



Klik of tik om tekst in te voeren.

## Gebruikershandleiding

Klik of tik om tekst in te voeren.

## Datum

13 oktober 2021

## Auteurs

Klik of tik om tekst in te voeren.

## Inhoudsopgave

Introductie .....	1
Starten met de Workshop OpenAPI's .....	2
Deel 1: OpenAPI bookmark inrichten .....	3
Deel 2: Informatie tonen op website .....	6
Deel 3: Geometrische en administratieve gekoppelde informatie toevoegen aan de OpenAPI bookmark .....	10
Deel 4: Gekoppelde informatie toevoegen aan website .....	14
Deel 5: Autocomplete functie toevoegen aan een website .....	17
Notities .....	18

## Introductie

Hoe kan een website worden uitgebreid met de slimme API's van KaartViewer? KaartViewer beschikt sinds de nieuwe versie over API's, die overal ingezet kunnen worden. Wil je bv. een download van een kaartlaag klaar zetten, dan kan dat via een KaartViewer API. In KaartViewer Beheer wordt bepaald welke API's beschikbaar zijn. We laten in een voorbeeld zien (Postcodechecker voor de Omgevingswet), hoe je zelf de API kunt toepassen. Via een stukje eenvoudige javascript code wordt dit toegelicht.

De workshop bestaat uit vijf onderdelen:

### Deel 1: OpenAPI bookmark inrichten

Voordat de OpenAPI gebruikt kan worden in een website, moet de OpenAPI bookmark worden ingericht in de beheeromgeving met een kaartlaag (bijvoorbeeld verblijfsobjecten). Het resultaat is een OpenAPI response gebaseerd op de identificatie van een verblijfsobject.

### Deel 2: Informatie tonen op website

Om informatie over de verblijfsobjecten te tonen op een website wordt de OpenAPI aanvraag toegevoegd aan geprogrammeerde code. Het resultaat is de informatie weergegeven in een tabel.

### Deel 3: Geometrische en administratieve gekoppelde informatie toevoegen aan de OpenAPI bookmark

Of een verblijfsobject binnen een beschermd gebied\* valt of dat een verblijfsobject valt binnen een monument\*, kan gecheckt worden aan de hand van een koppeling. Met OpenAPI is het mogelijk om de gekoppelde gegevens te tonen in de response.

### Deel 4: Gekoppelde informatie toevoegen aan website

De gekoppelde gegevens kunnen worden weergegeven in de website. Aan de hand van geprogrammeerde filters kan gekozen worden om bepaalde verblijfsobjecten te laten zien. Bijvoorbeeld alle verblijfsobjecten binnen een beschermd gebied of alle monumenten.

### Deel 5: Autocomplete functie toevoegen aan een website

Een autocomplete functie kan worden toegevoegd aan de website om bijvoorbeeld te checken of een bepaalde postcode en huisnummer vallen binnen een beschermd gebied of dat er sprake is van een monument.

\*De kaartlagen "beschermd gebieden" en "monumenten" bevat fictieve data.

## Starten met de Workshop OpenAPI's

Voordat we aan de slag gaan met het toevoegen van een OpenAPI bookmark loggen we in op de workshop omgeving en voegen we de service toe die we gaan gebruiken tijdens deze workshop.

1. Open <https://workshop.kaartviewer.nl/admin/>
2. Log in met de gebruikersnaam training en het wachtwoord training
3. Open de kaartenbakken 
4. Zoek de kaartenbak "Cursist [eigen nummer]" en ga naar de services
5. Klik op  om een service toe te voegen
6. Voeg de service <https://tutorials.kaartviewer.nl/geoserver/kid2021/wfs?> toe.
7. Geef de service de naam KID2021 Workshop API. Kies bij type service voor WFS. Zet de versie op 1.1.0 en selecteer het output formaat json.

