

HTML

```
<!-- placeholder element voor de kaart -->
<div id="map"></div>

<script>

    //OpenStreetMap topografische achtergrond - optioneel
    const openstreetmap =
L.tileLayer('http://{s}.tile.osm.org/{z}/{x}/{y}.png');

    //PDOK BRT topografische achtergrond - optioneel
    const backgroundLayer =
L.tileLayer('https://service.pdok.nl/brt/achtergrondkaart/wmts/v2_0/gr
ijs/EPSG:3857/{z}/{x}/{y}.png', {
    attribution: '<a
href="https://creativecommons.org/licenses/by/3.0/nl/">CC BY 3.0</a>
Kadaster', // Attributie van kaartmateriaal
    maxZoom: 18, // Maximale zoom niveau van deze service
    minZoom: 8, // Minimale zoom niveau van deze service
    tileSize: 256,
    zoomOffset: 2
    });

    //PDOK landsgrens
    const landsgrens =
L.tileLayer.wms("https://service.pdok.nl/kadaster/bestuurlijkegebieden
/wms/v1_0?", {
    layers: 'Landgebied', //layername uit GetCapabilities
    format: 'image/png',
    colors: '255 165 0',
    transparent: true,
    attribution: '<a
href="https://creativecommons.org/publicdomain/zero/1.0/deed.nl">CC0
1.0</a> PDOK'
    });

    // Kaart aanmaken
    const centerPoint = [52.1009, 5.6463]; //Geografisch
middelpunt NL
    var map = L.map('map', {
    center: centerPoint, // Coördinaten van het startpunt van
de kaart
    layers: [landsgrens], // Ingeschakelde kaartlagen
    attributionControl: false,
    zoomControl:true,
    zoom: 7
    });

    //Niet de shorthand getJSON notatie, ASMX heeft contenttype
header nodig
    var requestUrl = "/test/GeoServices.asmx/GetData";
    $.ajax({
    type: "GET",
    url: requestUrl,
    data: "",
    contentType: "application/json; charset=utf-8",
    dataType: "json",
    success: function (msg) {
    const data = JSON.parse(msg.d);
    const featureLayer = L.geoJson(data.features, {
    onEachFeature: onEachFeature, //call function om
popup/tooltip in te richten
```

```

        style: styleFeature //call function om
conditioneel de kleur te bepalen
    });
    featureLayer.addTo(map); //voeg toe aan de kaart
},
error: function (xhr, status, error) {
    console.log(xhr);
    console.log(status);
    console.log(error);
}
});

function onEachFeature(feature, layer) {
    if (feature.properties && feature.properties.statnaam) {
        const displayText = feature.properties.statnaam + " -
Aantal inwoners: " + feature.properties.aantalInwonersTotaalT_1;
        layer.bindPopup(displayText);
        layer.bindTooltip(displayText);
    }
}

function styleFeature(feature) {
    return {
        fillColor:
getColor(feature.properties.aantalInwonersTotaalT_1),
        weight: 0.5, //lijndikte
        opacity: 1,
        color: '#333333', //border
        dashArray: '0', //stippellijn
        fillOpacity: 0.7
    };
}

function getColor(d) {
    var color = '#fee5d9';
    if (d > 5000) color = '#fcbba1';
    if (d > 10000) color = '#fc9272';
    if (d > 30000) color = '#fb6a4a';
    if (d > 50000) color = '#de2d26';
    if (d > 100000) color = '#a50f15';
    return color;
}

</script>

```