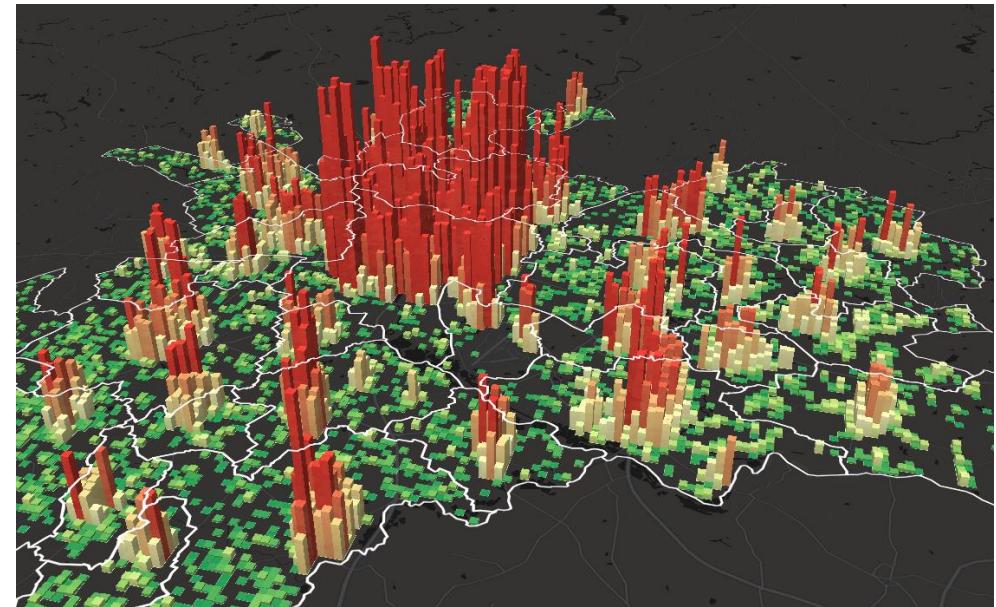




Introduction à la bibliothèque JavaScript MapboxGL

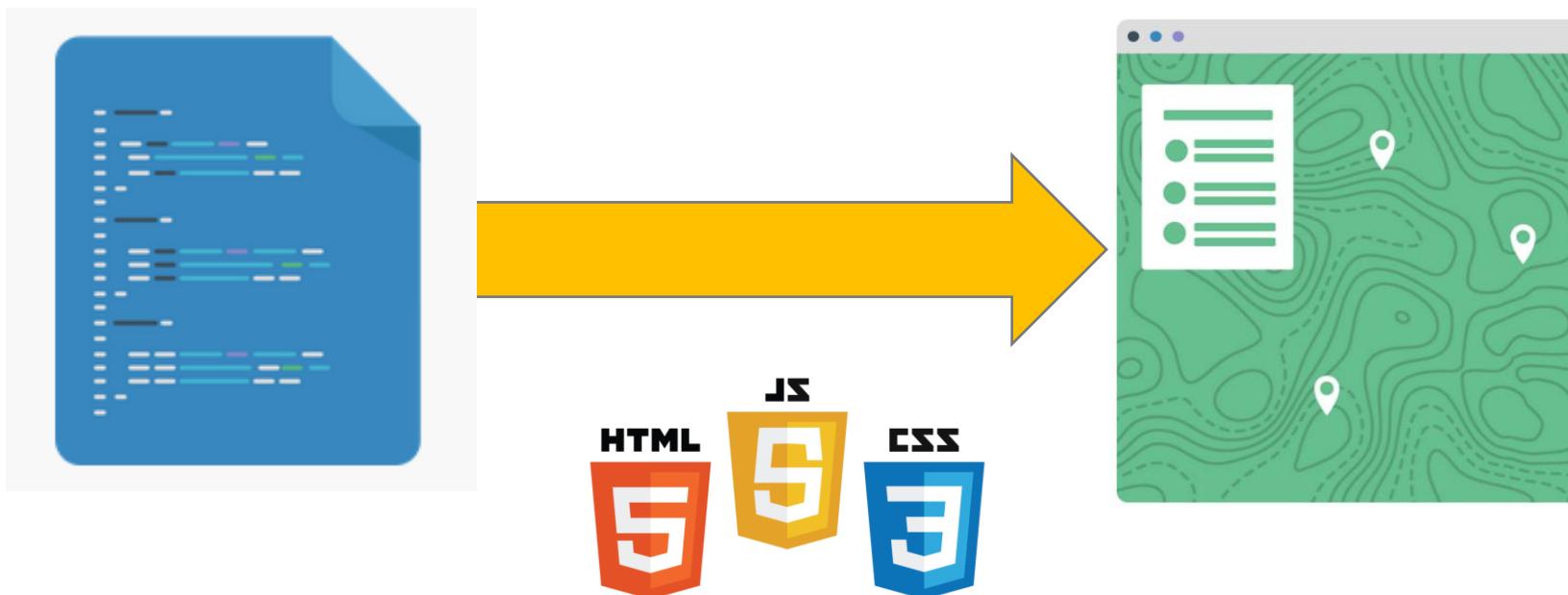


@Boris Mericskay



Objectifs atelier

- Publication de données spatiales sur le Web
- De la page HTML à l'application en ligne
- Familiarisation avec le Javascript, l'HTML et le CSS

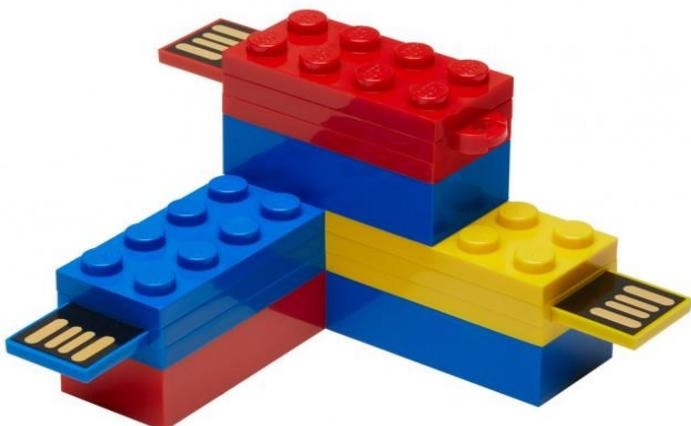


Prise en main de Mapbox GL



Coder = LEGO

- Vous allez à partir de maintenant « jouer » au LEGO en assemblant des lignes de codes pour construire des cartes sur le Web!



```
4 L.mapbox.accessToken = 'pk.eyJ1IjoibmluYW5vdW4iLCJhIjoiSkN4dn dmTSJ9.6plSt07M5AuAbDa601m54A';
5 var map = L.mapbox.map('map', 'mapbox.light').setView([48.11,-1.66], 13);
6 // Ajouts des WMS
7
8 var orthophotographie = L.tileLayer.wms('http://geobretagne.fr/geoserver/photo/wms?', {
9 format: 'image/png',
10 transparent: true,
11 layers: 'ortho-ouverte'
12 }).addTo(map);
13
14 var quartiers = L.tileLayer.wms('http://geobretagne.fr/geoserver/re nnesmetropole/wms?', {
15 format: 'image/png',
16 transparent: true,
17 layers: 'quartiers_vdr'
18 }).addTo(map);
```



MapBoxGL

MapBoxGL est une bibliothèque JavaScript libre de cartographie en ligne *open-source* utilisant le WebGL pour l'affichage (tuiles vectorielles, affichage 3D)

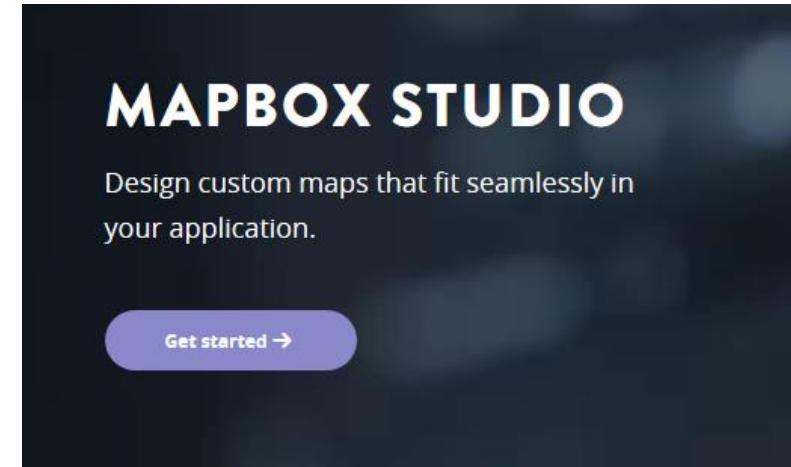
<https://docs.mapbox.com/mapbox-gl-js/api/>

Exemples: <https://docs.mapbox.com/mapbox-gl-js/example/>

Tutoriels : <https://docs.mapbox.com/help/tutorials/?product=Mapbox+GL+JS>

L'écosystème Mapbox

- Mapbox Studio
 - Créer des fonds de carte (*style*)
 - Héberger des jeux de données (*tilessets*)
 - Sous forme de tuiles vectorielles
- API MapboxGL.js
 - Bibliothèque JavaScript pour créer des cartes Web
 - La mobilisation de MapboxGL nécessite une clef d'accès = besoin d'un compte Mapbox



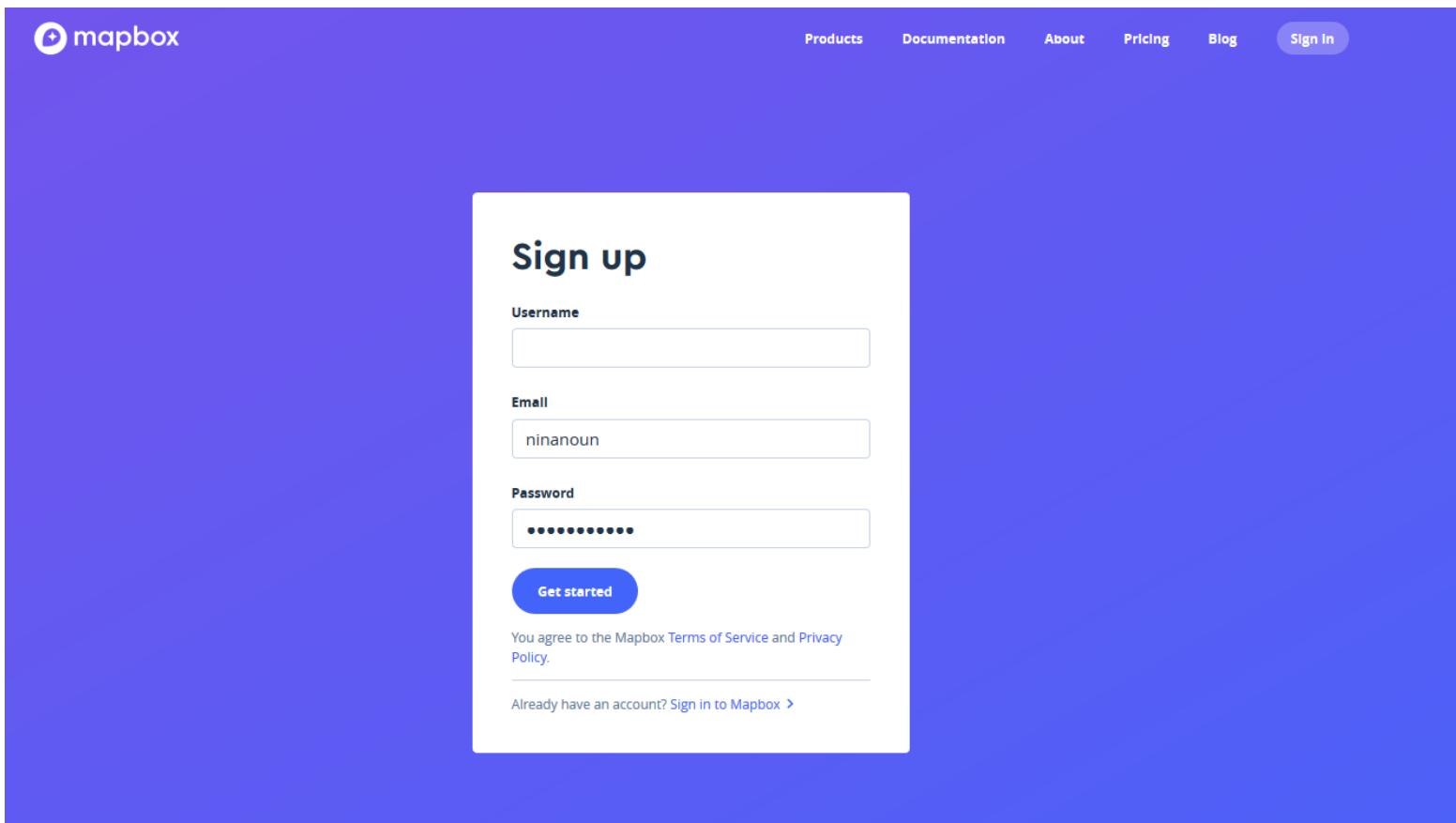
Mapbox GL JS

Current version: [mapbox-gl.js v0.44.0](#)

Mapbox GL JS is a JavaScript library that uses WebGL to render interactive maps from [vector tiles](#) and [Mapbox styles](#). It is part of the Mapbox GL ecosystem, which includes [Mapbox Mobile](#), a compatible renderer written in C++ with bindings for desktop and mobile platforms. To see what new features our team is working on, take a look at our [roadmap](#).

Créer un compte Mapbox

<https://www.mapbox.com/signup/>



Limitations d'un compte gratuit

PAY-AS-YOU-GO

Free to start \$0 up to	Web apps 50,000 map views / mo 50,000 geocode requests / mo 50,000 directions requests / mo 50,000 Matrix elements / mo	Mobile SDKs 50,000 monthly active users 50,000 geocode requests / mo 50,000 directions requests / mo 50,000 Matrix elements / mo
Then \$0.50 per	Web apps 1,000 web map views 1,000 geocode requests 1,000 directions requests 1,000 Matrix elements / mo	Mobile SDKs 500 monthly active users 1,000 geocode requests 1,000 directions requests 1,000 Matrix elements / mo

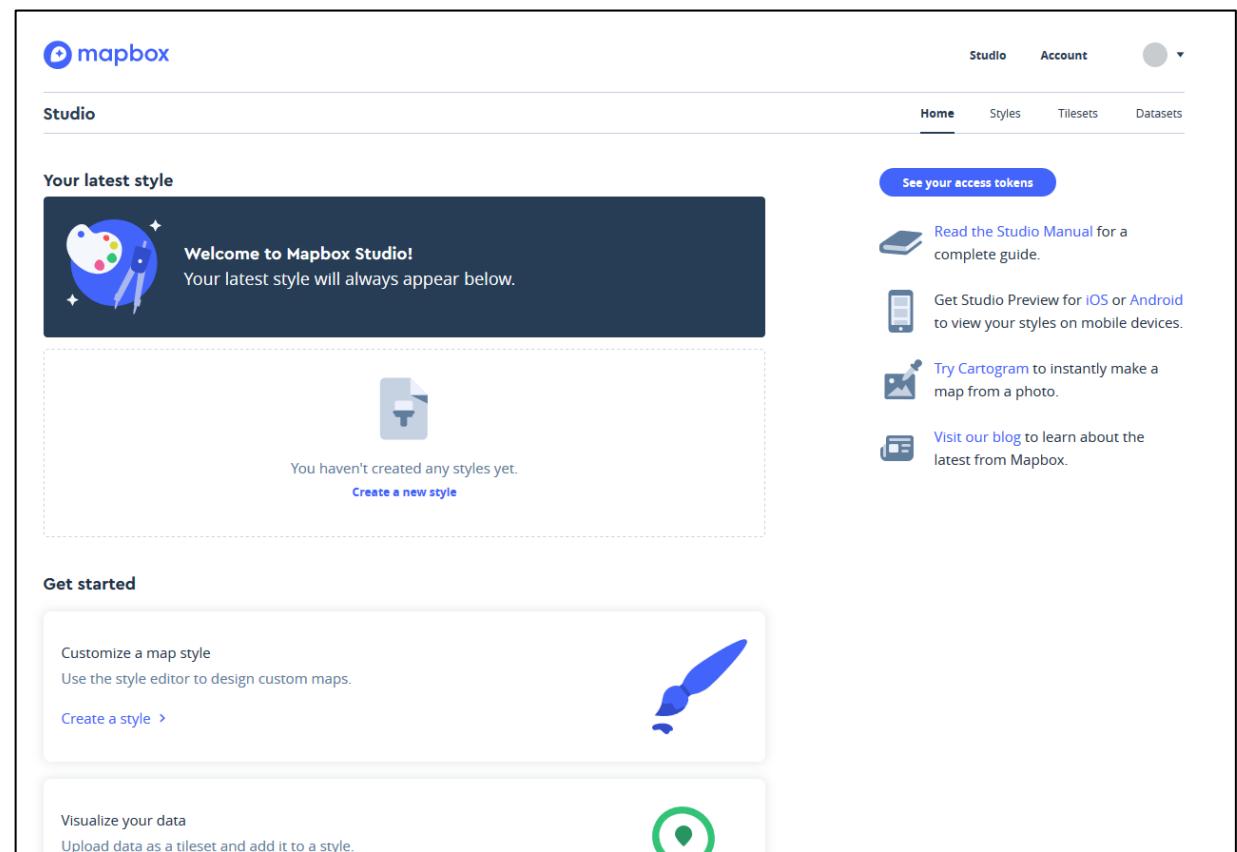
Pay-as-you-go plan includes

Satellite & street maps	Mapbox Studio	50 GB tileset storage
5 GB dataset storage	Unlimited Studio styles	Public/free web & mobile apps

[Current plan](#)

Mapbox Studio

- Environnement en ligne de gestion des :
 - Fonds de carte (*Styles*)
 - Jeux de données (*Tilesets*)
 - Des clefs d'accès à l'API (*Account*)



Mapbox Studio

- Créer et gérer des jeux de données (Tilessets)

The screenshot shows the Mapbox Studio interface, specifically the Tilessets section. At the top, there's a navigation bar with tabs for Home, Styles, **Tilessets** (which is highlighted with a red box), and Datasets. Below the navigation, the title "Tilessets" is displayed. A search bar and sorting options (Sort by: Name, Modified, Size) are present. A "New tileset" button is located in the top right of the main content area. The main content displays a list of "5 Default tilesets":

- Terrain (RGB-encoded dem)**: Default tileset
- Mapbox Satellite**: Default tileset
- Mapbox Terrain V2**: Default tileset
- Mapbox Traffic V1**: Default tileset
- Mapbox Streets v7**: Default tileset

Each item in the list has a "Menu" icon. To the right of the list, there are three sections with descriptive text:

- What is a tileset?**: A tileset is a collection of raster or vector data broken up into a uniform grid of square tiles at 22 preset zoom levels. [Read more.](#)
- How to create tilesets**: Click **New tileset** to upload your data. Mapbox renders [vector tiles](#) from your data so you can create styles from it. You can also use Mapbox default tilesets. Read the [Vector tiles docs](#) to find out more.
- How to use tilesets**: Once your vector tiles are ready, you can add them to a new or existing style. First open your style in the style editor. Next, you can either create a new layer with this tileset as the source, or you can change an existing layer's data source to this tileset.

Importer des données dans Mapbox Studio

- Mapbox studio permet de stocker 50GO de données vectorielles et matricielles

The image shows two screenshots of the Mapbox Studio interface. The left screenshot displays the 'Tilesets' page with four default tilesets listed: 'Mapbox Satellite', 'Mapbox Traffic V1', 'Mapbox Terrain V2', and 'Mapbox Streets v7'. A blue button labeled 'New tileset' is highlighted with a red box. The right screenshot shows a modal dialog titled 'New tileset' with instructions: 'Drag and drop a MBTiles, KML, GPX, GeoJSON, Shapefile (zipped), or CSV file here to convert it into vector tiles. To create raster tiles, drag and drop a GeoTIFF file.' It includes a 'Select a file' button and a 'Create from dataset' link.

Tilesets

Search Sort by Name Modified Size

New tileset

4 Default tilesets

- Mapbox Satellite Default tileset
- Mapbox Traffic V1 Default tileset
- Mapbox Terrain V2 Default tileset
- Mapbox Streets v7 Default tileset

Menu ⋮

Menu ⋮

Menu ⋮

Menu ⋮

Menu ⋮

New tileset

Upload file Create from dataset

Drag and drop a MBTiles, KML, GPX, GeoJSON, Shapefile (zipped), or CSV file here to convert it into vector tiles. To create raster tiles, drag and drop a GeoTIFF file.

Select a file

Mapbox Studio

- Créer et gérer des fonds de cartes (Styles)

The screenshot shows the Mapbox Studio interface. At the top, there's a navigation bar with 'Studio' and 'Account' tabs. Below the navigation is a secondary navigation bar with 'Home', 'Styles' (which is highlighted with a red box), 'Tilesets', and 'Datasets'. The main content area is titled 'Styles' and features a 'Create a new style' section with a 'Basic Template' button and a 'More ways to create' button. A 'Your styles' section includes a search bar and a sorting dropdown set to 'Modified'. A central message states 'You haven't created any styles yet.' with a 'Create a style' button. To the right, there's a 'How to create styles' section with instructions and a link to 'View Classic styles or projects >'. At the bottom, there are links for 'Changelog', 'Developer documentation', 'Studio manual', 'Contact', 'Terms + Privacy', and '© Mapbox'.

mapbox

Studio

Home Styles Tilesets Datasets

Styles

Create a new style

Basic Template The best way to get started. Create

More ways to create Pick a template or upload a style.