### KID2021 Workshop API

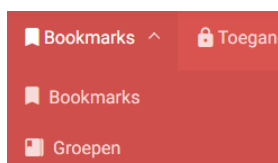
Naam	Service	Projectie
KID2021 Workshop API	WFS	EPSG:28992
URL		
https://tutorials.kaartviewer.nl/geoserver/kid2021/wfs?		URL OPNIEUW CONTROLEREN
Versie	Output formaat	Servicetype
1.1.0	json	vector
Certificaat locatie (optioneel)		Certificaat wachtwoord (optioneel)
D:\foldernaam\certificaat.cer		*****
Certificaat key locatie (optioneel)		
D:\foldernaam\certificaat.key		
Gebruiker (optioneel)	Wachtwoord (optioneel)	<input type="checkbox"/> Beveiligd ophalen
Gebruiker	Wachtwoord	
Printen		Informatie ophalen
<input type="checkbox"/> Print rechtstreeks (upload certificaat naar keystore)		<input checked="" type="checkbox"/> IN is functie beschikbaar
<input type="button" value="X VERWIJDEREN"/>		<input type="button" value="ANNULEREN"/> <input type="button" value="✓ OPSLAAN"/>

8. Publiceer alle beschikbare kaartlagen

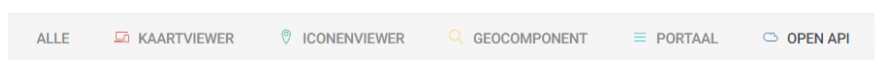
## Deel 1: OpenAPI bookmark inrichten

In het eerste deel van de cursus gaan we aan de slag met het aanmaken van een OpenAPI bookmark met verblijfsobjecten.

1. Ga naar de bookmarks



2. Kies voor de verschijningsvorm OpenAPI



3. Voeg een nieuwe OpenAPI bookmark toe via

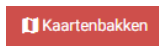


4. Geef de bookmark een naam en omschrijving en voeg je eigen cursist nummer toe. Kies bij de kaartenbak voor Cursist (eigen nummer). Er hoeft geen kopie gemaakt te worden. Geef bij Viewer aan dat het gaat om de OpenAPI verschijningsvorm en klik op opslaan.

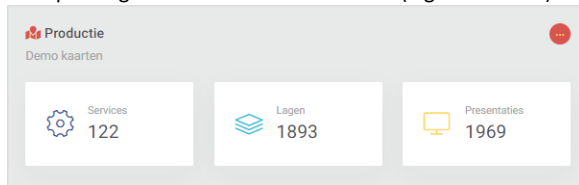
5. Voeg een nieuwe map toe aan de OpenAPI door te verplaatsen naar de OpenAPI bookmark map. Noem de map "Verblijfsobjecten".



6. Ga in het menu naar de kaartenbakken



7. Klik op de lagen in kaartenbak van Cursist (eigen nummer)



8. Zoek op “verblijfsobject” en klik op de alias “verblijfsobjecten ysselsteyn”.

Lagen										Zoeken: verblijfsobject	
ID	Alias	Laag	Service	Gepubliceerd	Attributen	Formulieren	Presentaties	Metadata	Geometry		
5	Verblijfsobjecten ysselsteyn	kiz2021 verblijfsobjecten_yssesteyn	KIZ2021 Workshop API	mike	1	1	1	1	1		

9. Voeg een nieuwe OpenAPI presentatie toe met de knop

+ TOEVOEGEN

Nieuwe presentatie aanmaken

Weergave naam:

Viewer:

ANNULEREN

+ TOEVOEGEN

10. Geef de presentatie de naam “Verblijfsobjecten” en kies voor een OpenAPI presentatie.

Om gebruik te kunnen maken van de OpenAPI moet er een primary key worden aangemaakt.

