"

- Pure marketing van Netscape
- Veel kritiek...
- Internet Explorer: **JScript**
- November 1996: JavaScript is ECMA Standard
- "ECMAScript"

# De naam "JavaScript"

- Pure marketing van Netscape
- Veel kritiek...
- Internet Explorer: **JScript**
- November 1996: JavaScript is ECMA Standard
- "ECMAScript"
- Vandaag: ECMAScript en JavaScript zijn synoniemen

# JavaScript laden

## Voor </body>

```
<body>
  <script>
    alert("Hello World")
  </script>
</body>
```

## In <head>

```
<head>
  <link [...]>
  <script>
    alert("Hello World")
  </script>
  <meta [...]>
</head>
```

# JavaScript laden

## In tag

```
<body>
  <script>
    alert("Hello World")
  </script>
</body>
```

## In file

```
<body>
  <script src="file:///home/runner/v
</script>
</body>
```

```
alert("Hello World")
```

# JavaScript in het kort

```
// Statements can be terminated by ;  
doStuff();  
  
// There's also a boolean type.  
true;  
false;  
  
// Strings are created with ' or ''.  
'abc';  
"Hello, world";
```

Bron: <https://learnxinyminutes.com/docs/javascript/>

# JavaScript in het kort

```
// Variables are declared with the `var` keyword.  
var someVar = 5;  
// Arrays are ordered lists of values, of any type.  
var myArray = ["Hello", 45, true];  
  
// The `if` structure works as you'd expect.  
if (someVar == 3){  
    // evaluated if someVar is 3  
} else {  
    // evaluated if it's not 3  
}
```

Bron: <https://learnxinyminutes.com/docs/javascript/>

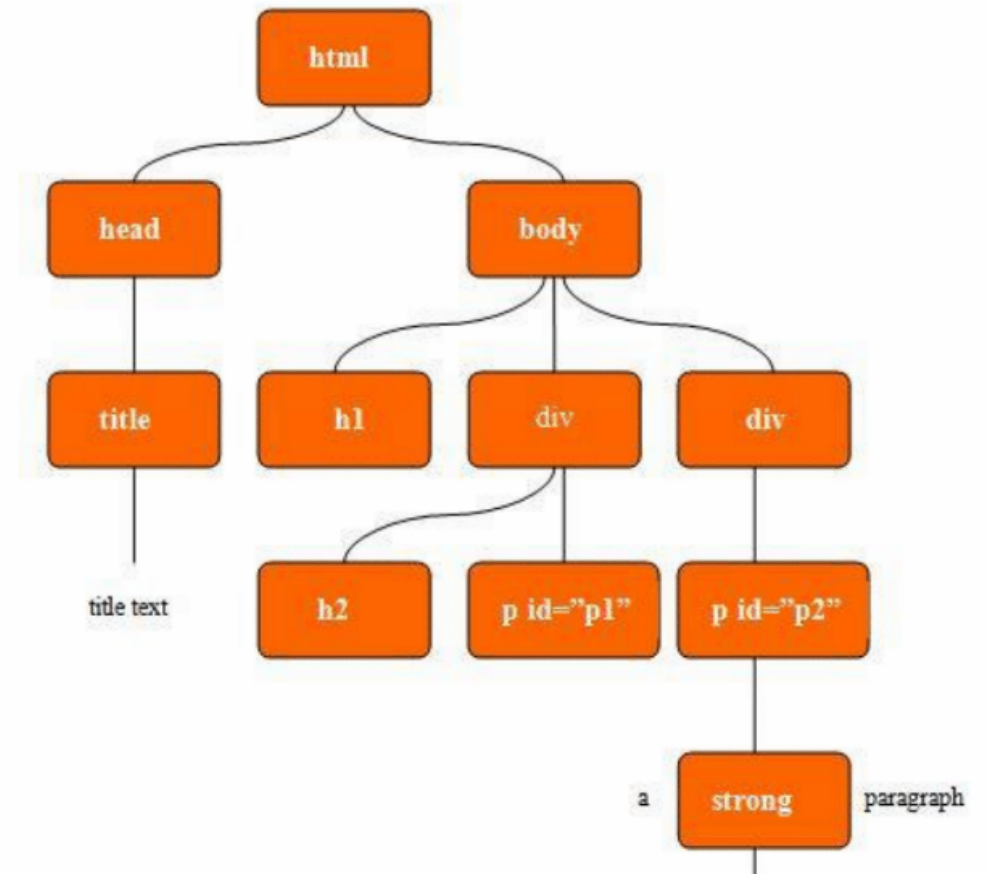
# JavaScript in het kort

```
// JavaScript functions are declared with the `function` keyword.  
function myFunction(thing){  
    return thing.toUpperCase();  
}  
myFunction("foo"); // = "FOO"  
  
// Function objects don't even have to be declared with a name  
// those are anonymous functions  
setTimeout(function() {  
    // this code will be called in 5 seconds' time  
}, 5000);
```

