Fetch

Data ophalen

Data ophalen

Wanneer we data op een externe plaats opvragen, moet er een gestructureerde vraag gesteld worden (**httprequest**).

Hierop krijgen we dan een gestructureerd antwoord (**httpresponse**).

HttpRequest

Een HttpRequest bestaat uit enkele delen

- methode
- url
- header fields
- message/body

Methode

In principe bestaan er 9 methodes waarvan de onderstaande 4 het meest gebruik worden

- GET ⇒ wordt gebruikt om data op te vragen
- POST ⇒ wordt gebruikt om data toe te voegen
- PUT ⇒ wordt gebruikt om data aan te passen
- DELETE ⇒ wordt gebruikt om data te verwijderen

Daarnaast bestaat er ook nog OPTIONS, HEAD, CONNECT, TRACE, PATCH

URL

De url waarnaar we de request willen sturen

Bij GET request kan deze ook queryParameters bevatten

Header fields

Aan een request kan je extra info meegeven door middel van header fields.

Dit zijn steeds key-value paren.

Enkele interessante headers zijn:

- Content-type => welke content is er meegestuurd
- Authorization => kan gebruikt worden om credentials mee te geven die toegang geven tot de data

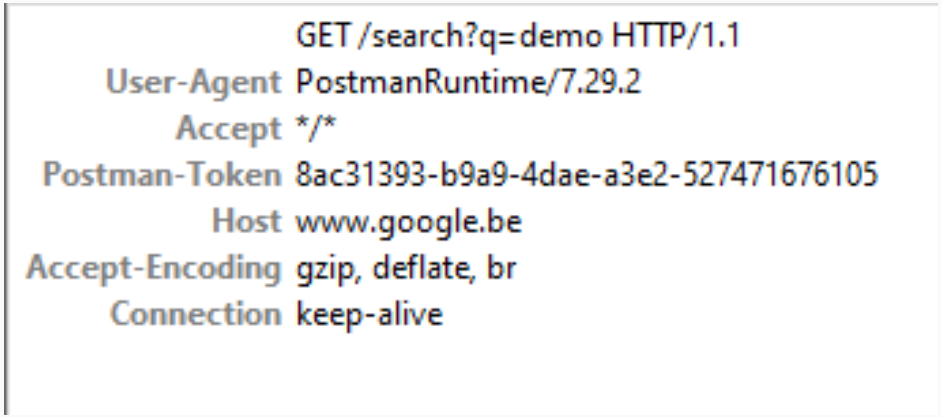
Andere headers kan je hier terug vinden

Message/body

Wanneer we een post of put request hebben, wordt er meestal data meegestuurd. Dit kunnen we doen in de body van de request.

Vaak is deze data in een JSON formaat.

Request Voorbeeld



HttpResponse

Net zoals een httpRequest hebben we ook een httpResponse met enkele onderdelen

- Status
- response headers
- body

Status

Elke response krijgt een **statuscode** en een **statusMessage**

Een statuscode geeft aan wat er fout gelopen is

- 1xx ⇒ informational
- 2xx ⇒ Success
- 3xx ⇒ Redirection
- 4xx ⇒ Client Error
- 5xx ⇒ Server Error

Response Headers

Net zoals er bij requests headers zijn, hebben response ook headers. Één van de belangrijkste hierbij is:

- Content-type => geeft aan in welk formaat de body staat

Andere headers kan je hier vinden.

Response Body

Bijna elke response bevat een responsebody. Dit kan in verschillende formatten.

- Blob => binary large object
- JSON
- html
- ...