11. Ga naar de attributen en maak van het attribuut “identificatie” een primary key door het type te wijzigen.

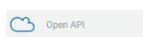
ATTRIBUTEN					Zoeken:	
ID	Naam	Attribuut	Type			
48	Bouwnummer	Bouwnummer	Tekst			
37	Functie	Functie	Tekst			
41	Gebruiksdoel	gebruiksdoel	Tekst			
36	Geometrie	geometrie	Geometrie			
38	Identificatie	identificatie	Primary Key			

12. Maak een formulier met alle verblijfsobjecten gegevens  
Voeg de attributen “Geom”, “Oppervlakte”, “Status”, “Gebruiksdoel”, “identificatie”, “Openbare ruimte” en “huisnummer” toe met

13. Voeg het formulier toe aan de OpenAPI verblijfsobjecten presentatie via informatie.

14. Ga naar bookmarks en zoek de zojuist aangemaakte OpenAPI bookmark op. Sleep de presentatie in de map “Verblijfsobjecten”.

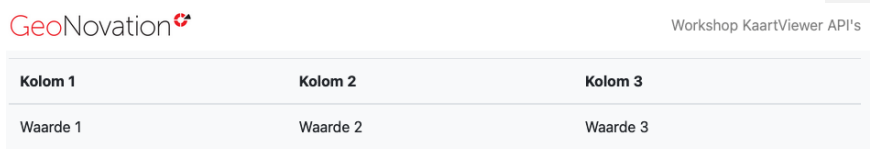
Het resultaat van de OpenAPI bookmark kunnen we bekijken via de knop



15. Open de 2<sup>e</sup> “Get” en vul de getallen 0984010000314476 in bij de sleutel (identificatie).  
Klik op “Execute” en controleer of er bij de response een antwoord terugkomt. Hierin zal je zien dat het formulier uit de presentatie wordt teruggegeven in de response.

## Deel 2: Informatie tonen op website

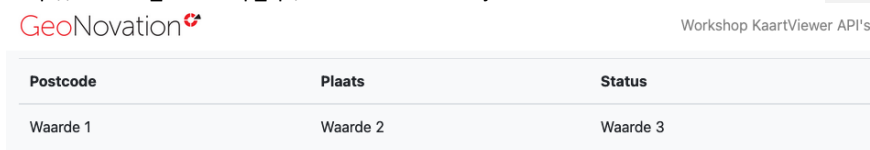
1. Verander {gebruikersnaam} na de gebruiker en open de link 'file:///C:/Users/{gebruikersnaam}/Documents/kid2021\_workshop\_api/index.html'. Bij het openen van deze link zie je een website en een tabel. Deze tabel is gevuld met statische data.



GeoNovation Workshop KaartViewer API's

Kolom 1	Kolom 2	Kolom 3
Waarde 1	Waarde 2	Waarde 3

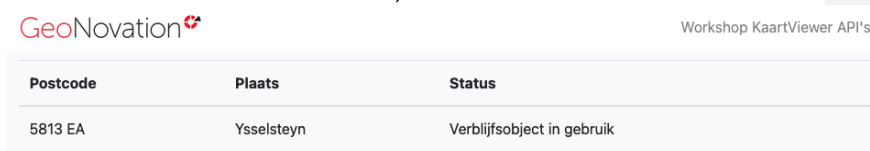
2. Open het programma 'Visual Studio Code'. Dit is het programma dat we gaan gebruiken om in te programmeren.
3. Klik op open folder en kies de folder 'kid2021\_workshop\_api'. Deze folder bestaat uit het publieke git project 'https://github.com/GeoNovationbv/kid2021\_workshop\_api'
4. Open het bestand 'index.html'. Hierin staat de code van hoe een tabel is opgebouwd. Pas de waardes 'Kolom 1', 'Kolom 2' en 'Kolom 3' aan en verander het in 3 attributen van het verblijfsobject. Klik op opslaan en ververs de pagina 'http://kid2021\_workshop\_api/index.html' en bekijk het resultaat.



GeoNovation Workshop KaartViewer API's

Postcode	Plaats	Status
Waarde 1	Waarde 2	Waarde 3

5. Ditzelfde doen we ook voor de waardes 1, 2 en 3.



GeoNovation Workshop KaartViewer API's

Postcode	Plaats	Status
5813 EA	Ysselsteyn	Verblijfsobject in gebruik

6. Deze data is nu statisch. We gaan nu de KaartViewer API gebruiken om een FeatureCollection te tonen in de tabel. Open het bestand 'opdracht2.js' in folder 'js'.