Your styles

Search Sort by Name Modified

You haven't created any styles yet.

Create a style

How to create styles

Customize a template style, upload your own stylesheet, or design from scratch by choosing "More ways to create".

View Classic styles or projects >

Changelog Developer documentation Studio manual Contact

Terms + Privacy © Mapbox

Mapbox Studio

- Créer et gérer ses clefs d'accès à l'API (Access tokens)

The screenshot shows the Mapbox Studio account interface. At the top, there's a navigation bar with 'Studio' (disabled), 'Account' (selected and highlighted with a red box), and other options like 'Dashboard', 'Access tokens', 'Statistics', 'Invoices', and 'Settings'. Below the navigation is a welcome message 'Welcome, dsfdsf!'. A 'Let's get started' section contains two cards: 'Integrate Mapbox' (with an icon of a smartphone) and 'Mapbox Studio' (with an icon of a map). The 'Access tokens' section is the main focus, featuring a button '+ Create a token'. It explains that tokens are needed for various Mapbox services. A 'Default public token' is listed with its ID: 'pk.eyJ1IjoizHNmZHNmIiw1YSI6ImNqaDNlMGpqMzBlaXkycXFsm3hsazB4bGg1fQ.gCQk04_Zp7L5EzlyeaZ6WQ'. Below the token are buttons for 'STYLES:READ', 'STYLES:WRITE', 'FONTS:READ', and 'DATASETS:READ'. A 'Refresh token' button is also present. To the right of the tokens section are sections for 'Plan' (Pay-as-you-go, Change plan), 'Current billing cycle usage' (No usage information to display, Learn more about pricing), and 'Tools & resources' (Integrate Mapbox, Design in Mapbox Studio, Documentation, Help).

Template de départ

<https://gitlab.huma-num.fr/bmericskay/mapboxgl>

```
<!DOCTYPE html>
<html>
<head>
    <meta charset='utf-8' />
    <title>MapboxGL</title>

    <link href="https://api.mapbox.com/mapbox-gl-js/v3.1.2/mapbox-gl.css" rel="stylesheet">
    <script src="https://api.mapbox.com/mapbox-gl-js/v3.1.2/mapbox-gl.js"></script>

    <style>
        #map {position:absolute; top:0; bottom:0; width:100%;}
    </style>

</head>

<body>
<div id='map'></div>

<script>
    // AccesToken
    mapboxgl.accessToken = 'pk.eyJ1IjoibWF1cmLjaW9iYXF1ZXJvIiwiYSI6ImNsCHNrZDd6aDA0NzcyaxJ6OTJiODVkMXYifQ.Sxnc-50cCwIwe1Om58AHQA';

    // Configuration de la carte
    var map = new mapboxgl.Map({
        container: 'map',
        style: 'mapbox://styles/mapbox/light-v9', // Fond de carte
        center: [-1.68, 48.12], // lat/long
        zoom: 15, // zoom
        pitch: 50, // Inclinaison
        bearing: -10 // Rotation
    });

    </script>

</body>
</html>
```

Template de départ

```
<!DOCTYPE html>
<html>
<head>
    <meta charset='utf-8' />
    <title>MapboxGL</title>

    <script src="https://api.mapbox.com/mapbox-gl-js/v2.0.1/mapbox-gl.js"></script>
    <link href="https://api.mapbox.com/mapbox-gl-js/v2.0.1/mapbox-gl.css" rel="stylesheet" />

    <style>
        #map {position:absolute; top:0; bottom:0; width:100%;}
    </style>
</head>

<body>
    <div id='map'></div>

    <script>
        // AccesToken
        mapboxgl.accessToken =
            'pk.eyJ1IjoibmluYW5vdW4iLCJhIjoiY2pjdhBoZGlzMnV4dDJxcGc5azJkbWRiYSJ9.o4dZRrdHcgVEKCveOXG1YQ';

        // Configuration de la carte
        var map = new mapboxgl.Map({
            container: 'map',
            style: 'mapbox://styles/mapbox/light-v9', // Fond de carte
            center: [-1.68, 48.12], // lat/long
            zoom: 15, // zoom
            pitch: 50, // Inclinaison
            bearing: -10 // Rotation
        });
    </script>
</body>
</html>
```

Appel API MapboxGL

Style de la carte

Clef d'accès à l'API

Fond de carte
Niveau de zoom
Centrage de la carte (X,Y)
Inclinaison de la carte
Rotation de la carte

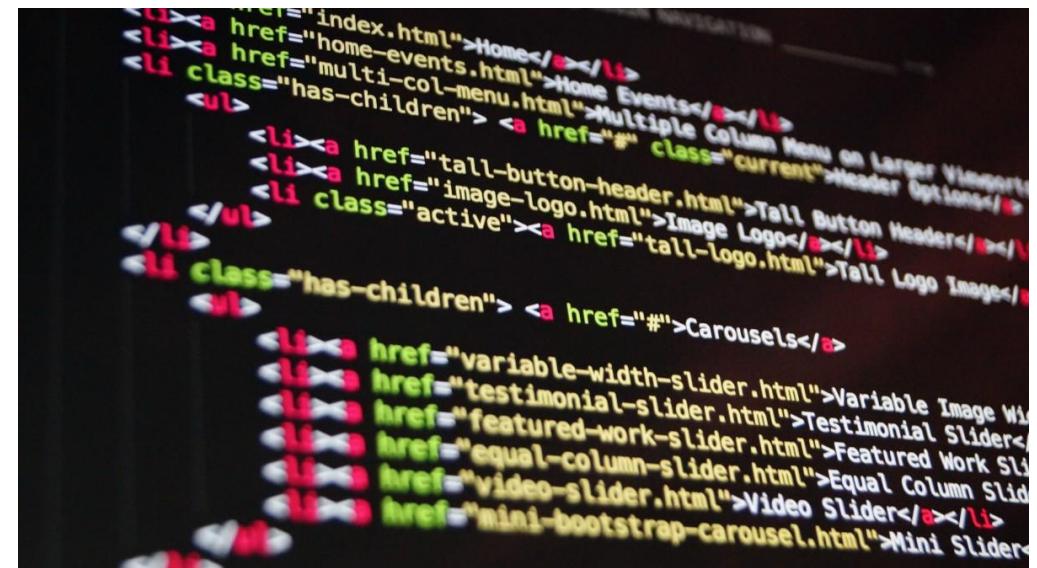
Coder en ligne ou en local

- Utiliser un éditeur de code installé (notepad, ATOM,...)

OU

- Utiliser un éditeur de code en ligne

<https://codepen.io ...>



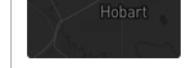
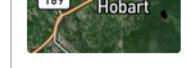
```
<li><a href="#">Home</a></li>
<li><a href="home-events.html">Home Events</a></li>
<li><a href="multi-col-menu.html">Multiple Column Menu on Larger Viewports</a>
<li class="has-children"> <a href="#" class="current">Header Options</a>
  <ul>
    <li><a href="tall-button-header.html">Tall Button Headers</a></li>
    <li><a href="image-logo.html">Image Logo</a></li>
    <li class="active"><a href="tall-logo.html">Tall Logo Images</a>
  </ul>
</li>
<li class="has-children"> <a href="#">Carousels</a>
  <ul>
    <li><a href="variable-width-slider.html">Variable Image Width Sliders</a>
    <li><a href="testimonial-slider.html">Testimonial Slider</a>
    <li><a href="featured-work-slider.html">Featured Work Slider</a>
    <li><a href="equal-column-slider.html">Equal Column Slider</a>
    <li><a href="video-slider.html">Video Slider</a></li>
    <li><a href="mini-bootstrap-carousel.html">Mini Sliders</a>
  </ul>
</li>
```

Les fonds de carte

Changer de fond de carte

- Les fonds de cartes de Mapbox > tuiles vectorielles

```
style: 'mapbox://styles/mapbox/dark-v11',
```

Style name	Style URL	Style image
Mapbox Streets	mapbox://styles/mapbox/streets-v12	
Mapbox Outdoors	mapbox://styles/mapbox/outdoors-v12	
Mapbox Light	mapbox://styles/mapbox/light-v11	
Mapbox Dark	mapbox://styles/mapbox/dark-v11	
Mapbox Satellite	mapbox://styles/mapbox/satellite-v9	
Mapbox Satellite Streets	mapbox://styles/mapbox/satellite-streets-v12	
Mapbox Navigation Day	mapbox://styles/mapbox/navigation-day-v1	
Mapbox Navigation Night	mapbox://styles/mapbox/navigation-night-v1	

Changer de fond de carte

- Aller plus loin avec la documentation

<https://docs.mapbox.com/api/maps/styles/>

<https://docs.mapbox.com/mapbox-gl-js/guides/styles/>

<https://docs.mapbox.com/api/maps/styles/>

Mapbox GL JS	
<input type="text"/> Search	Parameters
API REFERENCE	options (Object)
Map	Name
Properties and options	options.container
Markers and controls	(HTMLElement string)
Geography and geometry	Description
User interaction handlers	The HTML element in which Mapbox GL JS will render the map, or the element's string <code>id</code> . The specified element must have no children.
Sources	options.minZoom
Events	number
	default: 0
EXAMPLES	options.maxZoom
PLUGINS	number
STYLE SPECIFICATION	default: 22
TUTORIALS 	options.minPitch
TROUBLESHOOTING 	number
	default: 0
	options.maxPitch
	number
	default: 85
	options.style
	(Object string)?
	The map's Mapbox style. This must be an a JSON object conforming to the schema described in the Mapbox Style Specification , or a URL to such JSON.
	To load a style from the Mapbox API, you can use a URL of the form <code>mapbox://styles/:owner/:style</code> , where <code>:owner</code> is your Mapbox account name and <code>:style</code> is the style ID. Or you can use one of the following the predefined Mapbox styles :
	<ul style="list-style-type: none">• <code>mapbox://styles/mapbox/streets-v11</code>• <code>mapbox://styles/mapbox/outdoors-v11</code>• <code>mapbox://styles/mapbox/light-v10</code>• <code>mapbox://styles/mapbox/dark-v10</code>• <code>mapbox://styles/mapbox/satellite-v9</code>• <code>mapbox://styles/mapbox/satellite-streets-v11</code>• <code>mapbox://styles/mapbox/navigation-preview-day-v4</code>• <code>mapbox://styles/mapbox/navigation-preview-night-v4</code>

Changer de fond de carte

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <meta charset='utf-8' />
5      <title>MapboxGL</title>
6
7      <script src='https://api.tiles.mapbox.com/mapbox-gl-js/v0.44.0/mapbox-gl.js'></script>
8      <link href='https://api.tiles.mapbox.com/mapbox-gl-js/v0.44.0/mapbox-gl.css' rel='stylesheet' />
9
10     <style>
11         #map { position: absolute; top: 0; bottom: 0; width: 100%; }
12     </style>
13
14     </head>
15
16     <body>
17         <div id='map'></div>
18
19     <script>
20         // AccesToken
21         mapboxgl.accessToken = 'pk.eyJ1IjoibmluYW5vdW4iLCJhIjoiY2pjdhBoZG1zMnV4dDxcGc5azJkbWRiYSJ9.o4dZRrdHcgWEKCveC
22
23         // Configuration de la carte
24         var map = new mapboxgl.Map({
25             container: 'map',
26             style: 'mapbox://styles/mapbox/satellite-v9',
27             center: [-1.68, 48.12], // lat/long
28             zoom: 15, // zoom
29             pitch: 50, // Inclinaison
30             bearing: -10 // Rotation
31         });
32
33     </script>
34
35     </body>
36
37     </html>
```



Mettre un fond de carte personnel

- Mettre votre clef d'accès personnelle à la place de celle fournie

The screenshot shows the Mapbox Studio interface. On the left is a sidebar with links: Home, Styles, Tilesets, Datasets, Stats, and Classic. The main area has three sections: 'Styles' (with 'Go to styles' and 'New style' buttons), 'Tilesets' (with 'Go to tilesets' and 'New tileset' buttons), and 'Datasets' (with 'Go to datasets' and 'New dataset' buttons). At the bottom of the sidebar, there's a 'No updates' message and a 'Signout' button. In the center, under the 'Account' heading, it says 'Hello mastersiget, you are on Starter plan 50.0k map views'. Below that is the 'Access token' section, which contains a placeholder token: 'pk.eyJ1j0ibWZ0b3yc2Im9zLChjjoizv2lon.' This placeholder is highlighted with a red rectangular box. To the right of the token is a trash can icon. Below the token, it says 'If you need your access token to use any of the Mapbox APIs and libraries, click here to add or revoke tokens.' At the bottom of the central area, there are links for 'Help', 'On the blog', and 'Sign in'.

Incorporer un fond de carte personnel

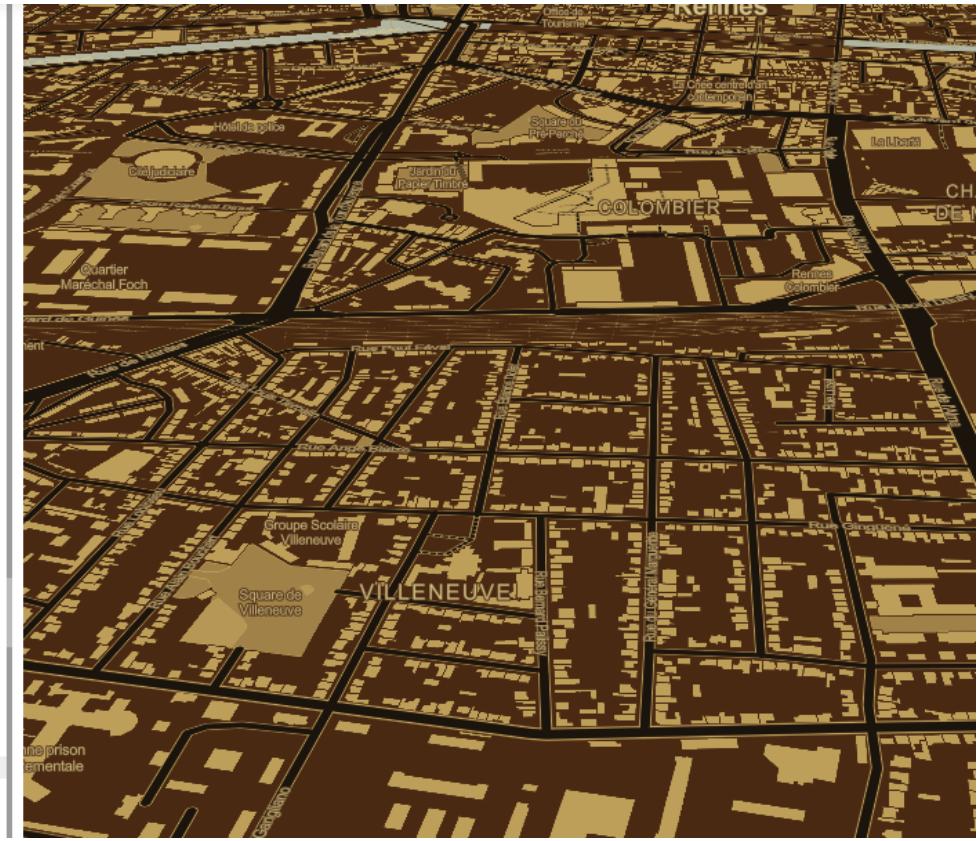
- Récupérer l'URL de votre fond de carte personnel

The screenshot shows the Mapbox Studio interface with the 'Styles' tab selected. A card for 'My Cartogram Style' is highlighted, showing it was published on June 27, 2017, and is private. A context menu is open over this card, with the 'Share, develop & use' option highlighted by a red box.

The screenshot shows the preview page for 'My Cartogram Style'. The title 'My Cartogram Style' is displayed, along with the publication date (June 27, 2017) and privacy status (Private). The preview map shows a cartogram of New York City where areas represent population density. A context menu is open over the map, with the 'Style URL' option highlighted by a red box. The URL 'mapbox://styles/mastersigat/cj4Foc5e...' is visible in the menu.

Incorporer un fond de carte personnel

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset='utf-8' />
5   <title>MapboxGL</title>
6
7 <script src='https://api.tiles.mapbox.com/mapbox-gl-js/v0.44.0/mapbox-gl.js'></script>
8 <link href='https://api.tiles.mapbox.com/mapbox-gl-js/v0.44.0/mapbox-gl.css' rel='stylesheet' />
9
10 <style>
11 #map { position:absolute; top:0; bottom:0; width:100%; }
12 </style>
13
14 </head>
15
16 <body>
17 <div id='map'></div>
18
19 <script>
20   // AccesToken
21 mapboxgl.accessToken = 'pk.eyJ1IjoibWFzdGVyc2lnYXQiLCJhIjoiY2loNG9mamxwMHp2dHgxbTBjY2h1b2RteiJ9.dDYKXX9907pbT6sTAJ4FvA';
22
23   // Configuration de la carte
24 var map = new mapboxgl.Map({
25   container: 'map',
26   style: 'mapbox://styles/mastersigat/cj4foc5eo3hsr2sqlomdgeuvd',
27   center: [-1.68, 48.12], // lat/long
28   zoom: 15, // zoom
29   pitch: 50, // Inclinaison
30   bearing: -10 // Rotation
31 });
32
33 </script>
34
35 </body>
36 </html>
37
```



Première carte MapboxGL.js

The screenshot shows the CodePen interface with three tabs: HTML, CSS, and JS.

HTML:

```
<html>
  <head>
    <title>First Mapbox Map</title>
    <meta name='viewport' content='initial-scale=1,maximum-scale=1,user-scalable=no' />
    <script src="https://api.mapbox.com/mapbox-gl-js/v2.0.1/mapbox-gl.js"></script>
    <link href="https://api.mapbox.com/mapbox-gl-js/v2.0.1/mapbox-gl.css" rel="stylesheet" />
  </head>
  <body>
    <div id='map'></div>
  </body>
</html>
```

CSS:

```
#map { position: absolute; top: 0; bottom: 0; width: 100%; }
```

JS:

```
// AccesToken
mapboxgl.accessToken =
  'pk.eyJ1IjoibmluYW5vdW41LCJhIjoiY2pjdBBoZG1zMnV4dDJxcGc5azJkbWRiYSJ9
  .o4dZRrdHcgVEKCve0XG1YQ';

// Configuration de la carte
var map = new mapboxgl.Map({
  container: 'map',
  style: 'mapbox://styles/ninanoun/ciyo7h3mc001q2rpngdsa9zwh', // fond de carte
  center: [-1.68, 48.12], // lat/long
  zoom: 15, // zoom
  pitch: 50, // Inclinaison
```

The main area displays a Mapbox GL.js map of Rennes, France, centered around the city's central business district. The map includes street names like Rue Auguste Blanqui, Rue Carnot, and Rue Lagraverend, along with various landmarks such as the Lycée général et technologique Victor et Hélène Bach, the Square de Coëtlogon, and the Cimetière du Nord. The map uses a dark gray base layer with light gray roads and green for parks. A blue line highlights a specific route or path through the city center.