Bron: <https://learnxinyminutes.com/docs/javascript/>

# Interactie HTML?

- Document Object Model





# Document Object Model

## Inhoud element met ID aanpassen

```
<html>
<body>
  <script>
    document.getElementById("demo").innerHTML = "Hello World!";
  </script>
</body>
```

# Document Object Model

## Inhoud elementen met een klasse aanpassen

```
<html>
<body>
  <script>
    var demos = document.getElementsByClassName("demo");
    for (var demo in demos) {
      if (demos.hasOwnProperty(demo)) {
        demo.innerHTML = "Hello World";
      }
    }
  </script>
</body>
```

**Dit kan simpler...**

# jQuery

- JS library specifiek voor DOM
- Gemaakt in 2006 door John Resig
- Maakt gebruik van **CSS selectors**

# jQuery

- JS library specifiek voor DOM
- Gemaakt in 2006 door John Resig
- Maakt gebruik van **CSS selectors**
- Vandaag nog de basis van complexere oplossingen

# jQuery

- JS library specifiek voor DOM
- Gemaakt in 2006 door John Resig
- Maakt gebruik van **CSS selectors**
- Vandaag nog de basis van complexere oplossingen
- Bootstrap <3 jQuery (zie semester 2)

# Waar vinden we jQuery?

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

# Waar vinden we jQuery?

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

```
<script src="file:///home/runner/work/slides-jquery/slides-jquery/script
```



# Waar vinden we jQuery?

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

```
<script src="file:///home/runner/work/slides-jquery/slides-jquery/script
```

## Liever een Content Delivery Network?

```
<script  
src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js">  
</script>
```

- Vaak server dicht bij
- Vaak in cache van andere sites!

# jQuery

```
<script src="file:///home/runner/work/slides-jquery/slides-jquery/scri
<script src="file:///home/runner/work/slides-jquery/slides-jquery/scri
<script>
    //mijn lokaal script
    $('demo').text('Hello World');
</script>
</body>
```

# jQuery

```
<script src="file:///home/runner/work/slides-jquery/slides-jquery/scri
<script src="file:///home/runner/work/slides-jquery/slides-jquery/scri
<script>
    //mijn lokaal script
    $(' .demo' ).text( 'Hello World' );
</script>
</body>
```

**Kortere syntax dan (Vanilla) JS!**

# A dollar is what I need!

`$()` is kort voor `jQuery()`

```
function jQuery(selector) {  
    return { [jQuery Magic] }  
}
```

```
var $ = jQuery
```

# jQuery syntax

Geef alle **img** tags:

```
var images = $('img')
```

# jQuery syntax

Geef alle `img` tags:

```
var images = $('img')
```

Geef alle klasse `afbeelding`

```
var images = $('img')  
images.addClass('afbeelding')
```

# jQuery syntax

Geef alle `img` tags:

```
var images = $('img')
```

Geef alle klasse `afbeelding`

```
var images = $('img')  
images.addClass('afbeelding')
```

In het kort

```
$('img').addClass('afbeelding')
```

**Let op...**

**Als JavaScript een trein was zou het proberen te remmen maar toch elk station voorbij rijden...**



# Document Ready?

```
<script>
  $(document).ready(function(){
    // plaats alle code hier
    // pas hier zijn alle DOM elementen geladen!
  })
</script>
```

```
<script>
  $(function(){ // korte schrijfwijze
    // plaats alle code hier
  })
</script>
```

# jQuery Selectors

| Selector  | Beschrijving                                 |
|---|--|
| <code>\$('#item_1')</code>                      | Het element met id 'item_1'                  |
| <code>\$(<code>img[alt="jQuery"]</code>)</code> | Alle images met alt-tekst 'jQuery'           |
| <code>\$(<code>img[alt^="j"]</code>);</code>    | Alle images met alt-tekst die met 'j' begint |
| <code>\$(this)</code>                           | Het huidige HTML element                     |
| <code>\$(<code>p.foo</code>)</code>             | Alle p-tags met class foo                    |