- Om een API op te halen gebruiken we de functie '\$.get()'. De url waarde is 'OPENAPIURL' en dit gaan we aanpassen. De waarde van deze URL kun je terugvinden in de OpenAPI website. Voer de 1<sup>e</sup> "GET" request uit en kopieer de waarde van 'Request URL' en vervang 'OPENAPIURL' door deze waarde.

```

Curl
curl -X 'GET' \
  'https://workshop.kaartviewer.nl/admin/rest/openapi/bookmark/20/domain/13/presentation/4/info?maxFeatures=10' \
  -H 'accept: */*'

Request URL
https://workshop.kaartviewer.nl/admin/rest/openapi/bookmark/20/domain/13/presentation/4/info?maxFeatures=10

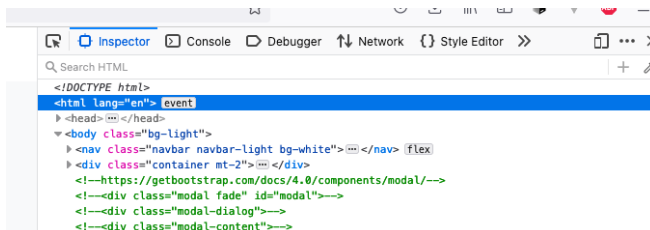
// Deel 2
//
var url = 'https://workshop.kaartviewer.nl/admin/rest/openapi/bookmark/20/domain/13/presentation/4/info?maxFeatures=10'
// // De data via de url ophalen
$.get( url, function( resultaat ) {

  // Door alle resultaten heen gaan
  // resultaat.features.map(function(feature) {

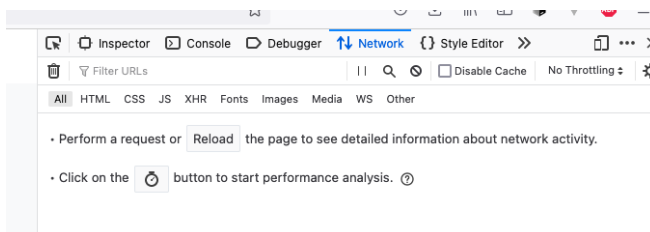
  // // De regels aan de tabel toevoegen
  // $('#table tbody').append('<tr><td></td><td></td><td></td></tr>')
  // });
});

```

- Als je nu de pagina ververs is er nog niks veranderd maar de API wordt wel aangeroepen. Om te testen of dit werkt gaan we de ontwikkelaar tools gebruiken. Druk op de knop F12.

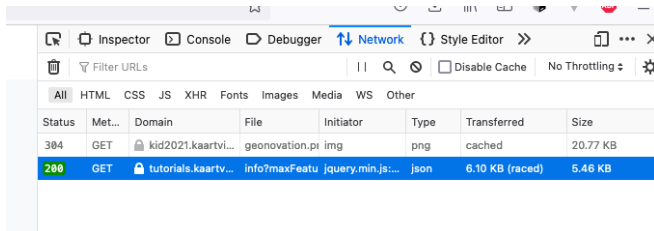


Dan komt er dit scherm aan de rechterkant naar voren. We hebben de 'Network' tabblad nodig om de API te bekijken. Klik op 'Network'.



De inhoud is nu nog leeg. Als je de website ververs worden er 2 aanroepen gedaan.





Als de 2<sup>e</sup> aanroep status 200 geeft is de API juist uitgevoerd.

- Om het resultaat in de tabel te krijgen gaan we terug naar de code. De inhoud van de FeatureCollection is nu in de variable 'resultaat' beschikbaar. Met de functie 'map' kunnen we door een lijst gaan.

Haal de `//` voor `console.log(feature)` weg.