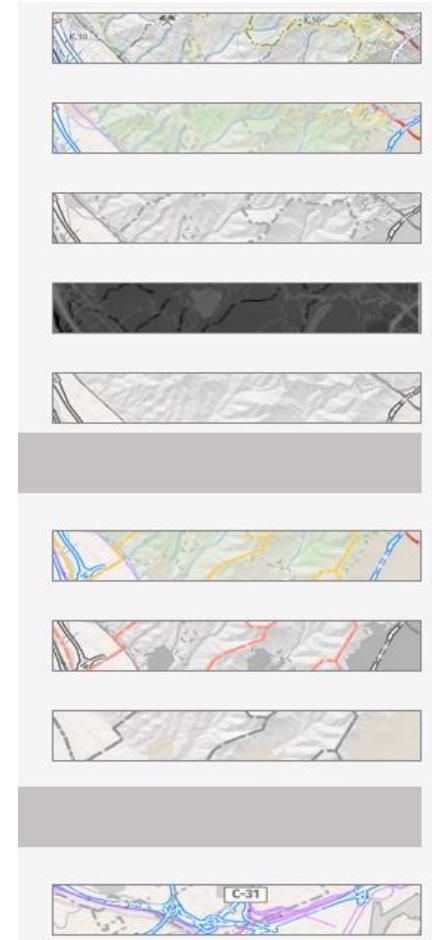
<https://codepen.io/BorisMericskay/pen/qByJqZB>

Où trouver des fonds de carte libre

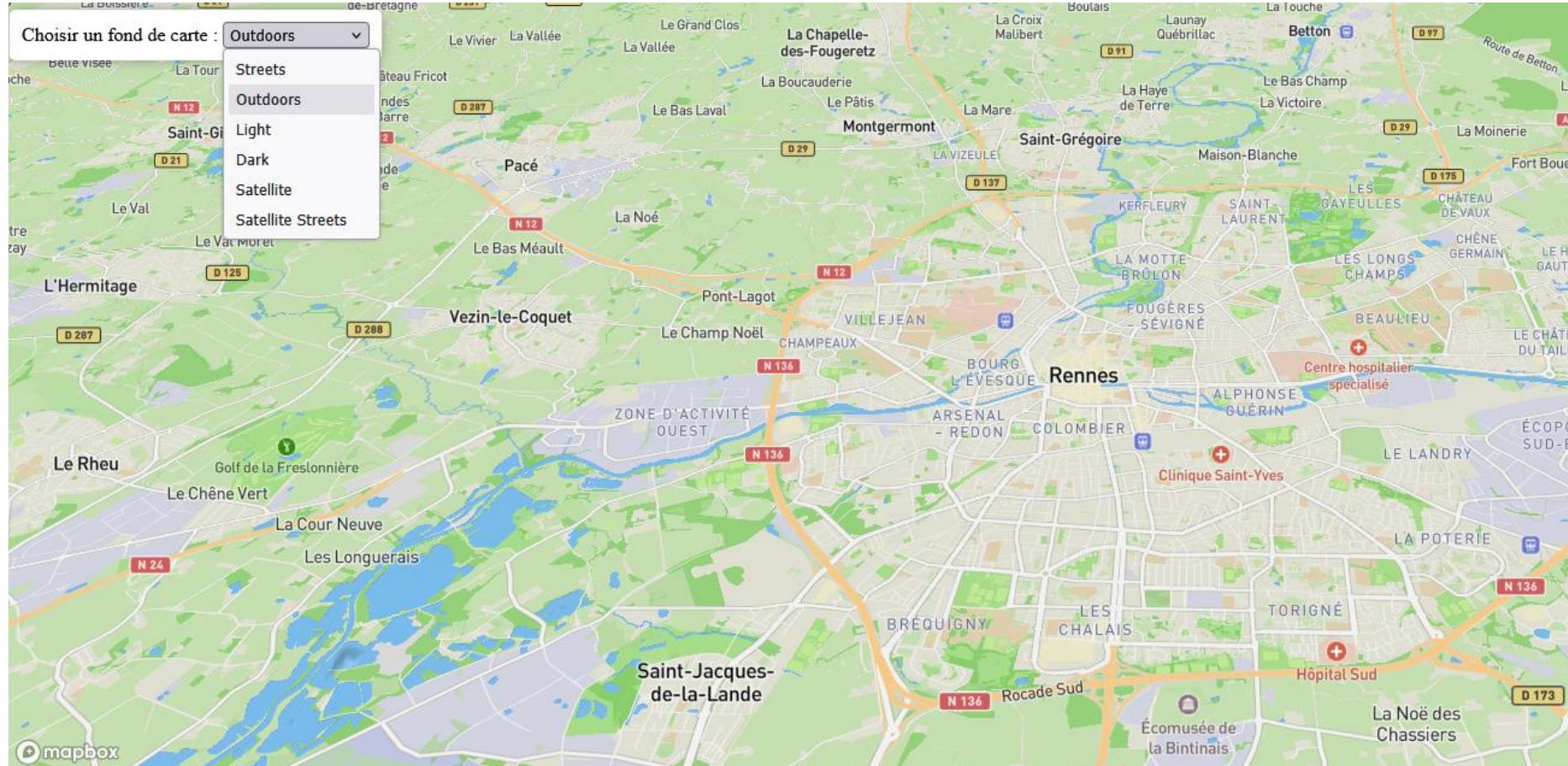
Il existe plusieurs flux de tuiles vectorielles en libre accès

Format .json / Basés sur les données OSM ou IGN

- [https://deck.gl/docs/api-reference/cartо/basemap](https://deck.gl/docs/api-reference/cartو/basemap)
- <https://openicgc.github.io/>
- <https://openmaptiles.geo.data.gouv.fr/>
- <https://github.com/Viglino/geoservice-style>



Ajouter des fonds de carte externes



<https://codepen.io/BorisMericskay/pen/JojXRq>

Ajout d'outils

A placer à la fin du script !

```
// Boutons de navigation  
  
var nav = new mapboxgl.NavigationControl();  
map.addControl(nav, 'top-left');
```

```
// Ajout Echelle cartographique  
map.addControl(new mapboxgl.ScaleControl({  
  maxWidth: 120,  
  unit: 'metric'}));
```

```
// Bouton de géolocalisation  
  
map.addControl(new mapboxgl.GeolocateControl  
({positionOptions: {enableHighAccuracy: true},  
trackUserLocation: true,  
showUserHeading: true}));
```

<https://docs.mapbox.com/mapbox-gl-js/api/markers/#geolocatecontrol>
<https://docs.mapbox.com/mapbox-gl-js/api/markers/#navigationcontrol>

<https://docs.mapbox.com/mapbox-gl-js/api/markers/#scalecontrol>

https://docs.mapbox.com/mapbox-gl-js/api/markers/?size=n_10_n#geolocatecontrol

Ajouter des données

OSM, données hébergées et données en local

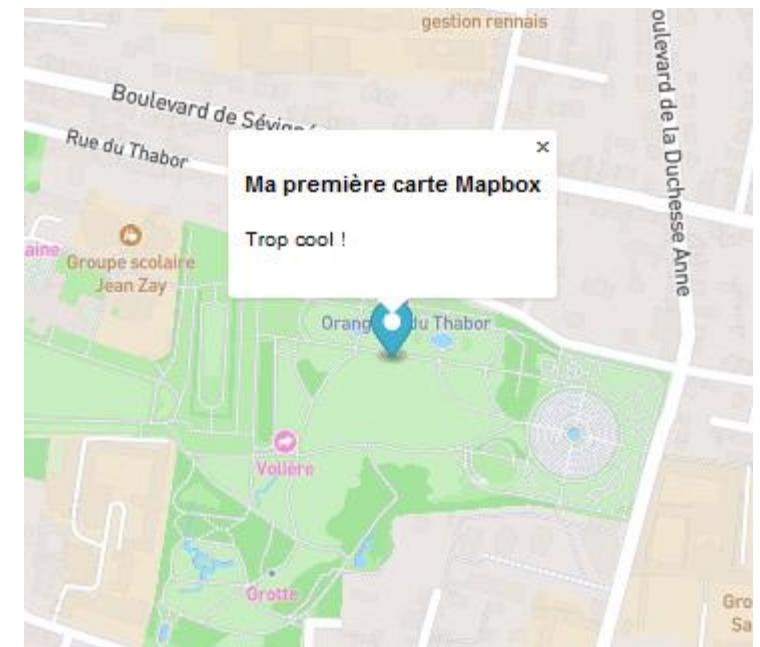
Ajouter un marqueur

- Ajouter marqueur + définir contenu et Popup

```
// Ajout Marqueur
const marker1 = new mapboxgl.Marker()
.setLngLat([-1.669099, 48.114799])
.addTo(map);

// Contenu de la popup du marqueur
var popup = new mapboxgl.Popup({ offset: 25 })
.setHTML("<h3>Ma première carte Mapbox</h3><p>Trop cool !</p>");

// Associer Contenu et Marqueur
marker1.setPopup(popup);
```

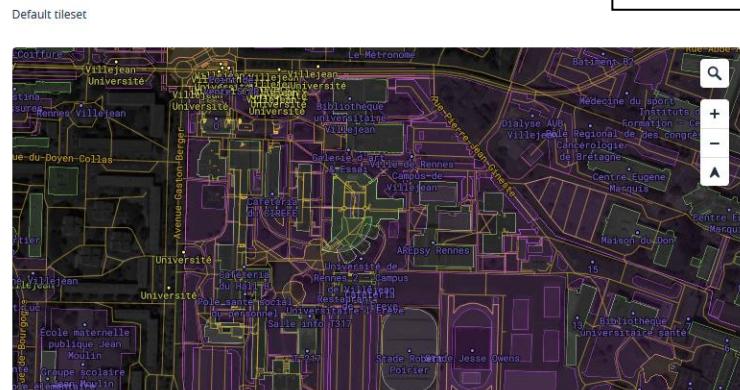


<https://codepen.io/BorisMericskay/pen/XWGyqqx>

Ajout de données OSM

- MapboxGL permet aussi de mobiliser des données OSM
 - Le jeu de données (*tileset*) Mapbox Streets v8 propose un ensemble de couches OSM (routes, bâtiments, labels, hydrologie,...)

Mapbox Streets v8



ID du tileset OSM

Default tileset

mapbox.mapbox-streets-v8

Tileset Details

Learn about the layers and fields in the Mapbox Streets v8 source

Type: vector

Zoom extent: z0 – z16. Data will be overzoomed to z22, but simplified past z16. Learn how to adjust zoom extent >

Bounds: -180.0,-85.0,180.0,85.0

Adding to a style

Create a new style or open an existing style in the style editor. Then, add a new layer and pick Mapbox Streets v8 as the source.

5 fields

Number, 0-2. The administrative level of the boundary

Disputed boundaries are 'true', all others are 'false'.

iso_3166_1

The IFC 3166-1 code is a standard of the International Organization for Standardization (ISO) for the representation of countries and territories.

Mapbox Streets V7		
admin	aeroway	airport_label
barrier_line	building	country_label
housenum_label	landuse	landuse_overview
marine_label	motorway_junction	mountain_peak
place_label	poi_label	rail_station_label
road	road_label	state_label
water	water_label	waterway
waterway_label		

Ajout de données OSM

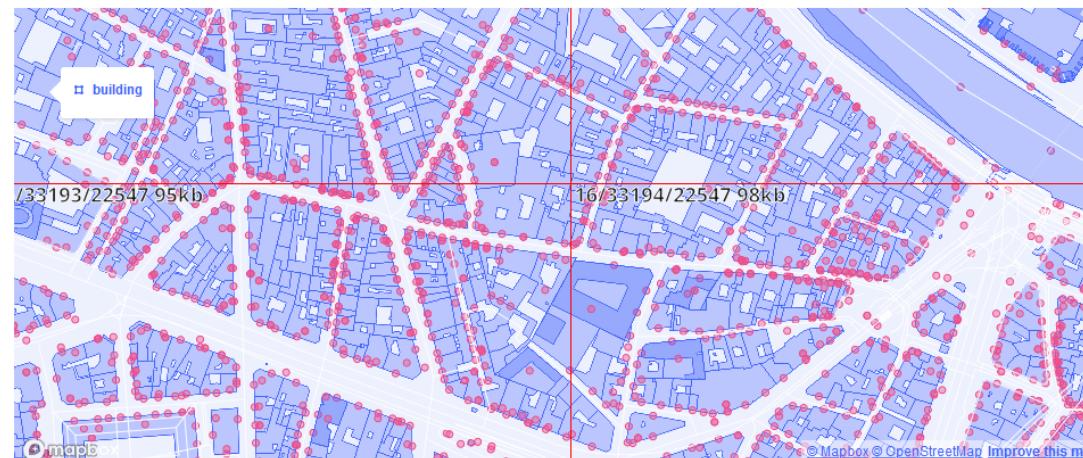
- Documentation

<https://docs.mapbox.com/vector-tiles/reference/mapbox-streets-v8/>

Mapbox Streets v8

Source id: mapbox.mapbox-streets-v8 

This is an in-depth guide to the data inside the Mapbox Streets vector tile source to help with styling. For full examples of using Mapbox Streets vector tiles to create a map style, see the default styles in [Mapbox Studio](#).



The map above uses minimal styling to illustrate the coverage, density, and zoom extents of various source layers in this tileset. Toggle between [Map](#) and [Style JSON](#) to see the relationship between the visual map and the source data.

[Map](#) [Style JSON](#)

Ajout de données OSM

- Ajouter cette commande à la fin du script

On appelle ici les routes issues d'OSM

```
map.on('load', function () {  
  
    map.addSource('mapbox-streets-v8', {  
        type: 'vector',  
        url: 'mapbox://mapbox.mapbox-streets-v8'});  
  
    map.addLayer({  
        "id": "Routes",  
        "type": "line",  
        "source": "mapbox-streets-v8",  
        "layout": {"visibility": "visible"},  
        "source-layer": "road",  
        "paint": {"line-color": "#FF7F50", "line-width": 1}  
    });  
  
});
```

map.on('load', callback) est un événement dans Mapbox GL JS qui se déclenche lorsque la carte est entièrement chargée et prête à être utilisée. Cela inclut le chargement de la carte de base, des tuiles de carte, des sources de données, des couches et des styles.

Ajout de données OSM

- Ajouter cette commande à la fin du script

On appelle ici les routes

```
map.on('load', function () {  
  
    map.addSource('mapbox-streets-v8', {  
        type: 'vector',  
        url: 'mapbox://mapbox.mapbox-streets-v8'});
```

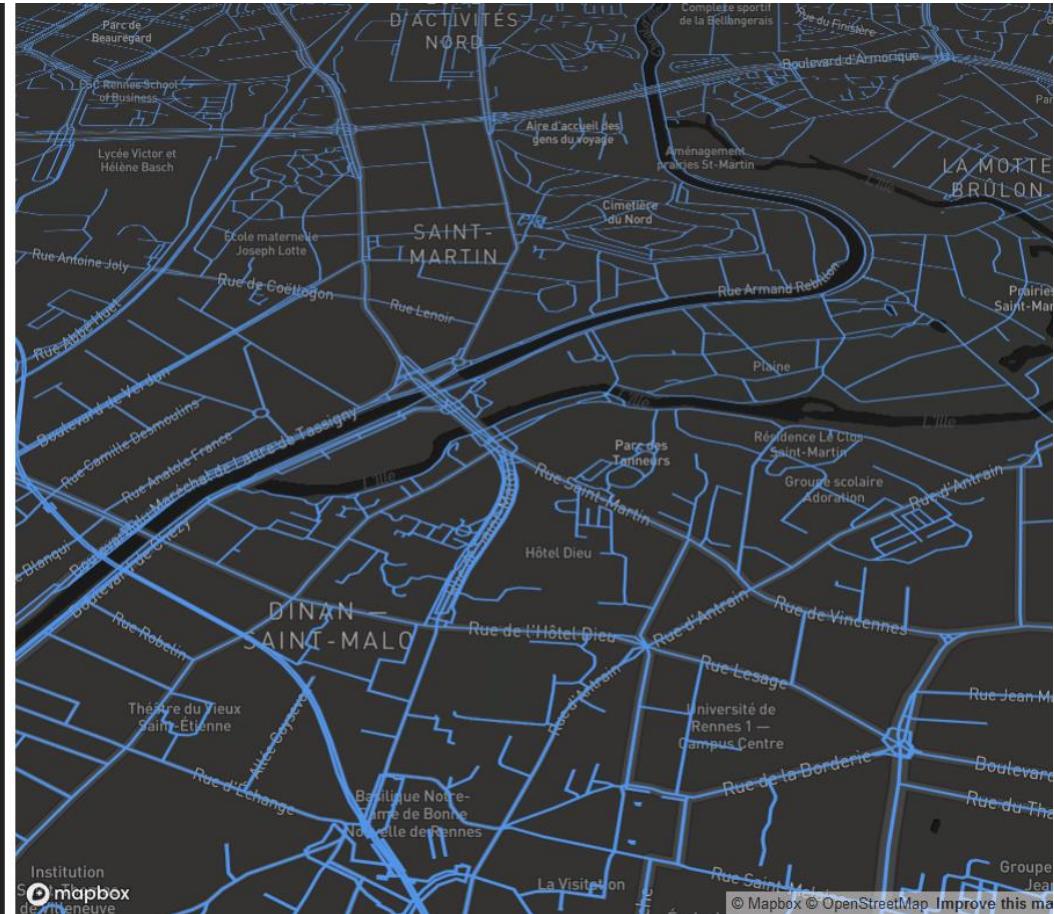
Définition de la source de données

```
map.addLayer({  
    "id": "Routes",  
    "type": "line",  
    "source": "mapbox-streets-v8",  
    "layout": {"visibility": "visible"},  
    "source-layer": "road",  
    "paint": {"line-color": "#FF7F50", "line-width": 1}  
});  
});
```

Couche de données

Ajout de données OSM

```
18 <script>
19   // AccesToken
20   accessToken =
21 'pk.eyJ1IjoibmluYW5vdW4iLCJhIjoiY2pjdhBoZG1zMnV4dDJxcGe5azJkbWRiYSJ9.o4dZRrdHcgVEKCve0XG1YQ';
22
23   // Configuration de la carte
24 var map = new mapboxgl.Map({
25   container: 'map',
26   style: 'mapbox://styles/mapbox/dark-v9', // Fond de carte
27   center: [-1.68, 48.12], // lat/long
28   zoom: 15, // zoom
29   pitch: 50, // Inclinaison
30   bearing: -10 // Rotation
31 });
32
33 map.on('load', function () {
34
35   map.addSource('mapbox-streets-v8', {
36     type: 'vector',
37     url: 'mapbox://mapbox.mapbox-streets-v8'});
38
39   map.addLayer({
40     "id": "Routes",
41     "type": "line",
42     "source": "mapbox-streets-v8",
43     "layout": {"visibility": 'visible'},
44     "source-layer": "road",
45     "paint": {"line-color": "#5496eb", "line-width": 2}
46   });
47
48 });
49
50 </script>
51
52 </body>
53 </html>
```



Ajout de données OSM

- Ajout du réseau hydrographique
 - Ajouter à la suite de l'appel de la couche des routes juste un appel de couche car la source est la même que pour les routes (*mapbox-streets-v7*)

```
// Hydrologie  
  
map.addLayer({  
    "id": "hydrologie",  
    "type": "line",  
    "source": "mapbox-streets-v8",  
    "source-layer": "waterway",  
    "paint": {"  
        "line-color": "#4dd2ff",  
        "line-width": 6  
    }  
});
```

Gérer fonds de carte ET données !

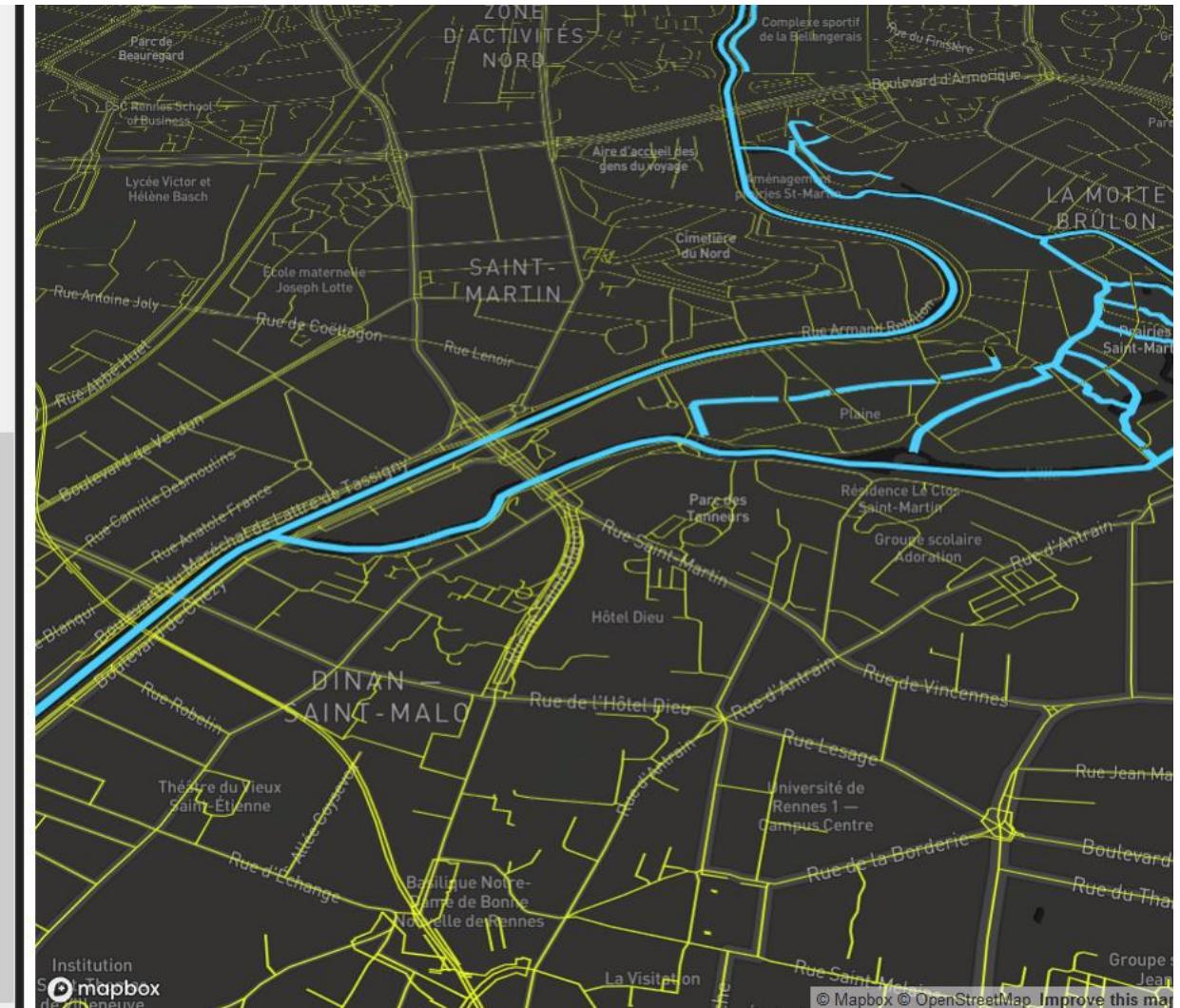
- Demander de régler le problème à l'IA



- Solution : <https://codepen.io/BorisMericskay/pen/xbxvLo>

Ajout de données OSM

```
27  center: [-1.68, 48.12], // lat/long
28  zoom: 15, // zoom
29  pitch: 50, // Inclinaison
30  bearing: -10 // Rotation
31 });
32
33 map.on('load', function () {
34
35 map.addSource('mapbox-streets-v8', {
36   type: 'vector',
37   url: 'mapbox://mapbox.mapbox-streets-v8'});
38
39 map.addLayer({
40   "id": "Routes",
41   "type": "line",
42   "source": "mapbox-streets-v8",
43   "layout": {"visibility": "visible"},
44   "source-layer": "road",
45   "paint": {"line-color": "#ebff1e", "line-width": 1}
46 });
47
48 // Hydrologie
49
50 map.addLayer({ "id": "hydrologie",
51   "type": "line",
52   "source": "mapbox-streets-v8",
53   "source-layer": "waterway",
54   "paint": {"line-color": "#4dd2ff",
55   "line-width": 7}
56 });
57
58 });
59
60 </script>
61
62 </body>
63 </html>
```