HttpResponse voorbeeld


```
HTTP/1.1 200 OK
Content-Type: text/html; charset=ISO-8859-1
Date: Thu, 08 Sep 2022 13:12:02 GMT
Expires: -1
Cache-Control: private, max-age=0
Content-Security-Policy: object-src 'none';base-uri 'self';script-src 'nonce-6y3s7ElfH4jSRVkUzv782A' 'strict-dynamic' 'report-sample' 'unsafe-eval' 'unsafe-inline' https: http:;report-uri https://csp.withgoogle.com/csp/gws/xsrp
Content-Encoding: gzip
Server: gws
X-XSS-Protection: 0
X-Frame-Options: SAMEORIGIN
Transfer-Encoding: chunked
Set-Cookie: AEC=AakniGPInuX-FovqDQzpnuJWXGhIgkIKOqSpp9npWw6ptxHQ4S22vSE7oP8; expires=Tue, 07-Mar-2023 13:12:02 GMT; path=/; domain=.google.be; Secure; HttpOnly; SameSite=lax
Connection: keep-alive
```



```
<!doctype html><html lang="nl-BE"><head><meta charset="UTF-8"><meta content="/images/branding/googleg/1x/googleg_standard_color_128dp.png" itemprop="image"><title>demo - Google Zoeken</title><script nonce="6y3s7ElfH4jSRVkUzv782A">(function(){
document.documentElement.addEventListener("submit",function(b){var a;if(a=b.target){var c=a.getAttribute("data-submitfalse");a="1"===c||"q"===c&&!a.elements.q.value?!0:!1}else a=!1;a&&(b.preventDefault(),b.stopPropagation())},!0);document.documentElement.addEventListener("click",funct
var a=window.performance;window.start=Date.now();a:{var b=window;if(a){var c=a.timing;if(c){var d=c.navigationStart,f=c.responseStart;if(f>d&&f<=window.start){window.start=f;b.wsrt=f-d;break a}}a.now&&(b.wsrt=Math.floor(a.now()))}}window.google=window.google||{};var h=function(g){g&
var h=this||self;var k,l=null!=(k=h.mei)?k:1,n,p=null!=(n=h.sdo)?n:!0,q=0,r,t=google.erd,v=t.jsr;google.ml=function(a,b,d,m,e){e=void 0===e?2:e;b&&(r=a&&a.message);if(google.dl)return google.dl(a,e,d),null;if(0>v){window.console&&console.error(a,d);if(-2===v)throw a;b=!1}else b=!a||!a.messag
a.fileName;g&&(0<g.indexOf("-extension:/")&&(e=3),c+="&script="+b(g),f&&g===window.location.href&&(f=document.documentElement.outerHTML.split("\n")[f],c+="&cad="+b(f?f.substring(0,300):"No script found.")));c+="&jsel="+e;for(var u in d)c+="&",c+=b(u),c+="=",c+=b(d[u]);c=c+
var c=[],e=0;window.ping=function(b){-1==b.indexOf("&zx")&&(b+="&zx="+Date.now());var a=new Image,d=e++;c[d]=a;a.onerror=a.onload=a.onabort=function(){delete c[d]};a.src=b;}).call(this);</script><style>a{color:#1a0dab;text-decoration:none;tap-highlight-color:rgba(0,0,0,.10)}a:visited{c
var k=this||self,l=function(a){var b=typeof a;return"object"==b&&null!=a||"function"==b},m=function(a,b){function c(){}c.prototype=b.prototype;a.j=b.prototype;a.prototype=new c;a.prototype.constructor=a;a.l=function(d,e,f){for(var g=Array(arguments.length-2),h=2;h<arguments.length;h++)g[
b}).join(" "));function w(){var a=k.navigator;return a&&(a=a.userAgent)?a:""};var x=function(a){x[" "](a);return a};x[" "]=function(){};var y=-1!=w().indexOf("Gecko")&&!(-1!=w().toLowerCase().indexOf("webkit")&&-1==w().indexOf("Edge"))&&!(-1!=w().indexOf("Trident")||-1!=w().indexOf("MSIE"))&
!1}e||(d=null)}}else"mouseover"==b?d=a.fromElement:"mouseout"==b&&(d=a.toElement);this.relatedTarget=d;c?(this.clientX=void 0!==c.clientX?c.clientX:c.pageX,this.clientY=void 0!==c.clientY?c.clientY:c.pageY,this.screenX=c.screenX||0,this.screenY=c.screenY||0):(this.clientX=void 0!==a.clientX?
var a=document.getElementById("st-toggle"),b=document.getElementById("st-card");a&&b&&a.addEventListener("click",function(c){b.style.display=b.style.display?"":"none";c.preventDefault()},!1);}).call(this);</script></div></div><style>.nGphre{float:right;padding-left:16px}.UykT9d{border-radiu
Het begrip 'demo' wordt echter vaker gebruikt voor opnamen in eigen beheer. <span class="BNeawe"><a href="/url?q=https://nl.wikipedia.org/wiki/Demo_(muziek)&amp;sa=U&amp;ved=2ahUKEwippv7mooX6AhWo_rsIHfjrBS0QmhN6BAgIEAc&amp;usg=AOvVaw0KoVxuAhR1LbPNXKcivOw2"><s
var f=this||self,g=function(a){return a};
var h,k=function(){if(void 0===h){var a=null,b=f.trustedTypes;if(b&&b.createPolicy){try{a=b.createPolicy("goog#html",{createHTML:g,createScript:g,createScriptURL:g})}catch(c){f.console&&f.console.error(c.message)}h=a}else h=a}return h};var p=function(a,b){this.h=a===I&&b||"";this.g=m},q=fu
function W(){R=null;if(S){var a={client:"heirloom-srp",hl:hl,json:"t",callback:"hS",q:S};"undefined"!==typeof ds&&ds&&(a.ds=ds);var b=document;var c="SCRIPT";"application/xhtml+xml"===b.contentType&&(c=c.toLowerCase());c=b.createElement(c);b=y();b=z.exec(u(b).toString());var e=b[3]||"";
"":")&&c.setAttribute("nonce",d);document.body.appendChild(c);S=null;R=setTimeout(W,500)}}function X(){for(;K.firstChild;)K.removeChild(K.firstChild);N=[];O=-1;U(!1)}function Y(){var a=K.querySelector(".scs");a&&(a.className="");0<=O?(a=K.childNodes[O],a.className="scs",M=a.textConter
function b(a){for(a=a.target||a.srcElement;a&&"A"!==a.nodeName;)a=a.parentElement;a&&(a.href||"").match(/\/search.*[?&]tbm=isch/)&&(a.href+="&biw="+document.documentElement.clientWidth,a.href+="&bih="+document.documentElement.clientHeight)}document.addEventListener("clic
function y(a){if(!a||z(a))return 0;if(!a.getBoundingClientRect)return 1;var c=function(b){return b.getBoundingClientRect()};return A(a,c)?0:K(a,c)}function A(a,c){var b;a:{for(b=a;b&&b!==null!==b;b=b.parentElement)if("hidden"===b.style.overflow)break a;b=null}if(!b)return!1;a=c(a);c=c(b);return a.botton
function z(a){return"none"===a.style.display?!0:document.defaultView&&document.defaultView.getComputedStyle?(a=document.defaultView.getComputedStyle(a),!!a&&("hidden"===a.visibility||"0px"===a.height&&"0px"===a.width)):!1}
function K(a,c){var b=c(a);a=b.left+window.pageXOffset;c=b.top+window.pageYOffset;var f=b.width;b=b.height;var g=0;if(0>=b&&0>=f)return g;var p=window.innerHeight||document.documentElement.clientHeight;0>c+b?g=2:c>=p&&(g=4);if(0>a+f||a>=(window.innerWidth||document.docum
(function(){for(var a=(new Date).getTime()-window.start,c=Q(a),b=0,f=0,g=0,p=document.getElementsByTagName("img"),q=N?"&biw="+window.innerWidth+"&bih="+window.innerHeight:"",S=function(h,r){h.onload=function(){f=(new Date).getTime()-window.start;r&&++u==g&&(b=f,B())};h.o
h.responseStart);for(var r=h=0,E=0,F=0,G=0,H=0,U=0,k;k=O[U++];){var m=k[0],w=k[1],l=k[2];k=k[3];var V=0===m,x=m&8,J=m&4;(m=1===m||2===m||x&&!J)&&!x&&(w&&!k||++E,l&&++H);w&&(m&&k&&++F,x&&!k&&++r);V||x?!l||++G:w||++F}q+="&dimem="+E+"&imel="+H+("&imex="+F+"&im
window._setImagesSrc=function(e,f){function g(b){b.onerror=function(){b.style.display="none"};b.src=f}for(var c=0;c<e.length;++c){var a=e[c],d=document.getElementById(a)||document.querySelector('img[data-iid="'+a+'"]');d?(a=void 0,(null==(a=google.c)?0:a.setup)&&google.c.setup(d),g(d)):(
var a=e;2===(window.performance&&window.performance.navigation&&window.performance.navigation.type)&&window.ping("/gen_204?ct=backbutton&ei="+a);}).call(this);})();(function(){
var b=[function(){google.tick&&google.tick("load","dcl")}];google.dclc=function(a){b.length?b.push(a):a()};function c(){for(var a=b.shift();a;)a(),a=b.shift()}window.addEventListener?(document.addEventListener("DOMContentLoaded",c,!1),window.addEventListener("load",c,!1)):window.attachEvent
window.jsl=window.jsl||{};window.jsl.dh=function(a,b,f){try{var g=document.getElementById(a);if(g)g.innerHTML=b,f&&f();else{var c={id:a,script:String(!!f),milestone:String(google.jslm||0)};google.jsla&&(c.async=google.jsla);var h=a.indexOf("_"),d=0<h?a.substring(0,h):"",k=document.createElemer
google.jslm=x?2:1;})();(function(){(function(){google.csct={};google.csct.ps='AOvVaw2eqN2n4aoeujbJj02qkvYk\x26ust\x3d1662729122712708';})();})();(function(){(function(){google.csct.rd=true;})();})();(function(){(function(){google.csct.rl=true;})();})();(function(){
```