In de code staat een `console.log(feature)`. Hiermee kun je kijken wat erin zit. Als je nu via F12 i.p.v. Network de Console opent zie je het resultaat.

Commented [NJ1]: Bij mij staat deze al aan, maar ik zie geen resultaat in de console

Commented [MV2R1]:



10. Nu gaan we de waardes in de tabel zetten. Haal de // voor de '\$('#table tbody')' weg. Als je de pagina ververst zie je nu dat er 10 lege rijen worden toegevoegd.

Nu gaan we de properties uit de feature selecteren. Vul na elke + de property in. Dit doe je daar feature.properties.{propertiennaam} in te vullen. {property} moet vervangen worden de de attribuut database naam en een spatie met een '+'.

Zie het eindresultaat in onderstaande afbeelding

Commented [NJ3]: Misschien nog extra verwijzen naar onderstaande afbeelding zodat cursisten niet per ongeluk letterlijk feature.properties.{postcode} typen bijvoorbeeld en zodat ze de juiste code invullen

```
// // De data van de API ophalen
$.get url, function( resultaat ) {

    // Door alle resultaten heen gaan
    resultaat.features.map(function(feature) {

        // De regels aan de tabel toevoegen
        $('#table tbody').append('<tr data-identificatie="' + feature.properties.identificatie + '"><td>' + feature.properties.postcode + '</td><td>' + feature.properties.woonplaats + '</td><td>' + feature.properties.status + '</td></tr>');

    });
};
```


Ververs de pagina en nu worden er 10 extra adressen data opgehaald.

Postcode	Plaats	Status
5813 EA	Ysselsteyn	Verblijfsobject in gebruik
5813CM	Ysselsteyn	Verblijfsobject in gebruik3
5813CM	Ysselsteyn	Verblijfsobject in gebruik3
5813CM	Ysselsteyn	Verblijfsobject in gebruik3
5813BD	Ysselsteyn	Verblijfsobject in gebruik3
5813BJ	Ysselsteyn	Verblijfsobject in gebruik3
5813BJ	Ysselsteyn	Verblijfsobject in gebruik3
5813CK	Ysselsteyn	Verblijfsobject in gebruik3
5813CK	Ysselsteyn	Verblijfsobject in gebruik3
5813AL	Ysselsteyn	Verblijfsobject in gebruik3
5813CR	Ysselsteyn	Verblijfsobject in gebruik3

11. Ga terug na index.html en verwijder de 1<sup>ste</sup> <tr></tr> in de <tbody> zodat er alleen dynamische data in komt en er in totaal 10 adressen zijn.

### Deel 3: Geometrische en administratieve gekoppelde informatie toevoegen aan de OpenAPI bookmark

In het derde deel worden beschermde gebieden en monumenten gekoppeld aan verblijfsobjecten. Ter illustratie kan de bookmark [Gekoppelde lagen API in KaartViewer](#) worden geopend om de benodigde lagen te bekijken.

1. Ga naar de kaartenbak van Cursist (eigen nummer). Zoek in de lagen op “bescherm” en klik op de alias “Beschermd gebied kid2021”.
2. Ga naar relaties en voeg een relatie toe 
3. Kies voor de optie “geometrie”. Het geometrische attribuut (geom/geometrie) staat ingesteld en kan gekoppeld worden aan een geometrisch attribuut uit een andere kaartlaag. Kies als koppellaag de verblijfsobjecten laag.

Gekoppelde laag toevoegen ×

ADMINISTRATIEF

GEOMETRIE

Relatie type ↓

Intersects

Attribuut

geom

Koppellaag:

Verblijfsobjecten yss€





op

Koppellaag attribuut:

geom

Intersects

OPSLAAN

ATTRIBUTEN	RELATIES/LIJSTEN	FORMULIEREN	PRESENTATIE
			
Relaties		Zoeken: 	
ID	Attribuut	Koppellaag	Koppellaag attribuut
 	2 geom (Geometrie)	1: Verblijfsobjecten ysselsteyn	geom (Geometrie)

- Ga naar formulieren om de geometrische gegevens en waarde van een beschermd gebied toe te voegen.