Filtrer des données OSM

- Il est possible de filtrer les données pour l'affichage
 - On peut par exemple filtrer les routes selon leur classe



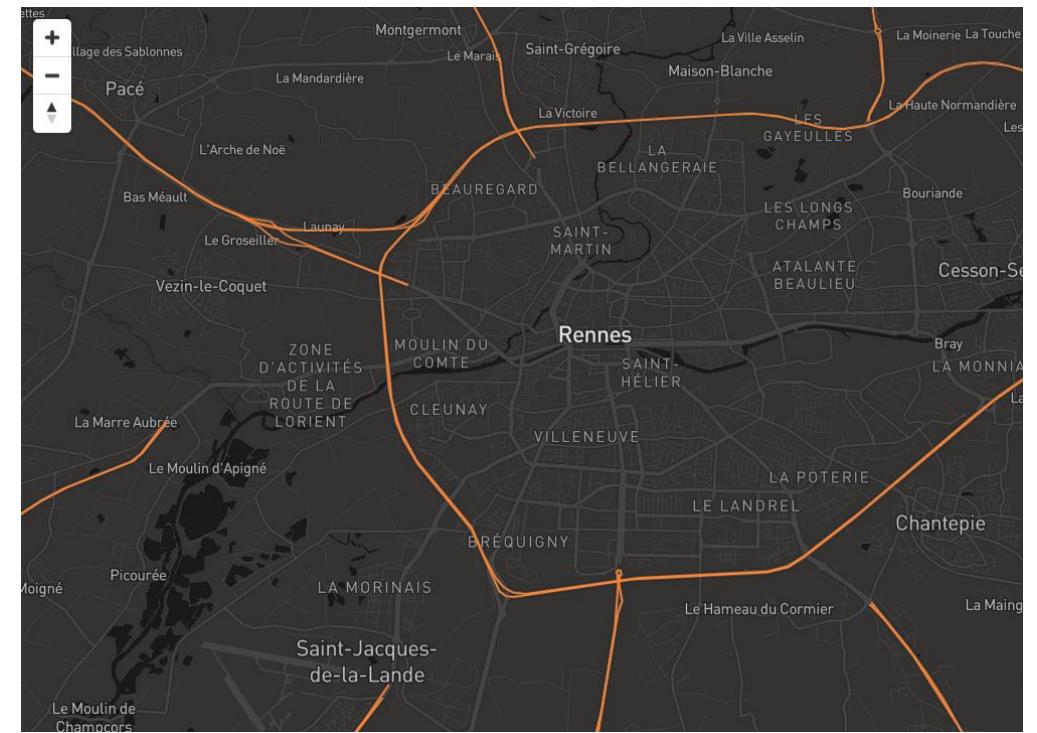
The screenshot shows the Mapbox Streets v7 dataset page. On the left, there's a world map with various place names labeled. On the right, the 'road' layer is selected, showing its properties:

- road**: This layer contains mostly LineStrings.
- class**: One of: 'motorway', 'motorway_link', 'trunk', 'primary', 'secondary', 'tertiary', 'link', 'street', 'street_limited', 'pedestrian', 'construction', 'track', 'service', 'ferry', 'path', 'golf'
- layer**: Number. Specifies z-ordering in the case of overlapping road segments. Common range is -5 to 5. Available from zoom level 13+.
- oneway**: Text. Whether traffic on the road is one-way. One of: 'true', 'false'
- structure**: Text. One of: 'none', 'bridge', 'tunnel', 'ford'. Available from zoom level 13+.
- type**: In most cases, values will be that of the primary key from OpenStreetMap tags.

Filtrer des données OSM

- Je ne veux afficher que les routes à double sens

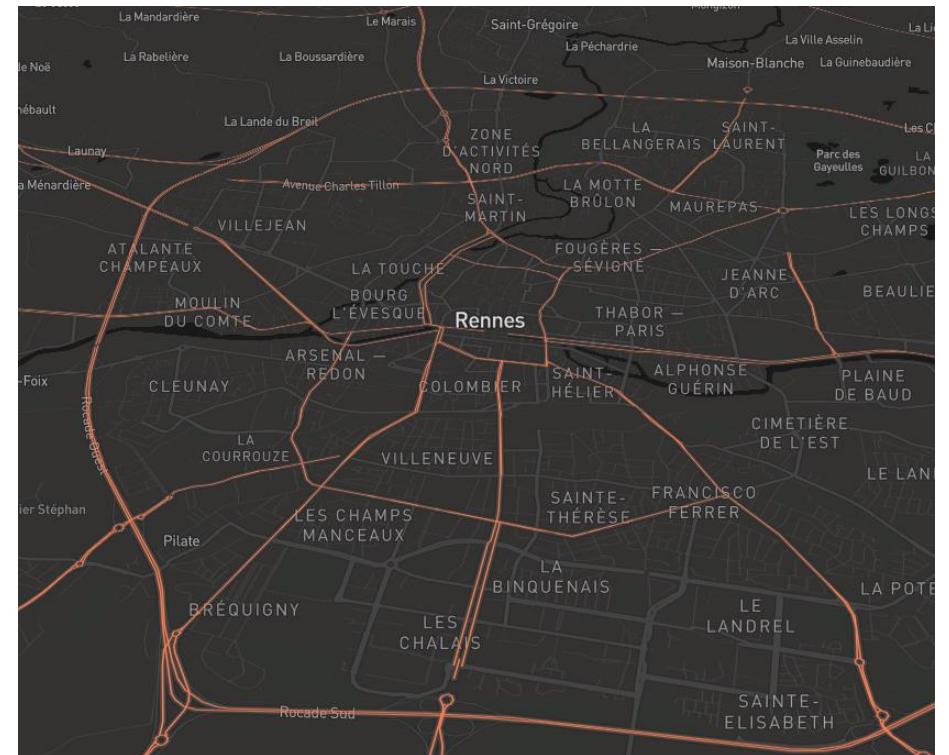
```
map.addLayer({  
  "id": "routes",  
  "type": "line",  
  "source": "mapbox-streets-v7",  
  "source-layer": "road",  
  "filter": [']==', 'class', 'trunk'],  
  "layout": {"visibility": 'visible'},  
  "paint": {"line-color": "#ff8533", "line-width": 1.3}  
});
```



Filtrer des données OSM

- Je ne veux afficher que les routes principales (double sens, principales,...)

```
map.addLayer({  
  "id": "routes",  
  "type": "line",  
  "source": "mapbox-streets-v7",  
  "source-layer": "road",  
  "filter": ["all", ["in", "class", "motorway", "trunk", "primary"]],  
  "layout": {"visibility": 'visible'},  
  "paint": {"line-color": "#ff8533", "line-width": 1.3}  
});
```



Ajout de données OSM

- Ajout les **bâtiments** (ajouter juste un appel de couche car la source est la même que pour les routes)

```
// Batiments

map.addLayer({
    "id": "batiments",
    "type": "fill",
    "source": "mapbox-streets-v8",
    "source-layer": "building",
    "paint": {"fill-color": "#FEFEFE",
              "fill-opacity": 0.8}
});
```

Exemple

#MapboxGL / Afficher et filtrer des données d'OSM



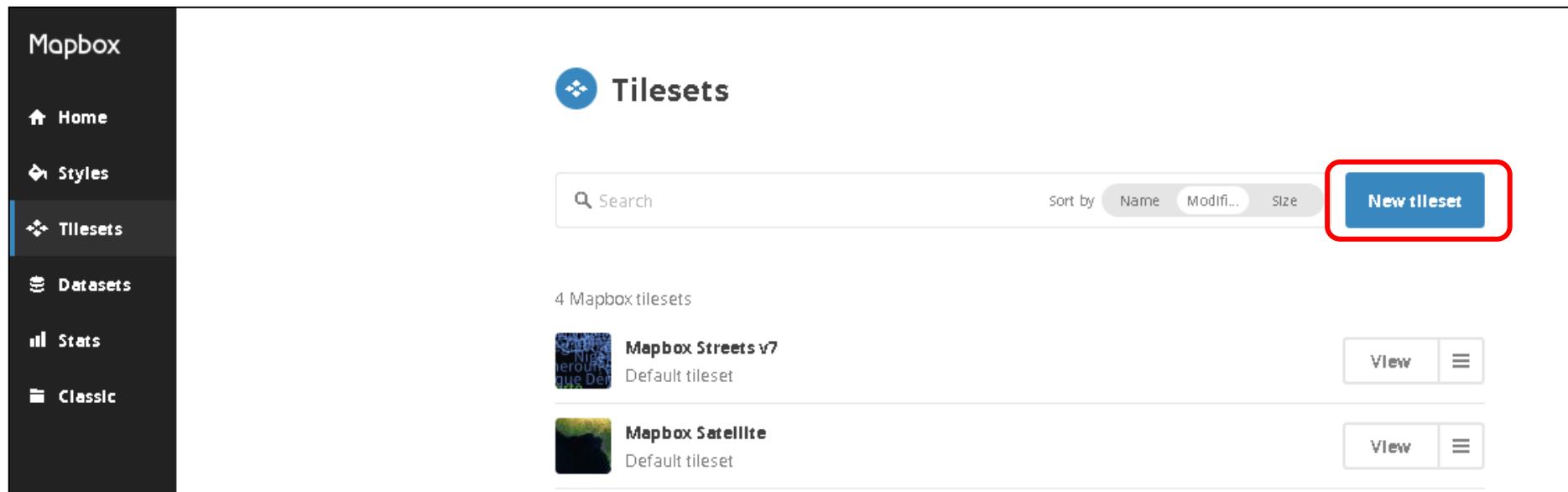
Built with [blockbuilder.org](#)

[Open](#)

<https://codepen.io/BorisMericskay/pen/abjRBav>

Ajout de données personnelles

- Première étape charger des données comme des Tilesets dans le Studio de Mapbox (csv, geojson, gpx, kml, shapefile zippé)
 - Intégrer le jeu de données des arrêts de bus et celui de la base équipements



Ajout de données personnelles

- Aller chercher les infos dans le studio de Mapbox (Tilessets)

The screenshot shows the Mapbox Tilesets interface. On the left is a dark sidebar with navigation links: Home, Styles, Tilesets (which is selected and highlighted in blue), Datasets, Stats, and Classic. Below these are notifications: "No updates" and "Account". The main area is titled "Tilesets" with a subtitle "3 Mapbox tilesets". It lists three default tilesets: "Mapbox Satellite" (Default tileset), "Mapbox Streets v7" (Default tileset), and "Mapbox Terrain V2" (Default tileset). Below this, it says "8 tilesets" and lists three user-created tilesets: "IRIS-6apbfw" (345 KB, Modified 20 days ago), "limites_proprietes-auaqb7" (103 MB, Modified 22 days ago), and "parcours-des-lignes-de-bus-du-1rjnez" (2 MB, Modified a month ago). A red box highlights the "IRIS-6apbfw" tileset. At the bottom right, there are links for "110 MB of 5.0 GB used", "Refresh", and "Upgrade Plan".

Name	Size	Last Modified	Action
Mapbox Satellite			View ☰
Mapbox Streets v7			View ☰
Mapbox Terrain V2			View ☰
IRIS-6apbfw	345 KB	Modified 20 days ago	View ☰
limites_proprietes-auaqb7	103 MB	Modified 22 days ago	View ☰
parcours-des-lignes-de-bus-du-1rjnez	2 MB	Modified a month ago	View ☰

Ajout de données personnelles

```
//Mes datas

map.addSource('nomdelasource', {
    type: 'vector',
    url: 'mapbox://' iddutileset'});

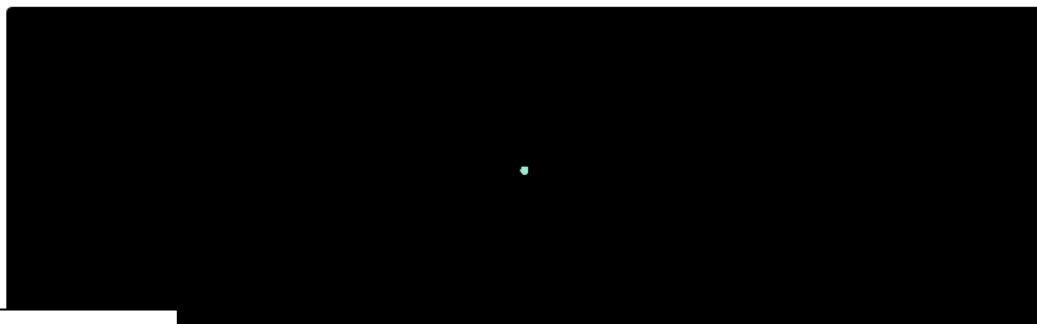
map.addLayer({
    'id': 'nomgenerique',
    'type': 'circle',
    'source': 'nomdelasource ',
    'source-layer': 'nomdelacouche',
    'layout': {'visibility': 'visible'},
    'paint': {'circle-radius': {'base': 1.5,'stops': [[13, 2], [22, 60]]}, 'circle-color': '#f31616',}, minzoom:10
});
```

Ajout de données personnelles

Bus-5ypx1k

Modified a few seconds ago

Preview



Nom de la couche

Layer details

Bus-5ypx1k

code	Type
String	
codeinseecommune	Type
String	
coordonnees	Type
String	
estaccessiblepmr	Type
String	
id	Type
String	
mobilier	Type
String	

8 properties

Add tileset to style
Replace
Make private
Delete

ID de votre Tileset

Map ID: ninanoun.58widelk

Details

Format: pbf Type: vector Size: 514 KB

Zoom extent: z0 ~ z14
Data will be visible above zoom 14, but may appear simplified. [Learn how to adjust zoom extent](#)

Bounds: -1.9,47.9,-1.5,48.3

Ajout des arrêts de bus

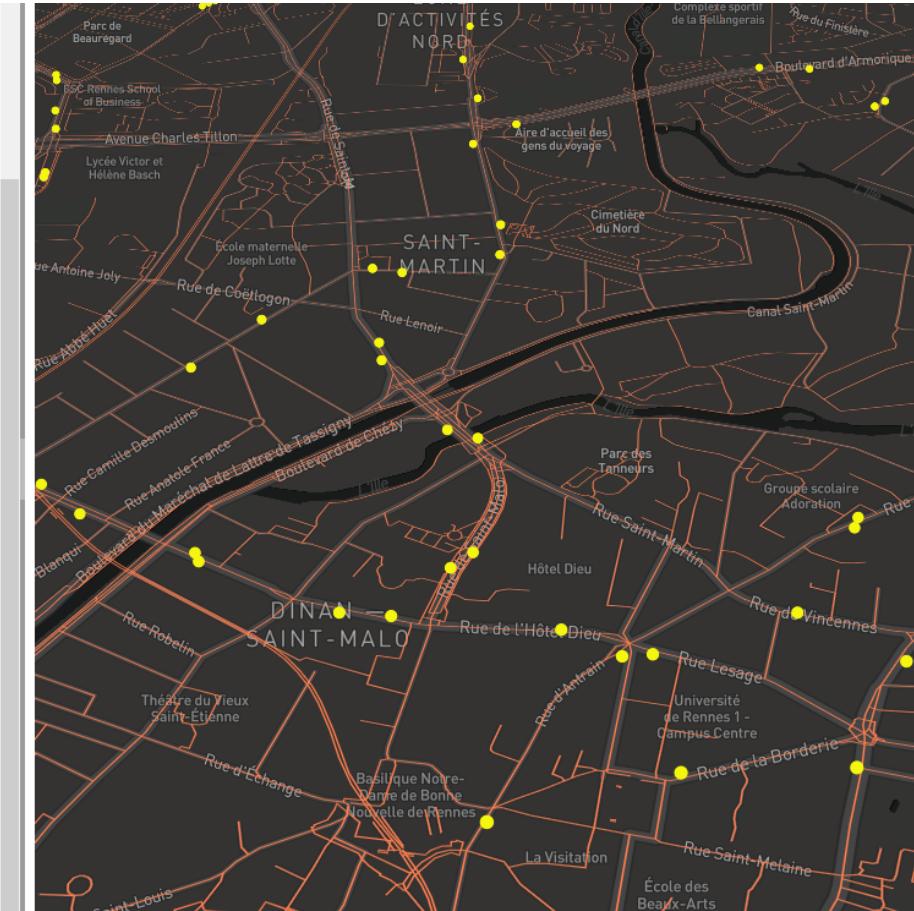
```
// Arret de bus

map.addSource('Arrets', {
    type: 'vector',
    url: 'mapbox://ninanoun.58widelk'});
Bien renseigner l'ID de votre Tilesets

map.addLayer({
    'id': 'Arrets',
    'type': 'circle',
    'source': 'Arrets',
Bien renseigner le nom la couche du tileset
    'source-layer': 'Bus-5ypx1k',
    'layout': {'visibility': 'visible'},
    'paint': {'circle-radius': {'base': 1.5,'stops': [[13, 2], [22, 60]]}, 'circle-color': '#f31616'}, minzoom:12
});
```

Ajout des arrêts de bus

```
19+ <script>
20+   // AccesToken
21  mapboxgl.accessToken = 'pk.eyJ1IjoibmluYW5vdW4iLCJhIjoiY2pjdhBoZG1zMnV4dDJxcGc5azJkbWRiYSJ9.o4dZRrdHcgVEKCveOXG1Y
22
23  // Configuration de la carte
24  var map = new mapboxgl.Map({
25    container: 'map',
26    style: 'mapbox://styles/mapbox/dark-v9',
27    center: [-1.68, 48.12], // lat/long
28    zoom: 15, // zoom
29    pitch: 50, // Inclinaison
30    bearing: -10 // Rotation
31  });
32
33  map.on('load', function () {
34
35  map.addSource('mapbox-streets-v7', {
36    type: 'vector',
37    url: 'mapbox://mapbox.mapbox-streets-v7'});
38
39  map.addLayer({
40    "id": "routes",
41    "type": "line",
42    "source": "mapbox-streets-v7",
43    "layout": {"visibility": 'visible'},
44    "source-layer": "road",
45    "paint": {"line-color": "#FF7F50", "line-width": 1}
46  });
47
48+ map.addSource('Arrets', {
49  type: 'vector',
50  url: 'mapbox://ninanoun.58widelk'});
51
52+ map.addLayer({
53  'id': 'Arrets',
54  'type': 'circle',
55  'source': 'Arrets',
56  'source-layer': 'Bus-Syypx1k',
57  'layout': {'visibility': 'visible'},
58  'paint': {'circle-radius': 5, 'circle-color': '#f5f60d',}
59  });
60
61
62  });
63
64 </script>
```



Ajout la couche équipements publics

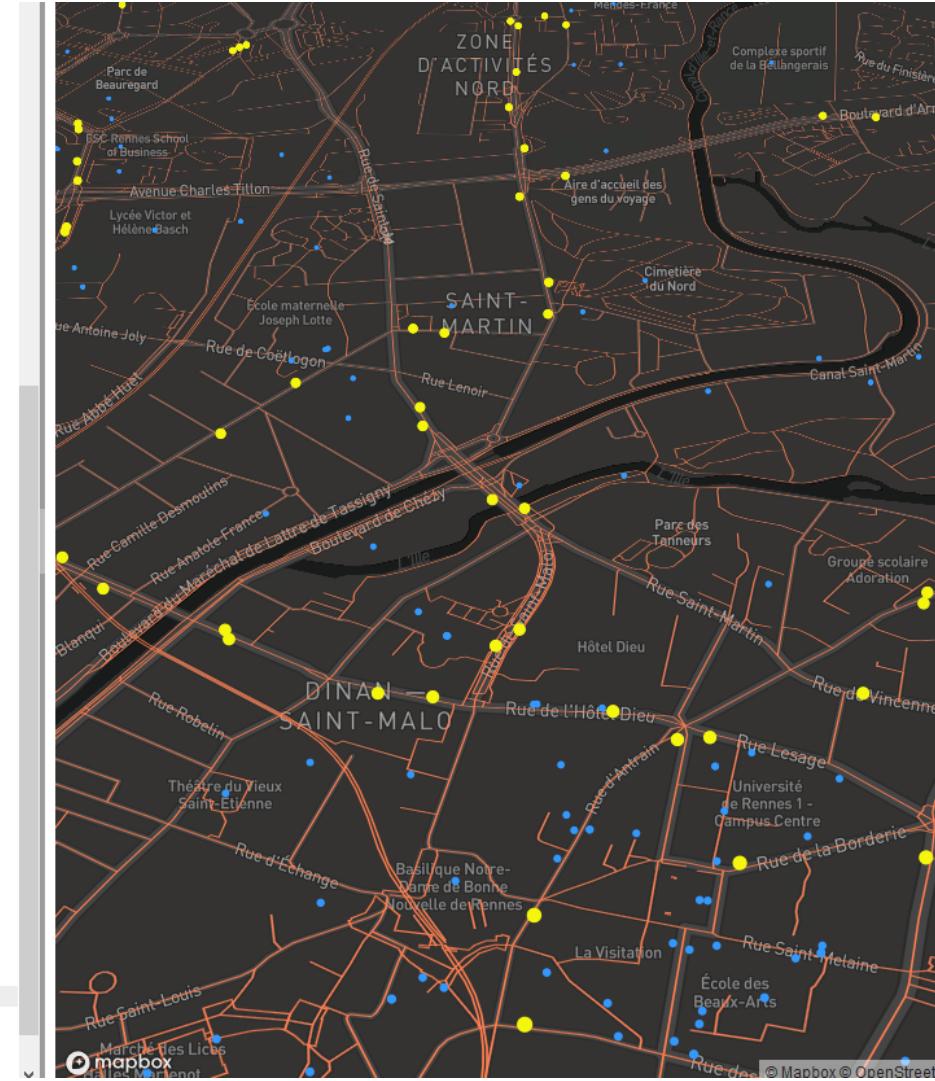
```
// Equipements publics

map.addSource('Equipements', {
  type: 'vector',
  url: 'mapbox://ninanoun.4xcn5ude'});
Bien renseigner l'ID de votre Tilesets

map.addLayer({
  'id': 'Equipements',
  'type': 'circle',
  'source': 'Equipements',
'source-layer': 'base-orga-var-6k0zky',
  Bien renseigner le nom de la couche a afficher
  'layout': {'visibility': 'visible'},
  'paint': {'circle-radius': {'base': 1.5,'stops': [[13, 2], [22, 60]]}, 'circle-color': '#16f337'}, minzoom:14
});
```

