

**AJAX**

- AJAX = Asynchronous JavaScript and XML.
- Met AJAX kan op de achtergrond data geladen en getoond worden in de webpagina, zonder de hele pagina opnieuw te moeten laden.
- Als datatype wordt niet enkel XML gebruikt, maar dit kan even goed platte tekst of JSON zijn
- Een AJAX request kan enkel naar een webserver gebeuren.
- Bekende applicaties die gebruik maken van AJAX: Google Maps, Youtube, Facebook,...

**AJAX en jQuery**

- AJAX zonder jQuery maakt gebruik van het XMLHttpRequest object.
- Met de AJAX jQuery methodes kun je makkelijker tekst, HTML, XML of JSON op halen van de externe server en ook meteen de data laden in de HTML pagina.
- Met de AJAX jQuery methodes moet je geen rekening meer houden met de verschillen tussen de browsers.



Webdesign Javascript

jQuery.load()

jQuery.load()

```
$(selector).load(URL, data, callback);
```

- De URL wordt geladen.
- en in het geselecteerde element `$(selector)` getoond.
- Er kan eventueel `data` mee gestuurd worden naar de server
- De URL kan ook selectors bevatten om slechts een bepaald gedeelte van de URL te tonen.
- Een callback functie kan uitgevoerd worden wanneer de load beëindigd is, vooral voor errorafhandeling.
- Als veiligheidsmechanisme kan dit alleen uitgevoerd worden wanneer domein, protocol en poort van doelbestand en bronbestand gelijk zijn (= same-origin policy).
- Vermits URL op webserver moet staan, moet doelbestand dus ook op webserver staan en kunnen we dit dus niet zomaar (zonder vb. XAMPP) lokaal gebruiken.

https://www.w3schools.com/jquery/jquery_ajax_load.asp

Front-end Webdesign Javascript 1-12 jQuery AJAX

3



Webdesign Javascript

jQuery.get() and jQuery.post()

Difference HTTP GET and HTTP Post

GET

Data wordt meegegeven in de URL:

```
/test/demo_form.php?name1=value1&name2=value2
```

GET requests:

- can be cached
- remain in the browser history
- can be bookmarked
- should never be used when dealing with sensitive data
- have length restrictions
- should be used only to retrieve data

POST

Data wordt meegestuurd in het bericht zelf:

```
POST /test/demo_form.php HTTP/1.1
Host: w3schools.com
name1=value1&name2=value2
```

POST requests:

- are never cached
- do not remain in the browser history
- cannot be bookmarked
- have no restrictions on data length

Front-end Webdesign Javascript 1-12 jQuery AJAX

4



Webdesign Javascript

jQuery.get()

jQuery.get()

`$.get(URL, data, callback);` is een shorthand functies van

- Data van `URL` wordt opgehaald.
- Wanneer dit gebeurt is, wordt de `callback` functie uitgevoerd.
- Hier kun je de data veel meer uitrafelen.
- Ook errorafhandeling gebeurt in de callback functie.

```
$.ajax({
  url: url,
  data: data,
  success: success,
  dataType: dataType
});
```

Voorbeeld:

```
$.get( "ajax/test.html", function(data) {
  $(".result").html(data);
  alert("Load was performed.");
});
```

https://www.w3schools.com/jquery/jquery_ajax_get_post.asp en
<http://api.jquery.com/jquery.get/>

Front-end Webdesign Javascript 1-12 jQuery AJAX

5



Webdesign Javascript

jQuery.post()

jQuery.post()

`$.post(URL, data, callback);` is een shorthand functies van

- Data van `URL` wordt opgehaald
- er kan `data` worden meegegeven
- Wanneer dit gebeurt is, wordt de `callback` functie uitgevoerd.
- Ook errorafhandeling gebeurt in de callback functie.

```
$.ajax({
  type: "POST",
  url: url,
  data: data,
  success: success,
  dataType: dataType
});
```

Voorbeeld:

```
$.post( "ajax/test.html", function(data) {
  $(".result").html(data);
  alert("Load was performed.");
});
```

https://www.w3schools.com/jquery/jquery_ajax_get_post.asp en
<http://api.jquery.com/jquery.post/>