Geef het formulier een naam en voeg de twee attributen toe met →.

- Ga naar informatie en voeg het formulier toe aan de presentatie “Beschermd gebied kid2021” (klik op info en selecteer het formulier).

- Ga terug naar je eigen kaartenbak en zoek in de lagen op “monumenten”. Klik op de naam “Monumenten kid2021”.
- Ga naar de relaties/lijsten en voeg een nieuwe relatie toe. Ditmaal is het een administratieve relatie waarbij het attribuut “pandid” wordt gekoppeld aan de “pandidentificatie” uit de koppellaag genaamd “Verblijfsobjecten Ysselsteyn”.

ATTRIBUTEN	RELATIES/LIJSTEN	FORMULIEREN	PRESENTATIE
Relaties		Zoeken: <a href="#">+ TOEVOEGEN</a>	
ID	Attribuut	1: Koppellaag	Koppellaag attribuut
3	pandid (n)	Verblijfsobjecten ysselsteyn	pandidificatie (n)

- Ga naar formulieren en voeg aan dit formulier alle attributen toe.
- Voeg het monumenten-formulier toe aan de presentatie "Monumenten kid2021" op dezelfde manier als bij stap 5.
- Ga terug naar de relaties/lijsten en klik op de koppellaag "Verblijfsobjecten Ysselsteyn".
- Ga naar informatie en voeg de formulieren van de beschermde gebieden en monumenten toe.
- Ga naar de bookmark waarin de verblijfsobjecten laag staat en bekijk het resultaat door op de knop OpenAPI te drukken.

ATTRIBUTEN	RELATIES/LIJSTEN	FORMULIEREN	PRESENTATIE	INFORMATIE	AUTORISATIE	BOOKMARKS	METADATA
Presentaties		Bookmarks					
ID	Naam	1:					
7	Verblijfsobjecten						
6	Verblijfsobjecten ysselsteyn						
1 tot 2 van 2 resultaten							

- Herhaal stap 15 uit deel 1. Vul een identificatie in die behoort bij een verblijfsobject dat binnen een beschermd gebied valt én een monument is (TIP: bekijk de [KaartViewer](#)). Bijvoorbeeld 0984010001116764. De response van OpenAPI bevat de gekoppelde formulieren. Dit kan gecontroleerd worden met CTRL+ F monument\_type (attribuut) en CTRL + F waarde\_beschermd\_gebieden (attribuut).

Het is nu al mogelijk om te checken of het verblijfsobject binnen een beschermd gebied valt of een monument is. Scroll na het opzoeken van monument\_type of waarde\_beschermd\_gebieden naar beneden tot "Value". Bij het monument\_type is de value "Beeldbepalend pand". Voor het attribuut waarde\_beschermd\_gebieden is de value "beschermd".

Server response

Code Details

200 Response body

```

{
  "id": 1000,
  "geometry": {
    "type": "Point",
    "coordinates": [5.15, 51.5]
  },
  "displayname": "Hauptstadt der Schweiz",
  "type": "City",
  "population": 100000,
  "country": "Switzerland",
  "continent": "Europe",
  "timezone": "Europe/Zurich",
  "island": false,
  "islandchain": null,
  "action": "Set",
  "actionid": 1000,
  "actionname": "Set",
  "icon": "city",
  "width": 100,
  "height": 100,
  "imageurl": "city.png"
}

```

Download

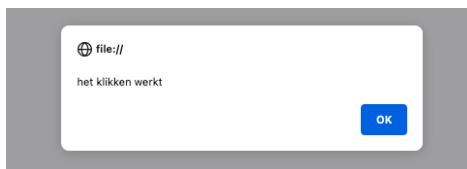
## Deel 4: Gekoppelde informatie toevoegen aan website

1. Nu gaan we de 2<sup>e</sup> API gebruiken om uitgebreide data van 1 feature op te halen. Eerst gaan we er voor zorgen dat er in de tabel geklikt kan worden. Als je met de muis over de tabel gaat zie je alleen een cursor maar geen klik icoontje. Dit gaan we met CSS aanpassen. Open index.html en voeg 'style="cursor:pointer"' in <tbody></tbody> toe met als resultaat.

```
<tbody style="cursor: pointer">
</tbody>
```

Als je nu de pagina ververs zie je dat de cursor in een handje is veranderd.