Ajout la couche équipements

```
33+ map.on('load', function () {
34
35 // Ajout routes OSM
36
37+ map.addSource('mapbox-streets-v7', {
38   type: 'vector',
39   url: 'mapbox://mapbox.mapbox-streets-v7'});
40
41 map.addLayer({
42   "id": "routes",
43   "type": "line",
44   "source": "mapbox-streets-v7",
45   "layout": {"visibility": "visible"},
46   "source-layer": "road",
47   "paint": {"line-color": "#FF7F50", "line-width": 1}
48 });
49
50 // Ajout arrêts de bus
51
52+ map.addSource('Arrets', {
53   type: 'vector',
54   url: 'mapbox://ninanoun.58wide1k'});
55
56+ map.addLayer({
57   "id": 'Arrets',
58   "type": 'circle',
59   "source": 'Arrets',
60   "source-layer": 'Bus-5pxl1k',
61   "layout": {"visibility": "visible"},
62   "paint": {"circle-radius": 5, "circle-color": '#f5f60d'}
63 });
64
65 // Ajout couche équipements
66
67+ map.addSource('Equipements', {
68   type: 'vector',
69   url: 'mapbox://ninanoun.4xcn5ude'});
70
71+ map.addLayer({
72   "id": 'Equipements',
73   "type": 'circle',
74   "source": 'Equipements',
75   "source-layer": 'base-orga-var-6k0zky',
76   "layout": {"visibility": "visible"},
77   "paint": {"circle-radius": 3, "circle-color": '#3399ff'}
78 });
79
80 });
81
82 </script>
83
84 
```



Mettre en forme les données

- Pour personnaliser la symbologie des données se référer à la documentation

<https://www.mapbox.com/mapbox-gl-js/style-spec/#layers>

Type d'objets géographiques dans MapboxGL :

- circle (point)
- symbol (point avec pictogramme)
- line (ligne)
- fill (polygone)
- fill-extrusion (polygone 3D)
- ...

Mettre en forme les données

- Changer la taille
- Changer la couleur
 - <http://www.code-couleur.com/>
- Définir des niveaux de zoom (max/min)

```
'paint': {'circle-radius': {'base': 1.5,'stops': [[13, 2], [22, 60]]}, 'circle-color': '#3399ff'}, minzoom:14
```

Ajouter les limites de propriétés



The map displays a complex network of streets and buildings, likely representing a specific area in a city. The streets are outlined in blue, and the buildings are represented by various shades of gray and white.

limites_proprietes-auaqb7

Modified on Dec 16, 2016

Add to style

1 vector layer Stats

limites_proprietes

2 properties | This layer contains mostly LineStrings

Description String

Name String

Replace

Make private

Delete

Map ID

ninanoun.a4kdgiot

Format pbf Type vector Size 103 MB

Zoom extent z14 ~ z20

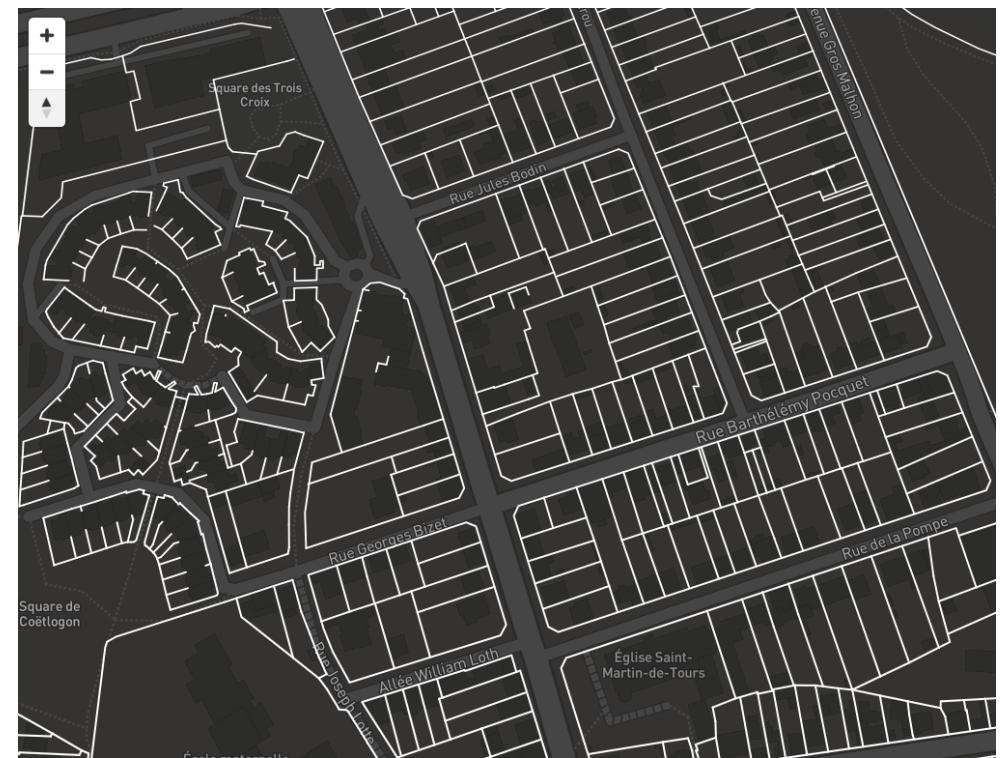
Data will not be visible below zoom 14. Data

Ajouter les limites de propriétés

```
//Proprietes

map.addSource('Proprietes', {
  type: 'vector',
  url: 'mapbox://ninanoun.a4kdgiot'
});

map.addLayer({
  'id': 'Proprietes',
  'type': 'line',
  'source': 'Proprietes',
  'source-layer': 'limites_proprietes',
  'layout': {'visibility': 'visible'},
  'line-join': 'round','line-cap': 'round'},
  'paint': {'line-color': '#FFFFFF', 'line-width': 0.5}
});
```



Ajouts de données 3D – OSM Bâti

```
// Ajout batiments 3D OSM

map.addLayer({
  'id': 'Batiments_3D',
  'source': 'composite',
  'source-layer': 'building',
  'filter': ['==', 'extrude', 'true'],
  'type': 'fill-extrusion',
  'minzoom': 15,
  'paint': {'fill-extrusion-color': '#555555', 'fill-extrusion-height':
  {'type': 'identity','property': 'height'},
  'fill-extrusion-base': {'type': 'identity','property': 'min_height'},
  'fill-extrusion-opacity': 0.8
  }
});
```

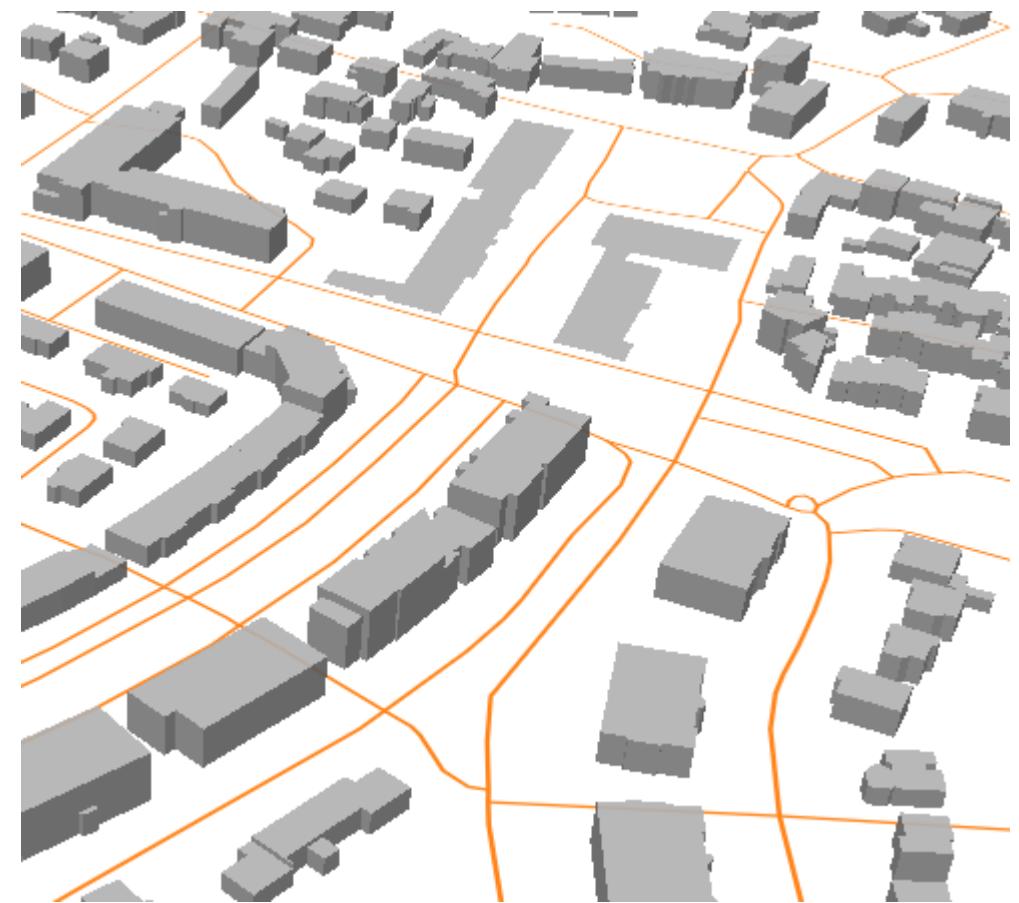


Ajouts de données 3D – BDTOPO Bâti

```
//BATIMENTS

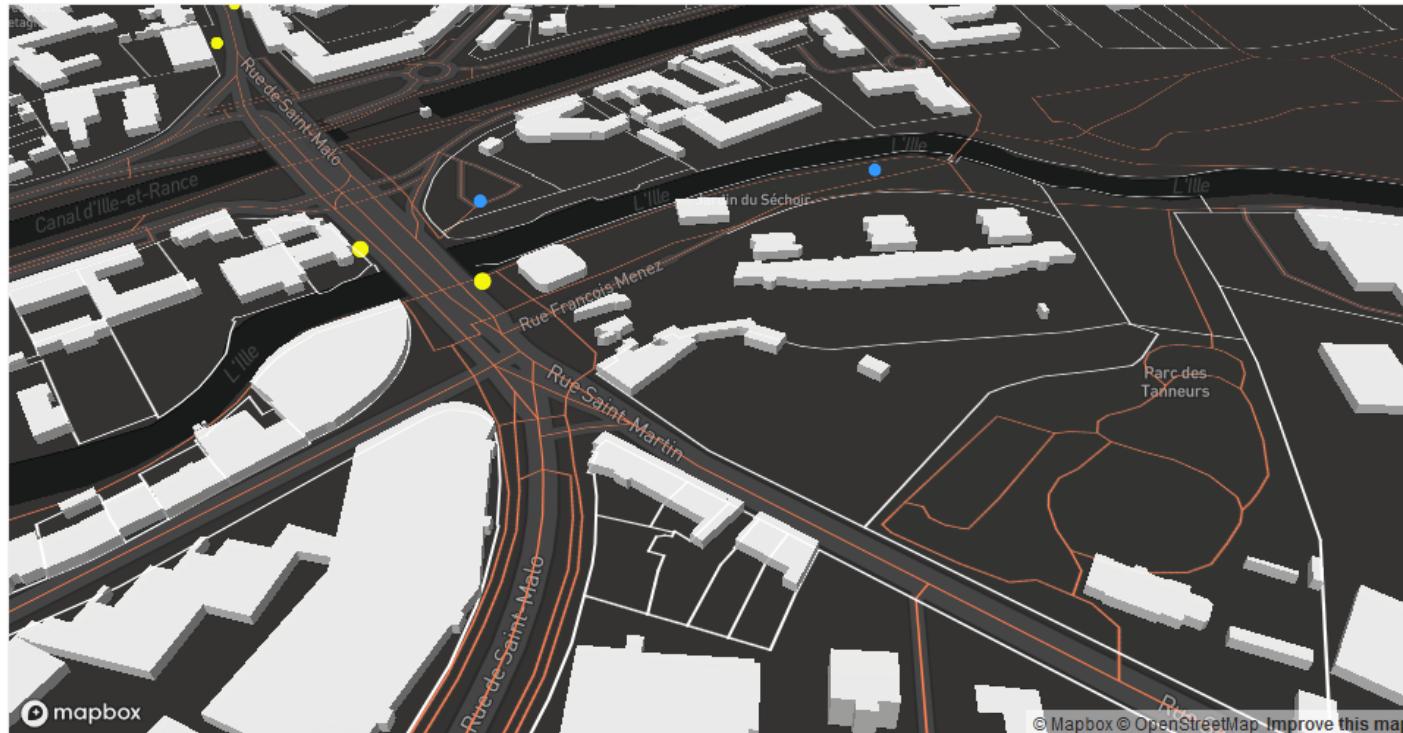
map.addSource('Batiments', {
  type: 'vector',
  url: 'mapbox://mastersigat.a4h4ovrl'
});

map.addLayer({
  'id': 'Batiments',
  'type': 'fill-extrusion',
  'source': 'Batiments',
  'source-layer': 'batiIGN-8zf03o',
  'layout': {'visibility': 'visible'},
  'paint':
    {'fill-extrusion-color': '#A9A9A9',
     'fill-extrusion-height': {'type': 'identity', 'property':
      'HAUTEUR'},
     'fill-extrusion-opacity': 0.90,
     'fill-extrusion-base': 0}
});
```



Exemple

#MapboxGL / Ajout de données personnelles



Built with [blockbuilder.org](#)

[Open](#)

<https://codepen.io/BorisMericskay/pen/YzjJpdQ>

Ajouter des données en local / en ligne

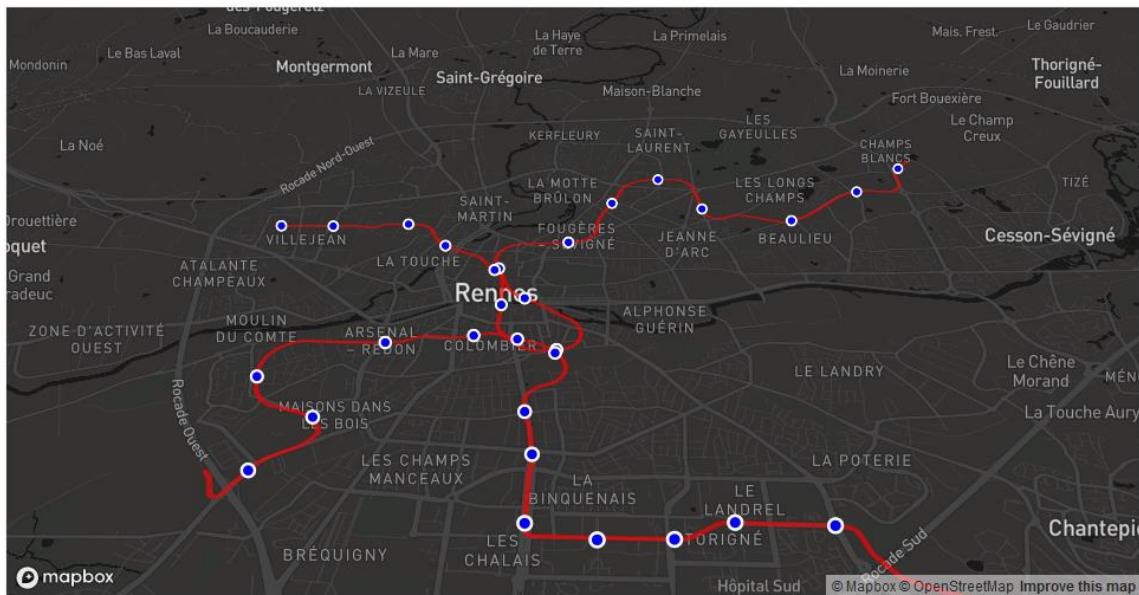
- Il est possible de mobiliser des jeux de données (Geojson) stockés en local (même dossier que la page html) ou accessible via une URL

```
map.on("load", function() {  
    //Couche EPCI  
    map.addLayer({  
        id: "epci",  
        type: "line",  
        source: {type: "geojson",  
            data: "./epci.geojson" },  
        paint: {'line-color': '#000000',  
            'line-width':1}  
    });  
});
```

```
map.on("load", function() {  
    //Couche EPCI  
    map.addLayer({  
        id: "epci",  
        type: "line",  
        source: {type: "geojson",  
            data: 'URL'  
        paint: {'line-color': '#000000',  
            'line-width':1}  
    });  
});
```

Ajouter des données issues de GitHub/Dropbox

#MapboxGL / Ajouter des Geojson - hébergés sur Github

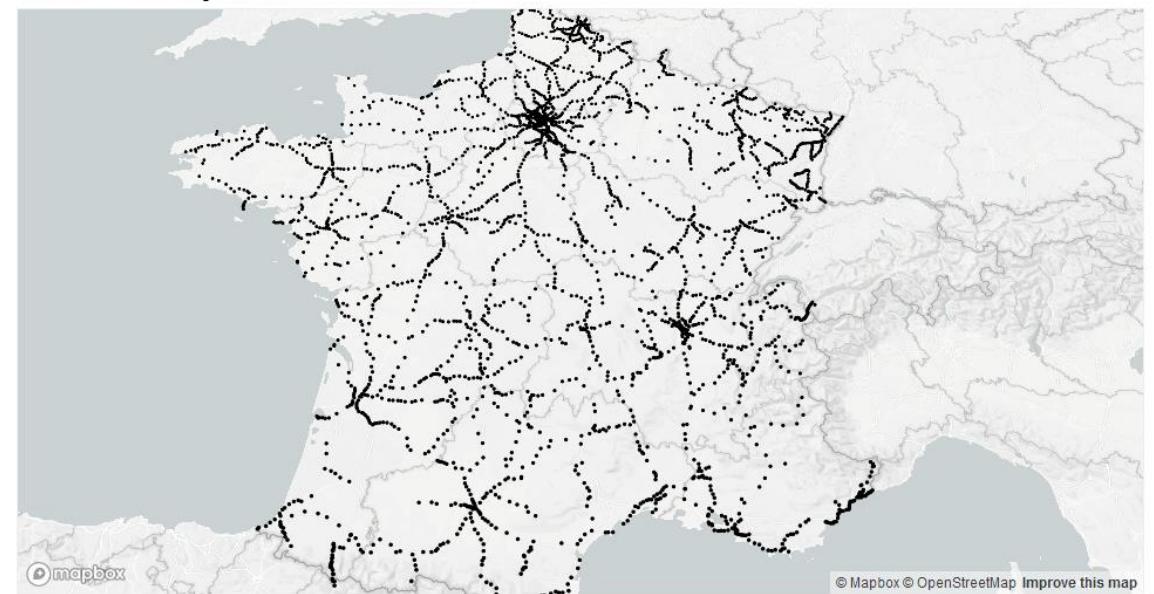


Built with [blockbuilder.org](#)

[Open](#)