Front-end Webdesign Javascript 1-12 jQuery AJAX

6



Webdesign Javascript

jQuery.ajax()

jQuery.ajax()

`$.ajax({settings});`

Ipv te werken met verschillende parameters wordt gewerkt met een configuratieobject {}

```
$.ajax({
  url : "ajax/test.html",
  success: function(data) {
    $(".result").html(data);
    alert("Load was performed.");},
  error: function(jqXHR, textStatus, errorThrown){
    alert("Request failed: " + textStatus + ": " + errorThrown);}
});
```

<http://api.jquery.com/jQuery.ajax/>

Front-end Webdesign Javascript 1-12 jQuery AJAX

7



Webdesign Javascript

jQuery.ajax()

jQuery.ajax()

Ipv te werken met de callback functies zou je ook kunnen werken met de Promise methodes (afgeleid van Deferred object methodes) van dit jqXHR (=jQuery XMLHttpRequest) object:

```
$.ajax("ajax/test.html")
  .done(function(data) {           // success callback
    $(".result").html(data);
    alert("Load was performed." );
  })
  .fail(function(jqXHR, textStatus, errorThrown){ // error callback
    alert("Request failed: " + textStatus + ": " + errorThrown);
  })
  .then(function() {...});         // .done en .fail samen
```

Dit werkt zowel voor `$.ajax()` als voor `$.get()` en `$.post()`

<http://api.jquery.com/jQuery.ajax/>

Front-end Webdesign Javascript 1-12 jQuery AJAX

8



Webdesign Javascript

Voorbeeld Open Weather Map

We maken volgende HTML bestand:

```
<!DOCTYPE html>
<html lang="nl">
  <head>
    <meta charset="utf-8">
    <title>Het weer voorbeeld</title>
  </head>

  <body>
    <h2>Het weer in...</h2>
    <input type="text" id="plaatsNaam" placeholder="Geef een
plaatsnaam" />
    <button id="toonWeer">Toon het weer?</button>
    <div id="resultaat"></div>

  </body>
</html>
```

Het weer in...

Beringen

Beringen, BE 13graden zwaar bewolkt  3.1

Front-end Webdesign Javascript 1-12 jQuery AJAX

9



Webdesign Javascript

Voorbeeld Open Weather Map

➤ Om met Open Weather Map te mogen werken hebben we een api key nodig. Maak een API key aan via: <https://openweathermap.org/price>

➤ Om het weer op de te vragen van een plaats kun je bijvoorbeeld volgende URL gebruiken:
<http://api.openweathermap.org/data/2.5/weather?q=London&appid=xxxxxxxxxx>

We moeten dus de volgende stappen maken:

- Voeg jQuery en een javascript bestand toe aan het HTML bestand.
- Maak in het javascript bestand:
 - Een click event voor de knop
 - Wanneer op de knop gedrukt wordt, lees je de waarde van het invoerveld.
 - Test of er iets werd ingevuld. (Je kan nog niet testen of het juist is)
 - Doe een AJAX-call naar bovenstaande URL.
 - In de success toon je de data in de console. In de error toon je een error bericht.

Front-end Webdesign Javascript 1-12 jQuery AJAX

10




Webdesign Javascript

Voorbeeld Open Weather Map

- Je stelt vast dat de temperatuur (`data.main.temp`) tussen de 270 en 310 graden ligt. Dat komt omdat de temperatuur standaard in graden Kelvin weergegeven wordt. Wil je dit aanpassen dan kun je na de plaatsnaam `&units=metric` toevoegen.
- Je kunt ook een taalsupport toevoegen, vb.: `&lang=nl`
- We willen nu volgende gegevens tonen in onze resultaat:
 - `data.name`
 - `data.sys.country`
 - `data.main.temp`
 - `data.weather[0].description`
 - `data.weather[0].icon`
 - `data.wind.speed`
- Tip: Om het icoon te tonen kun je linken naar de icoontjes van openweathermap: <http://openweathermap.org/img/w/10d.png> waar 10d kan vervangen worden door de icoon-code

Het weer in...

Beringen, BE 13graden zwaar bewolkt  3.1