2. Nu gaan de klik functie toevoegen aan de tabel. Open opdracht4.js. Om een klik uit te voeren gebruiken de functie 'on('click')' die al ingevuld is. Om te testen of het klikken werkt haal je de '/' voor de alert('het klikken werkt') weg. Ververs de pagina en klik op een regel.



3. Nu het klikken werkt gaan we de 2<sup>e</sup> API gebruiken om de gekoppelde data op te halen. Vervang 'OPENAPI' door de waarde van de 2<sup>e</sup> API van de OpenAPI documentatie. De identificatienummer achter info/ kan weggehaald worden omdat dit dynamisch wordt ingevuld.

Commented [NJ4]: Alles na info/ kan weg

```
window.loadFeatureInfo = function (identificatie) {
    // URL van API 2
    var url = 'https://workshop.kaartviewer.nl/admin/rest/openapi/bookmark/23/domain/11/presentation/4/info/'
    // De data via de url ophalen
    $.get( url + identificatie, function( resultaat ) {
```

Ververs de pagina. Als je nu net zoals in deel 2 op F12 klikt en naar de tab 'Network' gaat en op een regel klikt wordt de 2<sup>e</sup> API aangeroepen.

- Open index.html en ga naar Deel 4. Hier is een model gemaakt. Open de link <https://getbootstrap.com/docs/4.0/components/modal/> en lees kort even de introductie. Dit is een notificatie die je zelf kan stijlen. Ga nu terug naar opdracht4.js. Voeg de code '\$("#modal").modal('show')' toe na de functie '\$("#modal-body tbody").html("")'.

Ververs de pagina en klik op een regel. Er wordt nu een modal geopend. De alert('het klikken werkt') hebben we niet meer nodig dus die regel kan verwijderd worden.

Feature gegevens	
Weergavenaam	Waarde
Weergavenaam	Waarde
Weergavenaam	Waarde

Sluiten

- Nu gaan we de formulieren tonen in de modal. We beginnen met de basis formulier. Om de informatie van het formulier te krijgen hebben de 1<sup>ste</sup> feature info nodig. Er staat in commentaar uitgelegd hoe de API is opgebouwd.

Van elke lijst hebben we de 1<sup>ste</sup> record nodig. Het 1<sup>ste</sup> record is bij programmeren is het 0<sup>de</sup> record. We gaan dit testen door met de console.log.

Voeg de code 'console.log(resultaat.mainTabs[0].features[0].featureInfo[0])' na de '\$("#modal").modal('show')' toe, ververs de pagina, open met F12 de browser en klik op console.

Als je nu klikt komt de informatie van het formulier er te staan.

- Verander de waarde 'featureInfo' in 'loadData('featureInfo')' door de waarde dat in console.log() staat. We gaan de titel van het formulier zetten. Haal de // voor de '\$("#modal-title-' + key).html(featureInfo.FormName)' weg en ververs de pagina.

Feature gegevens	
Verblijfsobjecten ysselsteyn	
Weergavenaam	Waarde
Weergavenaam	Waarde
Weergavenaam	Waarde

Sluiten



- Haal de tekens `'//'` voor `'$($('#modal-table-' + key + ' tbody')).append('<tr><td>' + '</td><td>' + '</td></tr>')` weg. Als je pagina ververs en op een regel klikt worden er lege regels toegevoegd.