<https://codepen.io/BorisMericskay/pen/jOpeVJd>

#MapboxGL / Afficher des Geojson hébergés sur une Dropbox



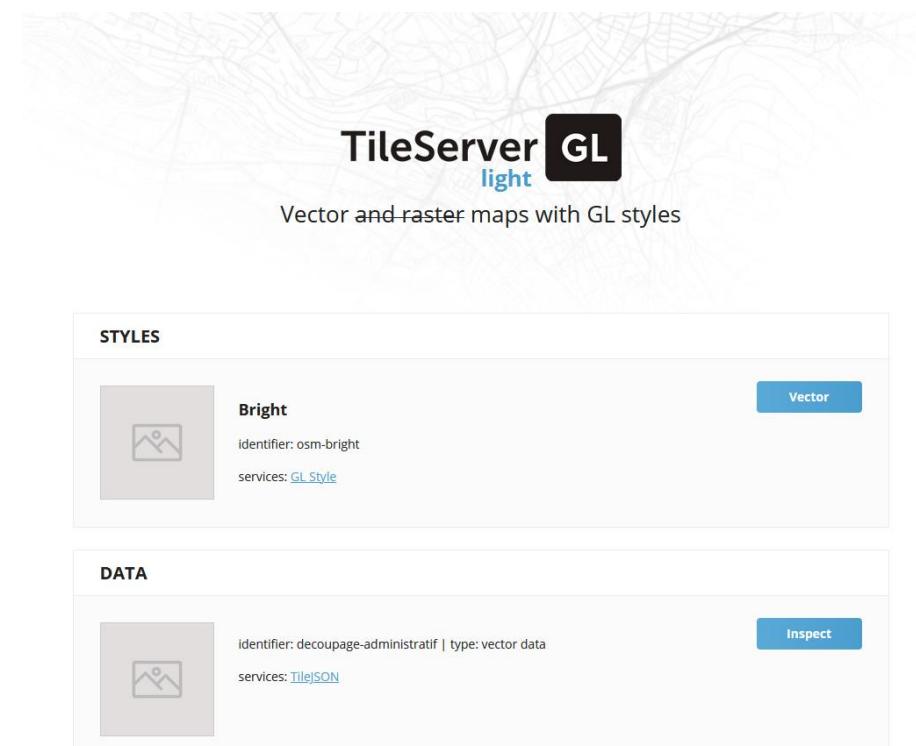
Built with [blockbuilder.org](#)

[Open](#)

<https://codepen.io/BorisMericskay/pen/gOjBgpK>

Ajout de tuiles vectorielles externes

- <https://openmaptiles.geo.data.gouv.fr/>



Ajouter des données externes

- Données sur un portail opendata (Geojson)

```
// Ajout lignes de metros

map.addSource('lignes', {
  type: 'geojson',
  data: 'https://data.rennesmetropole.fr/api/explore/v2.1/catalog/datasets/metro-du-reseau-star-traces-de-laxe-des-lignes/exports/geojson'
});

map.addLayer({
  'id': 'lignesmetros',
  'type': 'line',
  'source': 'lignes',
  'paint': {'line-opacity': 0.7, 'line-width': 3.5,
            'line-color': 'red'}
});
```

Ajouter des données externes

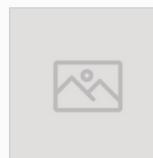
- Données sur Github (Geojson)

```
// Ajout lignes de metros

map.addSource('lignes', {
  type: 'geojson',
  data: 'https://raw.githubusercontent.com/mastersigat/main/metro-du-reseau-star-traces-de-laxe-des-lignes.geojson'
});

map.addLayer({
  'id': 'lignesmetros',
  'type': 'line',
  'source': 'lignes',
  'paint': {'line-opacity': 0.7, 'line-width': 3.5,
            'line-color': 'red'}
});
```

Ajout du cadastre (flux de tuiles vectorielles)

DATA	
	identifier: cadastre type: vector data services: TileJSON Inspect
	identifier: ifremer-atlasdce type: vector data services: TileJSON Inspect
	identifier: decoupage-administratif type: vector data services: TileJSON Inspect
	identifier: france-vector type: vector data services: TileJSON Inspect

<https://openmaptiles.geo.data.gouv.fr/>

Ajout du cadastre (flux de tuiles vectorielles)

```
// AJOUT DU CADASTRE ETALAB

map.addSource('Cadastre', {
    type: 'vector',
    url: 'https://openmaptiles.geo.data.gouv.fr/data/cadastre.json'  });

map.addLayer({
    'id': 'Cadastre',
    'type': 'line',
    'source': 'Cadastre',
    'source-layer': 'parcelles',
    'layout': {'visibility': 'visible'},
    'paint': {'line-color': '#000000'},
    'minzoom':16, 'maxzoom':19 });

map.setPaintProperty('communeslimites', 'line-width', ["interpolate",["exponential",1],["zoom"],16,0.3,18,1]);
```

Ajouter des données externes / IGN

The screenshot shows the IGN géoservices documentation page. The header includes the French Republic logo, the word "géoservices", a search bar, and a "CONNEXION" button. The main navigation menu has links for PRESENTATION, USAGES, CATALOGUE, SERVICES WEB, TÉLÉCHARGEMENT, DOCUMENTATION, ACTUALITÉS, and BASCULE. Below the menu, a breadcrumb trail shows the path: ACCUEIL > DOCUMENTATION > Services > API et services OGC > Tuiles vectorielles – TMS/WMTS > Connexion et requêtes. A large blue header "DOCUMENTATION" is followed by a sidebar titled "SERVICES" with sections for "Services Géoplateforme", "Disponibilité", "API et services OGC" (which is expanded to show "API Carto (REST)", "Calcul altimétrique (REST)", "Calcul d'isochrones - isodistances", "Calcul d'itinéraires", "Données vecteur - WFS (OGC)", and "Géocodage 2.0"). The main content area is titled "CONNEXION ET REQUÊTES" and contains a "Table des matières" with links to "Connexion au service de tuiles vectorielles" and "Requêtes possibles". A section titled "Connexion au service de tuiles vectorielles" explains the service's purpose and provides URLs for TMS and WMTS protocols.

The screenshot shows a JSON viewer interface with tabs for "JSON", "Données brutes", and "En-têtes". The JSON data describes a vector tile service. It includes fields like "name" (BDTOPO), "description" (La BD TOPO® contient une... de précision métrique.), "minzoom" (4), "maxzoom" (19), "crs" (EPSG:3857), and "format" (pbf). The "vector_layers" field contains three entries, each defining a layer with an "id", "geometry" (GEOMETRY), "maxzoom" (19), "minzoom" (15), "fieldsCount" (11 or 7), and "fields" (empty array). The third layer is highlighted in blue.

```
JSON Données brutes En-têtes
Enregistrer Copier Tout réduire Tout développer (lent) Filtrer le JSON
name: "BDTOPO"
description: "La BD TOPO® contient une... de précision métrique."
minzoom: 4
maxzoom: 19
crs: "EPSG:3857"
center: [...]
bounds: [...]
format: "pbf"
tiles: [...]
vector_layers:
  0:
    id: "ligne_orographique"
    geometry: "GEOMETRY"
    maxzoom: "19"
    minzoom: "15"
    fieldsCount: "11"
    fields: [...]
  1:
    id: "toponymie_lieux_nommes"
    geometry: "GEOMETRY"
    maxzoom: "19"
    minzoom: "15"
    fieldsCount: "7"
    fields: [...]
```

<https://geoservices.ign.fr/documentation/services/api-et-services-ogc/tuiles-vectorielles-tmswmts/connexion>

Ajouter des données externes / IGN

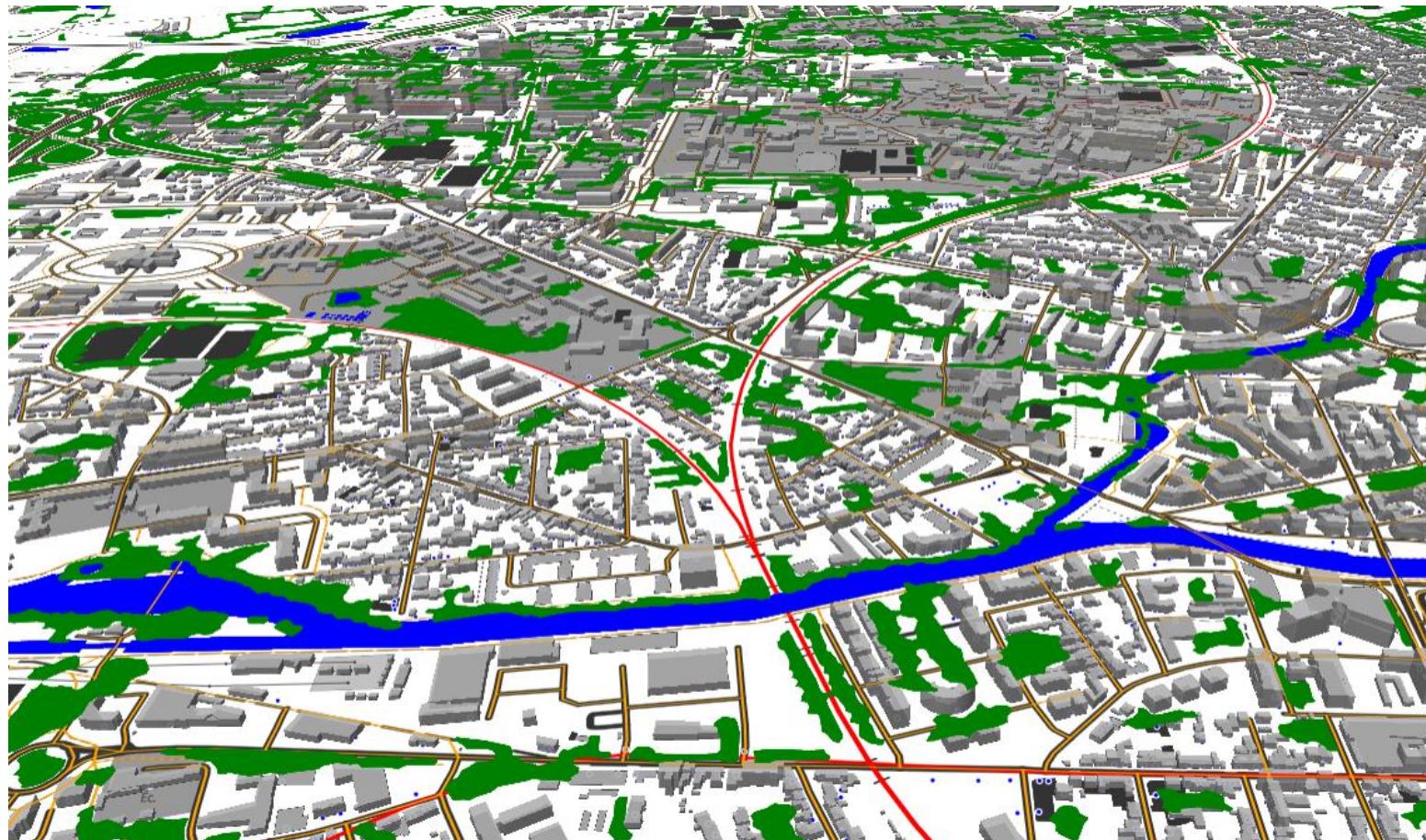
- Données en tuiles vectorielles

```
// Ajout BDTOPO

map.addSource('BDTOPO', {
  type: 'vector',
  url: 'https://wxs.ign.fr/topographie/geoportail/tms/1.0.0/BDTOPO/metadata.json',
  minzoom: 15,
  maxzoom: 19
});

map.addLayer({
  'id': 'batiments',
  'type': 'fill-extrusion',
  'source': 'BDTOPO',
  'source-layer': 'batiment',
  'paint': {'fill-extrusion-color': '#A9A9A9',
            'fill-extrusion-height':{'type': 'identity','property': 'hauteur'},
            'fill-extrusion-opacity': 0.90,
            'fill-extrusion-base': 0}
});
```

Ajouter des données externes



<https://codepen.io/BorisMericskay/pen/MWLjJqx>

Ajouter des données externes via des API's

- URL des API's
 - Contour d'une commune: https://api.gouv.fr/documentation/api_carto_cadastre
 - RPG : https://api.gouv.fr/documentation/api_carto_rpg
 - PLU : <https://apicarto.ign.fr/api/doc/gpu#/>
 - Parkings relais : <https://data.explore.star.fr/explore/dataset/tco-parcsrelais-star-etat-tr/api/>
 - VLS : <https://data.explore.star.fr/explore/dataset/vls-stations-etat-tr/api/>
- Besoin de jQuery (<https://fr.wikipedia.org/wiki/JQuery>)

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js" ></script>
```

API Carto Cadastre

- Contours commune: https://api.gouv.fr/documentation/api_carto_cadastre

```
dataCadastre = 'https://apicarto.ign.fr/api/cadastre/commune?code_insee=35238';

jQuery.when( jQuery.getJSON(dataCadastre)).done(function(json) {
    for (i = 0; i < json.features.length; i++) {
        json.features[i].geometry = json.features[i].geometry;
    };

    map.addLayer(
    {'id': 'Contourcommune',
     'type':'line',
     'source': {'type': 'geojson','data': json},
     'paint' : {'line-color': 'black',
                'line-width':2.5},
     'layout': {'visibility': 'visible'},
    });
});
```

API Carto RPG

- RPG : https://api.gouv.fr/documentation/api_carto_rpg

```
dataRPG =
'https://apicarto.ign.fr/api/rpg/v2?annee=2021&geom=%7B%22type%22%3A%20%22Point%22%2C%22coordinates%22%3A%5B-
1.647%2C48.146%5D%7D&_limit=100';

jQuery.when( jQuery.getJSON(dataRPG)).done(function(json) {
  for (i = 0; i < json.features.length; i++) {
    json.features[i].geometry = json.features[i].geometry;
  }

  map.addLayer(
    { 'id': 'RPG',
      'type':'fill',
      'source': {'type': 'geojson','data': json},
      'paint' : {'fill-color': 'red'},
      'layout': {'visibility': 'none'}
    });
});
```

API Carto GPU

- PLU : <https://apicarto.ign.fr/api/doc/gpu#/>

```
dataPLU = 'https://apicarto.ign.fr/api/gpu/zone-urba?partition=DU_243500139';

jQuery.when(jQuerygetJSON(dataPLU)).done(function(json) {
    // Filtrer les entités pour ne garder que celles avec typezone = 'U'
    var filteredFeatures = json.features.filter(function(feature)
        {return feature.properties.typezone === 'N';});

    // Créer un objet GeoJSON avec les entités filtrées
    var filteredGeoJSON = { type: 'FeatureCollection', features: filteredFeatures};

    map.addLayer({
        'id': 'PLU',
        'type': 'fill',
        'source': {'type': 'geojson',
            'data': filteredGeoJSON},
        'paint': {'fill-color': 'green',
            'fill-opacity': 0.5},
    });
});
```

API STAR

- Parkings relais : <https://data.explore.star.fr/explore/dataset/tco-parcsrelais-star-etat-tr/api/>

```
$.getJSON('https://data.rennesmetropole.fr/api/explore/v2.1/catalog/datasets/tco-parcsrelais-star-etat-tr/records?limit=20',
function(data) {var geojsonData4 = {
    type: 'FeatureCollection',
    features: data.results.map(function(element) {
        return {type: 'Feature',
            geometry: {type: 'Point',
                coordinates: [element.coordonnees.lon, element.coordonnees.lat]},
            properties: { name: element.nom,
                capacity: element.jrdinfosoliste}};
    })
};

map.addLayer({ 'id': 'Parcrelais',
    'type':'circle',
    'source': {'type': 'geojson',
        'data': geojsonData4},
    'paint': {'circle-color': 'orange'}
});});
```

API STAR

- VLS : <https://data.explore.star.fr/explore/dataset/vls-stations-etat-tr/api/>
- Position des bus en temps réel:
<https://data.explore.star.fr/explore/dataset/tco-bus-vehicules-position-tr>

Ajouter des données OSM via l'API

```
const ville = "Rennes";

$.getJSON(`https://overpass-api.de/api/interpreter?data=[out:json];area[name="${ville}"]->.searchArea;(node["amenity"]="bar"](area.searchArea));;out center;`,

function(data) {var geojsonData = {
  type: 'FeatureCollection',
  features: data.elements.map(function(element) {
    return {type: 'Feature',
      geometry: { type: 'Point', coordinates: [element.lon, element.lat] },
      properties: {}};
  })
};

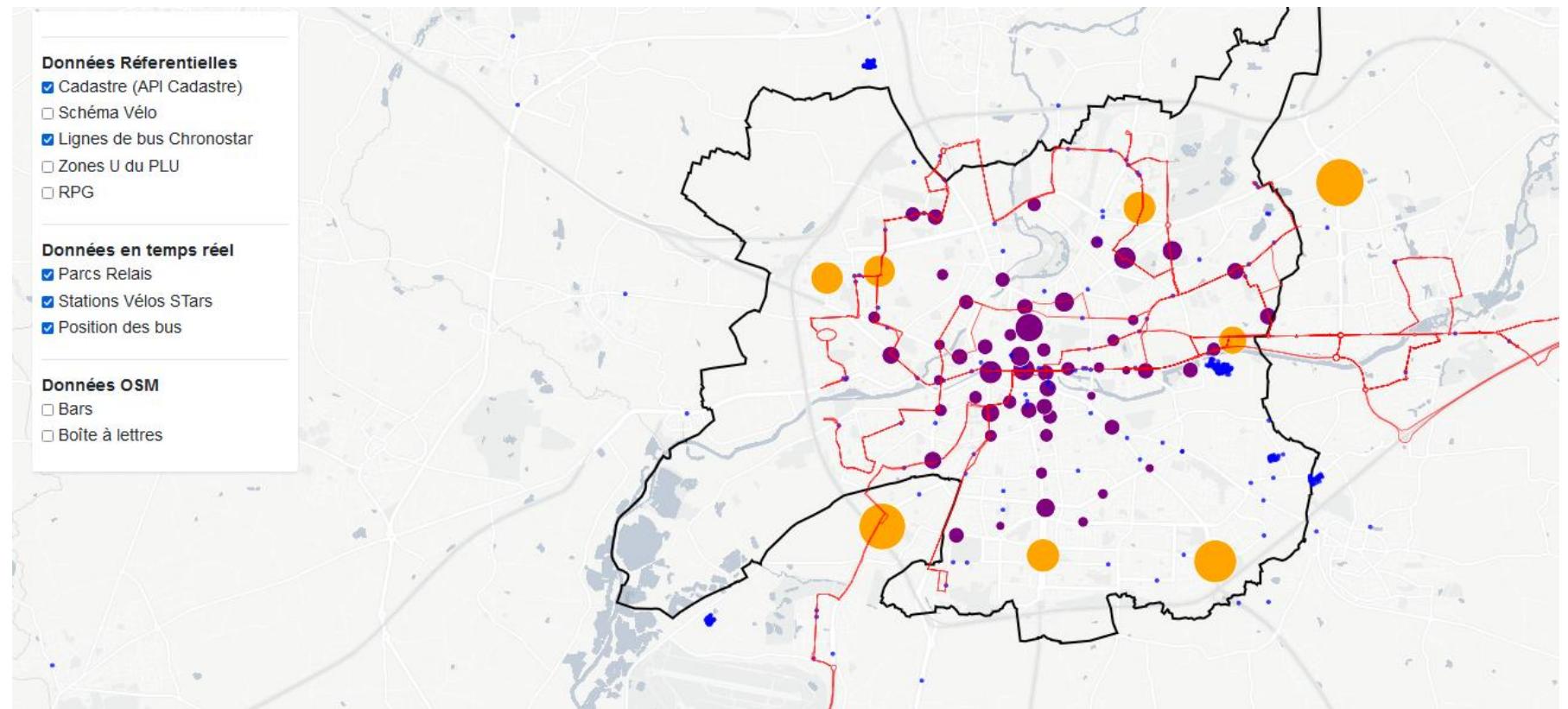
map.addSource('customData', {
  type: 'geojson',
  data: geojsonData
});

map.addLayer({
  'id': 'pubs',
  'type': 'circle',
  'source': 'customData',
  'paint': {'circle-color': 'green',
            'circle-radius': 5},
  'layout': {'visibility': 'none'}
});

});
```

Ajouter des données externes

- Via des API



<https://codepen.io/BorisMericskay/pen/GRXYjJr>

Interactivité avec les données

Hover et Click

Interactivité avec les données / Hover

- Hover de d'une couche (survol) = couche arrêts
 - Cette commande doit être placée à la fin du script

```
//Interactivité HOVER

var popup = new mapboxgl.Popup({
  closeButton: false,
  closeOnClick: false });

map.on('mousemove', function(e) {
  var features = map.queryRenderedFeatures(e.point, { layers: ['Equipements'] });
  // Change the cursor style as a UI indicator.
  map.getCanvas().style.cursor = (features.length) ? 'pointer' : '';

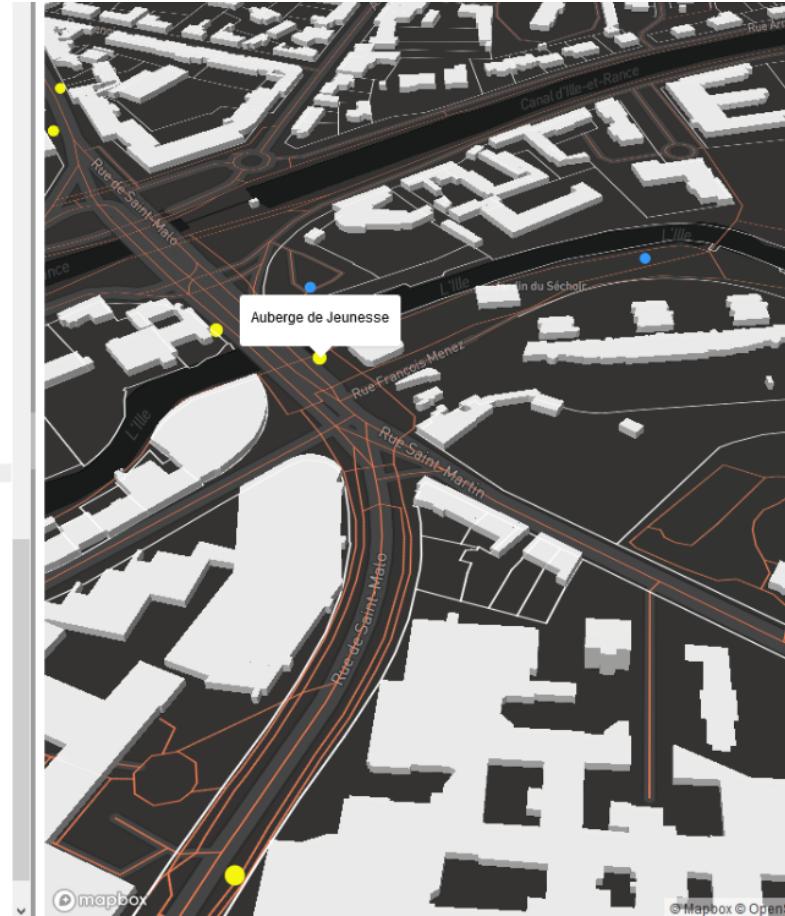
  if (!features.length) {
    popup.remove();
    return;
  }

  var feature = features[0];
  popup.setLngLat(feature.geometry.coordinates)
    .setHTML(feature.properties.organom)
    .addTo(map);