Fetch

Fetch

- Fetch is de moderne API om data extern op te halen
- Vroeger werd er gebruik gemaakt van XMLHttpRequest
- Fetch werkt op een reactieve manier

[codepen]

.then()

Fetch werkt asynchroon. Op het moment dat je fetch aanroept, heb je geen idee wanneer het resultaat binnen is.

Wanneer de resource opgehaald is wordt het resultaat doorgegeven aan de .then() functie doorgegeven.

Kijk na wat de status is van de response (ook 404 of 500 komt hier terecht) en maak gebruik van de juiste functie om de inhoud te returnen.

.then()

De mogelijke functies om de Body uit de response te halen geven allemaal een Promise terug. Dus ook hier moet je op het resultaat van deze functies wachten, waarvoor je een 2e .then() nodig hebt.

- response.json() ⇒ een object
- response.text() ⇒ een UTF-8 gecodeerde string
- response.blob() ⇒ een Blob
- response.arrayBuffer() ⇒ een ArrayBuffer
- response.formData() ⇒ een FormData object

.catch()

Met catch kunnen we errors opvangen wanneer er iets zou mislopen tijdens het uitvoeren van de fetch.

[codepen]

Options

Je kan fetch uitbreiden met opties en parameters zodat je online API's kan aanspreken zoals dat nodig is voor die API's.

Enkele options zijn:

- Method
- Headers
- Body

Voor een simpele get request hebben we dit meestal niet nodig

Post voorbeeld

[codepen]