# jQuery Selectors

| Selector  | Beschrijving                                 |
|---|--|
| <code>\$('#item_1')</code>                      | Het element met id 'item_1'                  |
| <code>\$(<code>img[alt="jQuery"]</code>)</code> | Alle images met alt-tekst 'jQuery'           |
| <code>\$(<code>img[alt^="j"]</code>);</code>    | Alle images met alt-tekst die met 'j' begint |
| <code>\$(this)</code>                           | Het huidige HTML element                     |
| <code>\$(<code>p.foo</code>)</code>             | Alle p-tags met class foo                    |

Combinatie CSS3 + jQuery specifiek

# DOM hiërarchie

## Descendant selector

| Selector                      | Beschrijving                               |
|-------------------------------|--|
| <code>\$('table tr')</code>   | Alle tr-tags binnen een table              |
| <code>\$('table td')</code>   | Alle td-tags binnen een table              |
| <code>\$('table td a')</code> | Alle linken in een td-tag binnen een table |
| <code>\$('ul *')</code>       | Alle tags binnen een ul-tag                |

# DOM hiërarchie

## Child selector

| Selector   | Beschrijving  |
|--|---|
| <code>\$( 'div &gt; p' )</code>                    | Alle p-tags die een child zijn van een div (rechtstreeks onder de div zitten)           |
| <code>\$( 'article &gt; section &gt; img' )</code> | Alle img-tags die child zijn van een section en de section is een child van een article |

# DOM hiërarchie

## Sibling selector (alle volgende)

| Selector                     | Beschrijving   |
|------------------------------|--|
| <code>\$( 'h3 ~ p' )</code>  | Alle p-tags die op hetzelfde niveau staan en volgen op een h3 tag  |
| <code>\$( 'li ~ li' )</code> | Alle li-tags die op hetzelfde niveau staan en volgen op een li tag |

# DOM hiërarchie

## Sibling selector (dadelijk aangrenzende)

| Selector                     | Beschrijving   |
|------------------------------|--|
| <code>\$( 'h3 + p' )</code>  | Alle p-tags die op hetzelfde niveau staan EN DADELIIJK volgen op een h3-tag  |
| <code>\$( 'li + li' )</code> | Alle li-tags die op hetzelfde niveau staan EN DADELIIJK volgen op een li-tag |

# addClass && removeClass

## addClass

```
$( 'img' ).addClass( 'afbeelding' )
```

## Voor

```
<img class="col-md-6">
```



# addClass && removeClass

## addClass

```
$( 'img' ).addClass( 'afbeelding' )
```

### Voor

```
<img class="col-md-6">
```

### Na

```
<img class="col-md-6 afbeelding">
```

# addClass && removeClass

## removeClass

```
$( 'img' ).removeClass( 'afbeelding' )  
$( 'img' ).removeClass( 'col-md-6' )
```

## Voor

```
<img class="col-md-6 afbeelding">
```

# addClass && removeClass

## removeClass

```
$( 'img' ).removeClass( 'afbeelding' )  
$( 'img' ).removeClass( 'col-md-6' )
```

### Voor

```
<img class="col-md-6 afbeelding">
```

### Na

```
<img class="">
```

# Event handling

Event = **actie** van de gebruiker (of browser)

```
$(function(){  
    $("#target").click(function() {  
        alert("Je klikte op mij!!!");  
    })  
})
```

Klik op me!

# Event handling

```
$("#target").click(function() {  
    $("#box").removeClass("blue");  
    $("#box").addClass("green");  
})  
$("#target").dblclick(function() {  
    $("#box").removeClass("green");  
    $("#box").addClass("blue");  
})
```

```
.blue {  
    background-color: blue;  
}  
  
.green {  
    background-color: green;  
}
```

Klik op me!

Box

**Er zijn enorm veel mogelijke events!**

**<https://oscarotero.com/jquery/>**

# Debug Helpers

## alert()

```
alert("Bericht");
```

alert

## console.log()

```
console.log("Bericht");  
console.log($("#console-demo"));
```

console.log

# Samengevat

- JavaScript -> programmeren in de browser
- jQuery -> JS Framework, focus op DOM `$("css selector")`
- Let op `$(document).ready()!!!`
- Aanpassen? `$( ).addClass() $( ).text()`
- Events -> Acties van gebruikers `$( ).click(function(){} )`
- `console.log()` -> Debug hulp `console.log("Bericht")`



**Slides:** <https://maartje.dev/slides-jquery/>

**PDF:** <https://maartje.dev/slides-jquery/presentation.pdf>