});
```

Interactivité avec les données / Hover

```
90      "type": "line",
91      "source": "Proprietes",
92      "source-layer": "limites_proprietes",
93      "layout": {"visibility": "visible",
94      "line-join": "round", "line-cap": "round"},
95      "paint": {"line-color": "#FFFFFF", "line-width": 1.5}
96    });
97
98    // Ajout batiments 3D
99
100   map.addLayer({
101     "id": "Batimenten_3D",
102     "source": "composite",
103     "source-layer": "building",
104     "filter": ["==", "extrude", "true"],
105     "type": "fill-extrusion",
106     "minzoom": 15,
107     "paint": {"fill-extrusion-color": "#FFFFFF", "fill-extrusion-height": {
108       "type": "identity", "property": "height"},
109     "fill-extrusion-base": {"type": "identity", "property": "min_height"},
110     "fill-extrusion-opacity": 0.9
111   }
112 });
113
114 });
115 //Interactivité HOVER
116
117 var popup = new mapboxgl.Popup({
118   closeButton: false,
119   closeOnClick: false
120 });
121
122 map.on('mousemove', function(e) {
123   var features = map.queryRenderedFeatures(e.point, { layers: ['Arrets'] });
124   // Change the cursor style as a UI indicator.
125   map.getCanvas().style.cursor = (features.length) ? 'pointer' : '';
126
127   if (!features.length) {
128     popup.remove();
129     return;
130   }
131
132   var feature = features[0];
133   popup.setLngLat(feature.geometry.coordinates)
134   .setHTML(feature.properties.nom)
135   .addTo(map);
136 });
137
138 </script>
139
```



Interactivité avec les données / Hover

- Hover de deux couches (survol)

```
//Interactivité HOVER

var popup = new mapboxgl.Popup({
  closeButton: false,
  closeOnClick: false });

map.on('mousemove', function(e) {
  var features = map.queryRenderedFeatures(e.point, { layers: ['Arrets'] ['Equipements'] });
  // Change the cursor style as a UI indicator.
  map.getCanvas().style.cursor = (features.length) ? 'pointer' : '';

  if (!features.length) {
    popup.remove();
    return;
  }

  var feature = features[0];

  popup.setLngLat(feature.geometry.coordinates)
    .setHTML(feature.properties.nom)
    .addTo(map);

  var feature = features[1];

  popup.setLngLat(feature.geometry.coordinates)
    .setHTML(feature.properties.organom)
    .addTo(map);
});
```

Interactivité avec les données / Hover

```
115
116 //Interactivité HOVER
117
118 var popup = new mapboxgl.Popup({
119   closeButton: false,
120   closeOnClick: false });
121
122 map.on('mousemove', function(e) {
123   var features = map.queryRenderedFeatures(e.point, { layers: ['Arrets'] ['Equipements'] });
124   // Change the cursor style as a UI indicator.
125   map.getCanvas().style.cursor = (features.length) ? 'pointer' : '';
126
127 if (!features.length) {
128   popup.remove();
129   return;
130 }
131
132 var feature = features[0];
133
134 popup.setLngLat(feature.geometry.coordinates)
135 .setHTML(feature.properties.nom)
136 .addTo(map);
137
138 var feature = features[1];
139
140 popup.setLngLat(feature.geometry.coordinates)
141 .setHTML(feature.properties.organom)
142 .addTo(map);
143 });
144
145
146 </script>
147
```

Modifier le style d'une Popup

Ajouter dans le Style

```
.Mypopup .mapboxgl-popup-content {  
    background-color: black;  
    color : white;  
    opacity : 0.7;  
}
```

Ajouter l'appel du CSS dans l'appel de la Popup

```
var popup = new mapboxgl.Popup({  
    className: "Mypopup",  
    closeButton: false,  
    closeOnClick: false});
```

Modifier le style du contenu d'une popup

- Reprendre la main avec une syntaxe en html dans le contenu à afficher
 - Soyez très vigilant avec les '

```
.setHTML('<b> Nom de l'arrêt</b>' + '<br>' + feature.properties.nom )
```

```
.setHTML('<h3> Nom l'organisme</h3>' + feature.properties.organom )
```

Interactivité avec les données / Click

- Click d'une couche (popup) = couche arrets

```
//Interactivité CLICK

map.on('click', function (e) {
    var features = map.queryRenderedFeatures(e.point, { layers: ['Arrets'] });

    if (!features.length) {
        return;
    }

    var feature = features[0];
    var popup = new mapboxgl.Popup({ offset: [0, -15] })
        .setLngLat(feature.geometry.coordinates)
        .setHTML('<h2>' + feature.properties.nom + '</h2><h3>' +
            +"Mobilier : " + feature.properties.mobilier + '</h3><p>' +
            +"Accessibilité PMR : " + feature.properties.estaccessiblepmr + '</p> ')
        .addTo(map);
});

map.on('mousemove', function (e) {
    var features = map.queryRenderedFeatures(e.point, { layers: ['Arrets'] });
    map.getCanvas().style.cursor = (features.length) ? 'pointer' : '';
});
```

Interactivité avec les données / Click

```
96  });
97  // Ajout batiments 3D
98
99
100 - map.addLayer({
101   'id': 'Batimenst_3D',
102   'source': 'composite',
103   'source-layer': 'building',
104   'filter': ['==', 'extrude', 'true'],
105   'type': 'fill-extrusion',
106   'minzoom': 15,
107   'paint': {'fill-extrusion-color': '#FFFFFF', 'fill-extrusion-height':
108     {'type': 'identity','property': 'height'},
109     'fill-extrusion-base': {'type': 'identity','property': 'min_height'},
110     'fill-extrusion-opacity': 0.9
111   }
112 });
113
114 });
115
116 //Interactivité CLICK
117
118 - map.on('click', function (e) {
119   var features = map.queryRenderedFeatures(e.point, { layers: ['Arrets'] });
120
121 -   if (!features.length) {
122     return;
123   }
124
125   var feature = features[0];
126   var popup = new mapboxgl.Popup({ offset: [0, -15] })
127     .setLngLat(feature.geometry.coordinates)
128     .setHTML('<h2>' + feature.properties.nom + '</h2><h3>' +
129     +"Mobilier : " + feature.properties.mobilier + '</h3><p>' +
130     +"Accessibilité PMR : " + feature.properties.estaccessiblepmr + '</p>' )
131     .addTo(map);
132 });
133
134 map.on('mousemove', function (e) {
135   var features = map.queryRenderedFeatures(e.point, { layers: ['Arrets'] });
136   map.getCanvas().style.cursor = (features.length) ? 'pointer' : '';
137 });
138
139
140 </script>
141
142 </body>
143 </html>
***
```



Interactivité sur le cadastre Etalab

```
//Interactivité CLICK sur cadastre

map.on('click', function (e) {
var features = map.queryRenderedFeatures(e.point, { layers: ['Cadastre'] });
if (!features.length) {
return;
}
var feature = features[0];
var popup3 = new mapboxgl.Popup({ className: "Mypopup2", offset: [0, -15] })
.setLngLat(e.lngLat)
.setHTML(feature.properties.id + '<br>' + 'Numéro:' + feature.properties.numero + '<br>' +
feature.properties.contenance + ' m2')
.addTo(map);
});
map.on('mousemove', function (e) {
var features = map.queryRenderedFeatures(e.point, { layers: ['Cadastre'] });
map.getCanvas().style.cursor = (features.length) ? 'pointer' : '';
});
```

Ajouter une image à la Popup

```

```

```
//Interactivité CLICK

map.on('click', function (e) {
    var features = map.queryRenderedFeatures(e.point, { layers: ['Arrets'] });

    if (!features.length) {
        return;
    }

    var feature = features[0];
    var popup = new mapboxgl.Popup({ offset: [0, -15] })
        .setLngLat(feature.geometry.coordinates)
        .setHTML('<h2>' + feature.properties.nom + '</h2>  <h3>
+ "Mobilier : " + feature.properties.mobilier + '</h3><p>
+ "Accessibilité PMR : " + feature.properties.estaccessiblepmr + '</p> ')
        .addTo(map);
});

map.on('mousemove', function (e) {
    var features = map.queryRenderedFeatures(e.point, { layers: ['Arrets'] });
    map.getCanvas().style.cursor = (features.length) ? 'pointer' : '';
});
```

Ajouter une image à la Popup

The screenshot shows a Mapbox editor interface with the following components:

- Top Bar:** Includes icons for heart, save, settings, and other tools.
- Left Sidebar:** Shows file navigation with items like "0/mapbox-", "1/mapbox-", and a red warning icon.
- Code Editor:** Divided into two panes:
 - CSS:**

```
1 #map { position: absolute; top: 0; bottom: 0; width: 100%; }
```

```
3 .Mypopup .mapboxgl-popup-content {
```

```
4     background-color: black;
```

```
5     color : white;
```

```
6     opacity : 0.7;
```

```
7 }
```

```
8 
```

```
9 
```
 - JS:**

```
96     return;
```

```
97 }
```

```
98 
```

```
99     var feature = features[0];
```

```
100    var popup = new mapboxgl.Popup({ offset: [0, -15] })
```

```
101        .setLngLat(feature.geometry.coordinates)
```

```
102        .setHTML('<h2>' + feature.properties.nom + '</h2> <hr>  <h3>'
```

```
+ "Mobilier : " + feature.properties.mobilier + '</h3><p>'
```

```
+ "Accessibilité PMR : " + feature.properties.estaccessiblepmr +
```

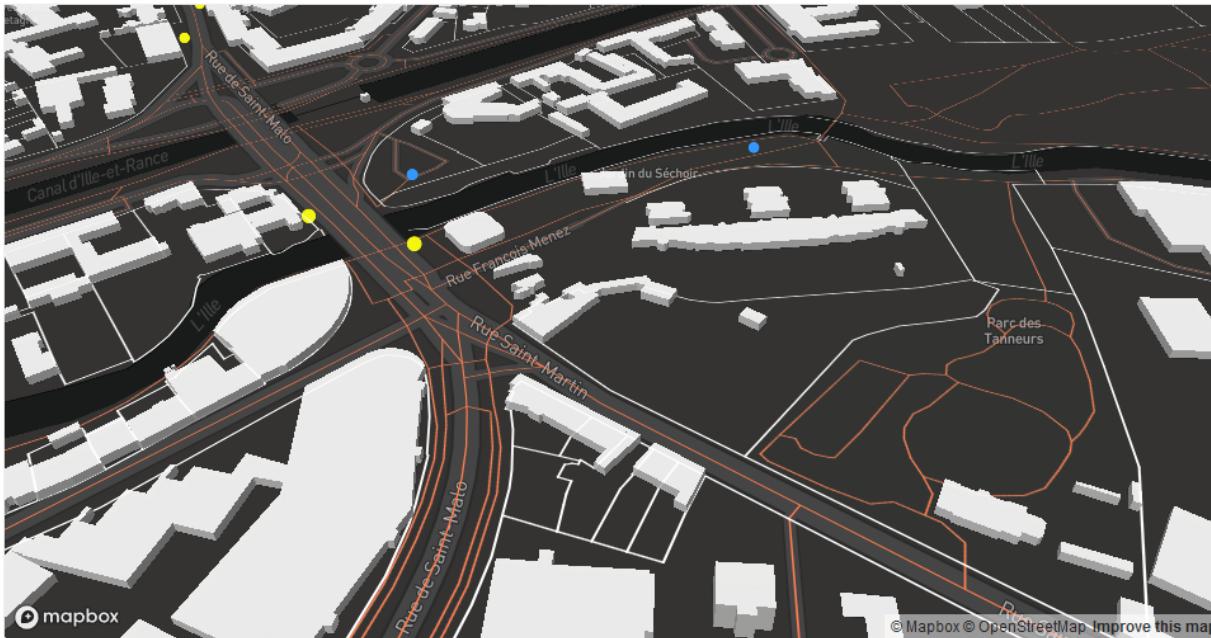
```
'</p>'
```

```
105    .addTo(map);
```

```
106 );
```
- Map View:** A dark-themed map of an urban area with several locations marked by blue dots. One location has a white info box overlay containing:
 - Auberge de Jeunesse**
 - STAR** logo
 - Mobilier : Abri simple**
 - Accessibilité PMR : true**

Exemple

#MapboxGL / Interactivité avec les données
(hover/click)



Built with [blockbuilder.org](#)

[Open](#)

<https://codepen.io/BorisMericskay/pen/JjBmEwp>

Menu de gestion des couches

Ajouter un menu pour gérer les couches

- Première étape: définir le style (CSS) de votre menu

```
#menu {  
width: 20%;  
Z-index: 1;  
top: 10px;  
left: 20px;  
position: absolute;  
opacity: 0.7;  
font-size: 14px;  
font-family: 'Helvetica Neue', Arial, Helvetica, sans-serif;  
}  
  
#menu a {  
border-radius: 5px;  
display: block;  
color: #000000;  
margin: 5px;  
padding: 10px 10px;  
text-align: center;  
font-weight: bold;  
border: solid 2px;  
background-color: #FFFFFF;  
text-decoration: none;  
}  
  
#menu a.active {  
background-color: #CC6600;  
color: #FFFFFF;  
}  
  
#menu a:hover:not(.active) {  
background-color: #CC6600;  
color: #FFFFFF;  
}
```

```
1  <!DOCTYPE html>  
2  <html>  
3  <head>  
4      <meta charset='utf-8' />  
5      <title>MapboxGL</title>  
6  
7      <script src='https://api.tiles.mapbox.com/mapbox-gl-js/v0.44.0/mapbox-gl.js'></script>  
8      <link href='https://api.tiles.mapbox.com/mapbox-gl-js/v0.44.0/mapbox-gl.css' rel='stylesheet' />  
9  
10 <style>  
11     #map { position:absolute; top:0; bottom:0; width:100%; }  
12     #menu {  
13         width: 20%; margin-right: auto; margin-left: auto;  
14         Z-index: 1; top: 10px; right: 10px; position: absolute;  
15         border-color: #FFFFFF; background-color: #808080 ;  
16         font-size: 12px; font-family: 'Helvetica Neue', Arial, Helvetica, sans-serif; }  
17  
18     #menu a {  
19         display: block; color: #FFFFFF; padding: 8px 16px;  
20         text-align: center; font-weight: bold;  
21         border-style: solid; border-color: #000000; }  
22  
23     #menu a.active { background-color: #CC6600;  
24         color: #FFFFFF; }  
25  
26     #menu a:hover:not(.active) {  
27         background-color: #CC6600;  
28         color: #FFFFFF; }  
29  
30 </style>  
31  
32 </head>  
33  
34 <body>  
35     <div id='map'></div>  
36  
37 <script>  
38     // AccessToken
```

Ajouter un menu pour gérer les couches

- Deuxième étape: créer un Div pour votre menu et placer la dans la Div de la carte

```
<div id="menu"></div>
```

```
81      <button id='Rennes2'>Université Rennes 2</button>
82  </div>
83
84  <div id="map"><div id="menu"></div></div>
85
86  <script>
87
88 // Appel de la carte
```

Ajouter un menu pour gérer les couches

- Dernière étape : Ajouter à la fin du script la commande pour configurer votre menu

```
// Gestion du menu des couches

var toggleableLayerIds = ['Routes', 'Arrets', 'Equipements', 'Proprietes', 'BatimentsIGN'];

for (var i = 0; i < toggleableLayerIds.length; i++) {var id = toggleableLayerIds[i];

    var link = document.createElement('a');
    link.href = '#';
    link.className = 'inactive';
    link.textContent = id;

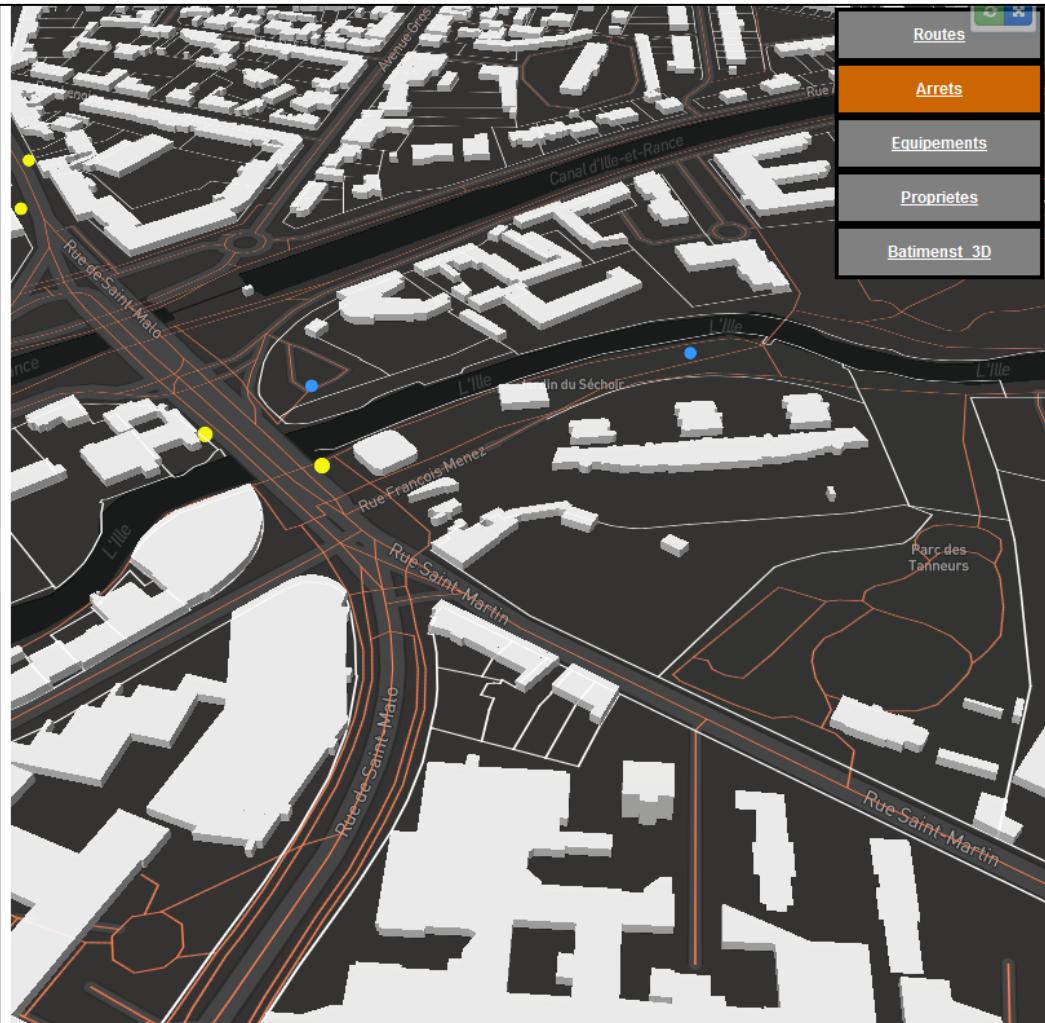
    link.onclick = function (e) {var clickedLayer = this.textContent;
        e.preventDefault();
        e.stopPropagation();
        var visibility = map.getLayoutProperty(clickedLayer, 'visibility');
        if (visibility === 'visible') {
            map.setLayoutProperty(clickedLayer, 'visibility', 'none');
            this.className = "";} else {this.className = 'active';
            map.setLayoutProperty(clickedLayer, 'visibility', 'visible');};

    }

    var layers = document.getElementById('menu'); layers.appendChild(link); }
```

Ajouter un menu pour gérer les couches

```
160
161
162 //Interactivité CLICK
163
164 map.on('click', function (e) {
165   var features = map.queryRenderedFeatures(e.point, { layers: ['Arrets'] });
166
167   if (!features.length) {
168     return;
169   }
170
171   var feature = features[0];
172   var popup = new mapboxgl.Popup({ offset: [0, -15] })
173     .setLngLat(feature.geometry.coordinates)
174     .setHTML('<h2>' + feature.properties.nom + '</h2><h3>' +
175     +"Mobilier : " + feature.properties.mobilier + '</h3><p>' +
176     +"Accessibilité PMR : " + feature.properties.estaccessiblepmr + '</p> ')
177     .addTo(map);
178 });
179
180 map.on('mousemove', function (e) {
181   var features = map.queryRenderedFeatures(e.point, { layers: ['Arrets'] });
182   map.getCanvas().style.cursor = (features.length) ? 'pointer' : '';
183 });
184
185
186 var toggleableLayerIds = ['Routes', 'Arrets', 'Equipements', 'Proprietes', 'Batimenst_3D'];
187
188 for (var i = 0; i < toggleableLayerIds.length; i++) {var id = toggleableLayerIds[i];
189
190   var link = document.createElement('a');
191   link.href = '#';
192   link.className = 'inactive';
193   link.textContent = id;
194
195   link.onclick = function (e) {var clickedLayer = this.textContent;
196   e.preventDefault();
197   e.stopPropagation();
198   var visibility = map.getLayoutProperty(clickedLayer, 'visibility');
199   if (visibility === 'visible') {
200     map.setLayoutProperty(clickedLayer, 'visibility', 'none');
201     this.className = '';} else {this.className = 'active';
202     map.setLayoutProperty(clickedLayer, 'visibility', 'visible');} };
203
204   var layers = document.getElementById('menu'); layers.appendChild(link);
205
206
207 </script>
```



Ajouter un menu pour gérer les couches

- Pour personnaliser si les couches sont active ou pas il suffit de le préciser dans la commande de visibilité *layout*
 - *visible* = couche active
 - *none* = couche non visible