Vervang het `'+'` tussen de 1ste `'<td>' + '</td>'` door de code `'+ attribute.DisplayName +'` en vervang `'+'` tussen de 2<sup>de</sup> `<td>' + '</td>'` door de code `'+ attribute.Value +'`.

Als de pagina ververs en op een regel klikt wordt nu de informatie van de basis formulier getoond.

- Om de gekoppelde gegevens op te halen hoeven we alleen de functie opnieuw aan te roepen en dan de 2<sup>e</sup> `featureInfo` op te halen. Kopieer `'loadData(resultaat.mainTabs[0].features[0].featureInfo[0], 0)'` en verander het naar `'loadData(resultaat.mainTabs[0].features[0].featureInfo[1], 1)'`. Nu wordt de titel van het formulier getoond en als er data is wordt de gekoppelde data weergegeven.

- Ditselfde doen we voor de laatste formulier door de code te kopiëren en te veranderen naar `'loadData(resultaat.mainTabs[0].features[0].featureInfo[2], 2)'`.

```
* 5. attributes: Lijst de attributen die in het formulier zit
*/

loadData(resultaat.mainTabs[0].features[0].featureInfo[0])
loadData(resultaat.mainTabs[0].features[0].featureInfo[1], 1)
loadData(resultaat.mainTabs[0].features[0].featureInfo[2], 2)
});
```

## Deel 5: Autocomplete functie toevoegen aan een website

1. In dit onderdeel gaan we de gegevens ophalen d.m.v. een adres autocomplete. Hier gaan we de locatieserver van PDOK gebruiken. Open index.html en zoek naar 'style:display: none' en haal deze code weg. Ververs de pagina en bekijk het resultaat.

Adres zoeken:

2. Als je nu in de tekstbalk typt gebeurt er nog niks. Open opdracht5.js. Hier is een autocomplete functie aangemaakt. KaartViewer beschikt over een API om de locatieserver van PDOK aan te roepen. Vervang de 1<sup>ste</sup> waarde 'OPENAPI' door 'https://workshop.kaartviewer.nl/admin/rest/routeapi/location/lng/0/lat/0'. Als je de pagina ververst en typt in de zoekbalk kun je naar alle adressen in Nederland zoeken.

Adres zoeken:	Kerkweg 1	
	1e Nijeveense Kerkweg, Nijeveen	
Postcode	Kerkweg 1, Roelofarendsveen	Status
5813AJ	Kerkweg 1a, Zeddam	Verblijfs
5813CA	Kerkweg 1-gar, Maarssen	Verblijfs
	Kerkweg 1A, Wekerom	
5813CA	Kerkweg 1, 6171ER Stein	Verblijfs
5813BJ	Kerkweg 1, 4585PJ Hengstdijk	Verblijfs
	Kerkweg 1, 6071AL Swalmen	
5813BB	Kerkweg 1, 4414AA Waarde	Verblijfs
	Kerkweg 1, 5096BK Hulsel	Verblijfs

3. Nu gaan we de informatie van het geselecteerde adres ophalen. Verander de waarde 'OPENAPI' door 'https://workshop.kaartviewer.nl/admin/rest/routeapi/location/lookup/'. Als je nu de pagina ververst, met F12 naar de console gaat en klikt op een adres zoekt met een huisnummer zie je identificatienummer.
4. Nu we de identificatienummer hebben kunnen we de OpenAPI aanroepen. Ga terug naar opdracht 4 en kopieer in de tabel klik 'loadFeatureInfo(identificatie)' en plak deze achter de console.log().

In de dataset die we gebruiken in de OpenAPI zitten niet alle adressen. Om te testen gaan we het adres 'Lovinckplein 7' gebruiken omdat dit adres in een beschermt gebied ligt en een monument is.

Nu wordt de modal geopend met de data van het gekozen adres uit de autocomplete.

## This image shows a full page of blank, lined paper. It features approximately 20 horizontal grey lines spaced evenly apart, typical of notebook paper. The lines extend across the entire width of the page, leaving small margins at the top and bottom. There are no vertical lines or other markings present.