```
map.addLayer({'id': 'Arrets',
              'type': 'circle',
              'source': 'Arrets',
              'source-layer': 'Bus-5ypx1k',
              'layout': {'visibility': 'visible'},
              'paint': {'circle-radius': 7, 'circl
```

```
map.addLayer({'id': 'Arrets',
              'type': 'circle',
              'source': 'Arrets',
              'source-layer': 'Bus-5ypx1k',
              'layout': {'visibility': 'none'},
              'paint': {'circle-radius': 7, 'circle-color': '#f5f60d'}  });
```

Ajouter un menu pour gérer les couches

#MapboxGL / Menu OK



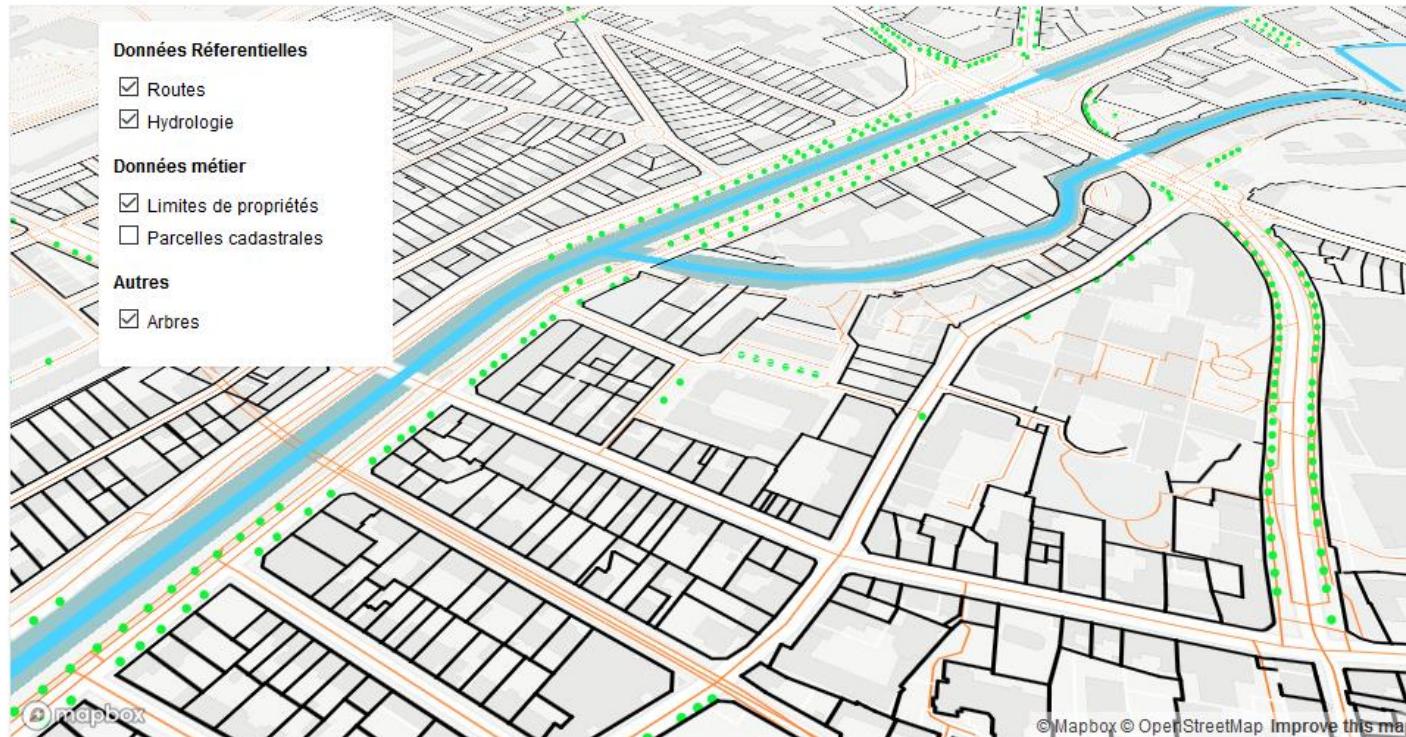
Built with [blockbuilder.org](#)

[Open](#)

<https://codepen.io/BorisMericskay/pen/NWBORVv>

Menu en mode CheckBox

#MapboxGL / Menu Checkbox



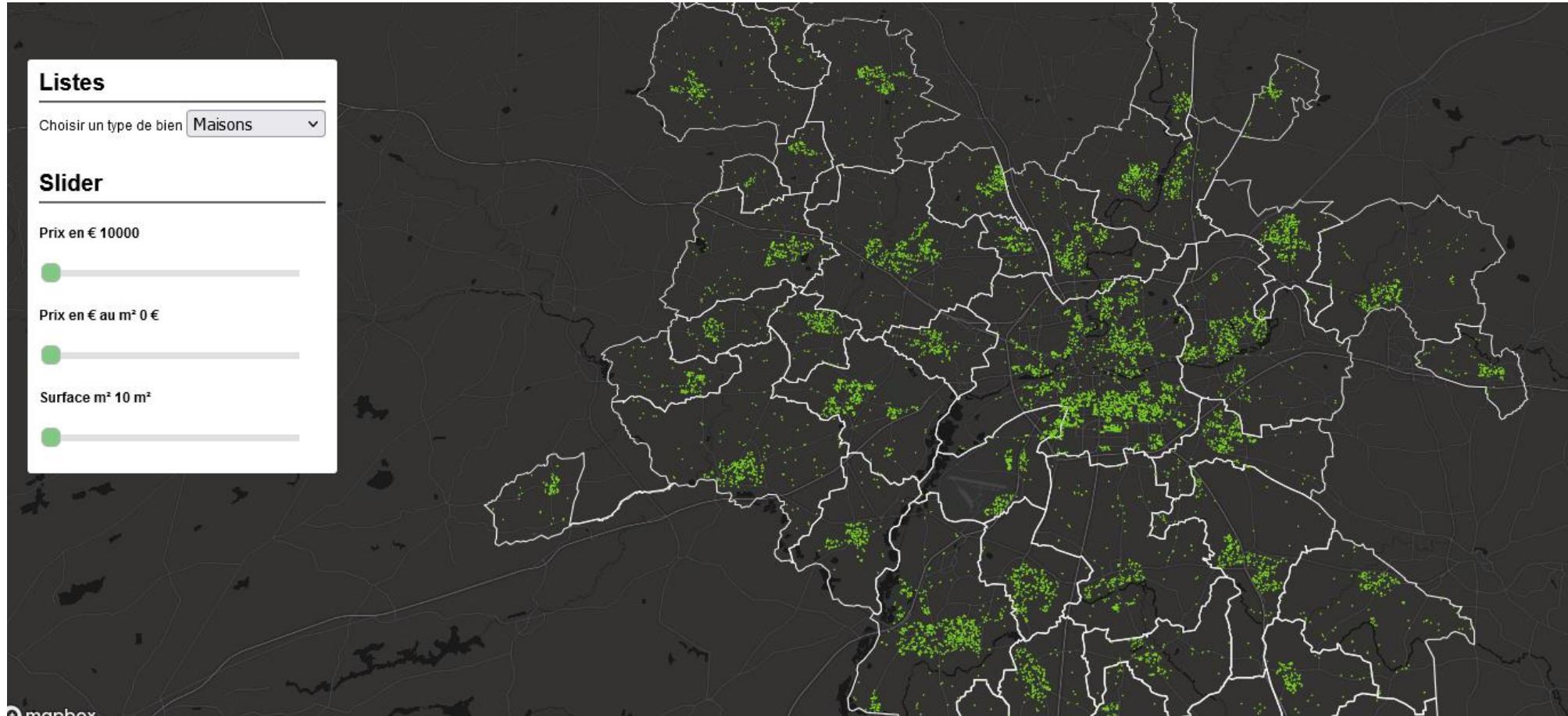
Built with [blockbuilder.org](#)

[Open](#)

<https://codepen.io/BorisMericskay/pen/ZEjmyqQ>

Filtrer des données

Exemple



<https://codepen.io/BorisMericzkay/pen/poZqweG>

Les onglets géographiques

Onglets géographiques

- L'idée est de proposer des boutons pour aller directement à un endroit sur la carte
 - Première étape rajouter une boutons dans la div *map*

```
<div>
  <button id='Gare'>Quartier Gare-Centre</button>
  <button id='Rennes1'>Université Rennes 1</button>
  <button id='Rennes2'>Université Rennes 2</button>
</div>
```

```
<body>
<div id='map'>

  <div>
    <button id='Gare'>Quartier Gare-Centre</button>
    <button id='Rennes1'>Université Rennes 1</button>
    <button id='Rennes2'>Université Rennes 2</button>
  </div>

  <div id="menu"></div>
</div>

<script>
```

Onglets géographiques

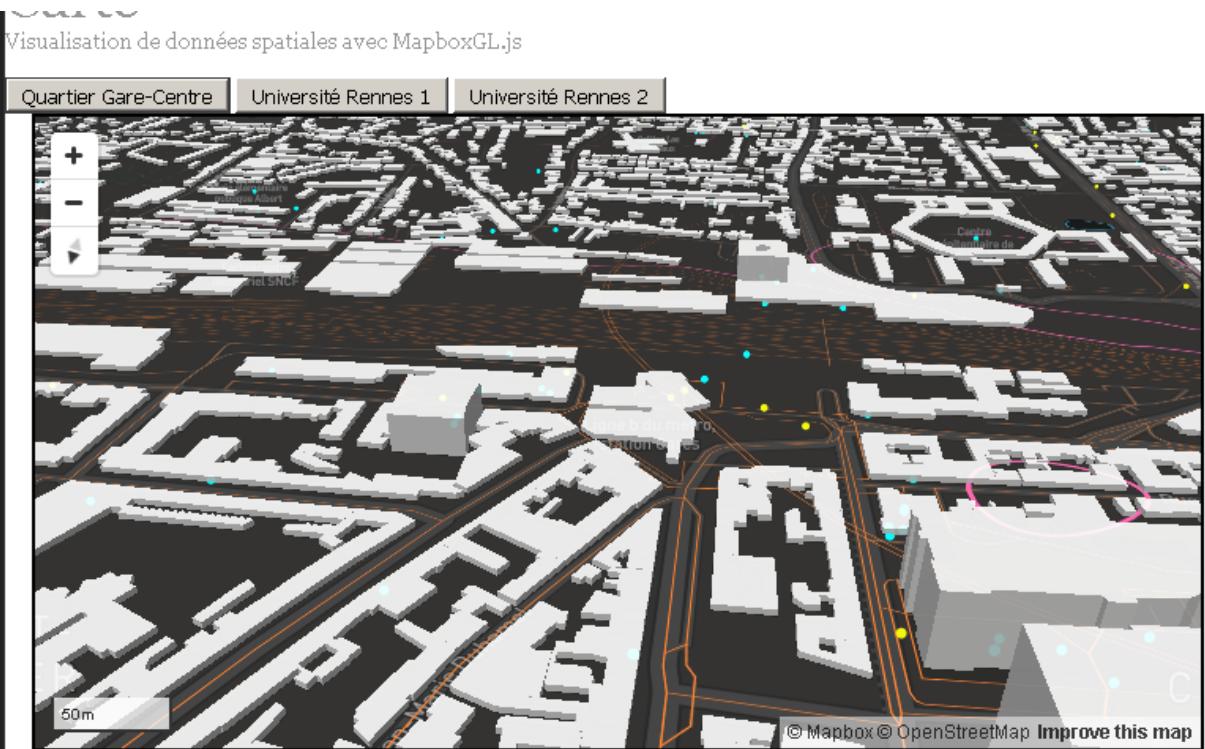
- Seconde étape, rajouter dans le script (à la fin) la configuration des onglets géographiques

```
// Configuration onglets geographiques

document.getElementById('Gare').addEventListener('click', function ()
{ map.flyTo({zoom: 16,
            center: [-1.672, 48.1043],
            pitch: 145,
            bearing: -197.6 });
});
```

Onglets géographiques

```
172     .setLngLat(feature.geometry.coordinates)
173     .setHTML(feature.properties.nom)
174     .addTo(map);
175 });
176
177 map.on('mousemove', function (e) {
178     var features = map.queryRenderedFeatures(e.point, { layers: ['Arrets'] });
179     map.getCanvas().style.cursor = (features.length) ? 'pointer' : '';
180 });
181
182 var nav = new mapboxgl.NavigationControl();
183 map.addControl(nav, 'top-left');
184 map.addControl(new mapboxgl.ScaleControl({
185     maxWidth: 120,
186     unit: 'metric'}));
187
188
189 // Configuration onglets géographiques
190
191 document.getElementById('Gare').addEventListener('click', function ()
192 { map.flyTo({zoom: 16,
193             center: [-1.672, 48.1043],
194             pitch: 145,
195             bearing: -197.6 });
196 });
197
198 </script>
199
```



Onglets géographiques

- Configurer les deux autres onglets géographique (Rennes 2 et Rennes1)

```
9 // Configuration onglets géographiques
10
11 document.getElementById('Gare').addEventListener('click', function () {
12     map.flyTo({zoom: 16,
13     center: [-1.672, 48.1043],
14     pitch: 145,
15     bearing: -197.6
16 });
17 });
18
19 document.getElementById('Rennes1').addEventListener('click', function () {
20     map.flyTo({zoom: 16,
21     center: [-1.6396, 48.1186],
22     pitch: 145,
23     bearing: 197.6
24 });
25 });
26
27 document.getElementById('Rennes2').addEventListener('click', function () {
28     map.flyTo({zoom: 16,
29     center: [-1.7023, 48.1194],
30     pitch: 45,
31     bearing: 50
32 });
33 });
34
35
36 </script>
```

Onglets géographiques

Mapbox_Onglets géographiques ↗
Boris Mericskay

HTML

```
21 <div id='ongletsgeo'>
22
23 * <div class="btn-group-sm" role="group" aria-label="Small
button group">
24
25 * <button type="button" id="Rennes" class="btn btn-outline-
primary">Vue départ</button>
26 * <button type="button" id="Gare" class="btn btn-outline-
primary">Gare SNCF</button>
27 * <button type="button" id="Rennes1" class="btn btn-outline-
primary">Université Rennes 1</button>
28 * <button type="button" id="Rennes2" class="btn btn-outline-
primary">Université Rennes 2</button>
29 </div>
```

CSS

```
1 #map { position: absolute;
2     top: 0;
3     bottom: 0;
4     width: 100%;
5     z-index: 0; }

7 #ongletsgeo {position: absolute;
8     top: 10px;
9     left: 10px;
10    z-index: 1;
11    width: 463px;
12    background-color: #ffffff;
```

JS

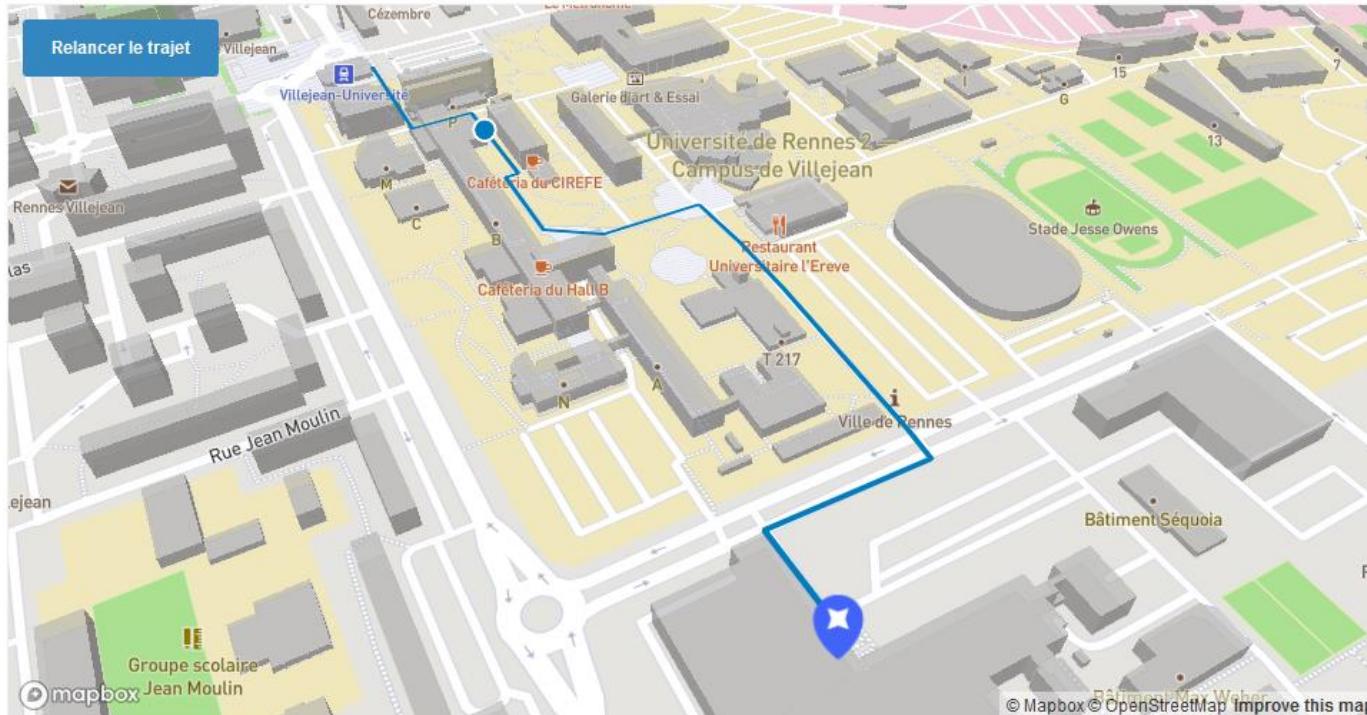
```
27 center: [-1.672, 48.104], 
28 pitch: 0,
29 bearing:0 });
30 });

32 document.getElementById('Gare').addEventListener('click',
function ()
33 { map.flyTo({zoom: 16,
34 center: [-1.672, 48.104],
35 pitch: 20,
36 bearing: -197.6 });
37 });
38
39
```

<https://codepen.io/BorisMericskay/pen/GRXNQQv>

Exemple itinéraire

#MapboxGL / Carte itinéraire



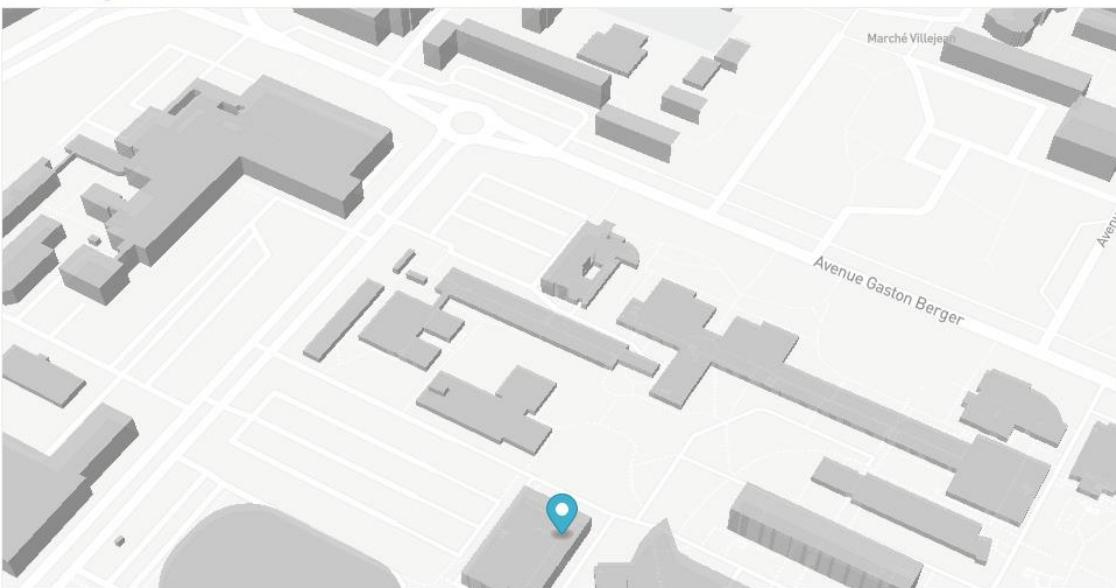
Built with [blockbuilder.org](#)

[Open](#)

<https://codepen.io/BorisMericskay/pen/dyqOmYz>

Animation

#MapboxGL / Animation 2



<https://codepen.io/BorisMericskay/pen/dyjQRLm>

#MapboxGL / Animation carte



<https://codepen.io/BorisMericskay/pen/MWBzoMy>

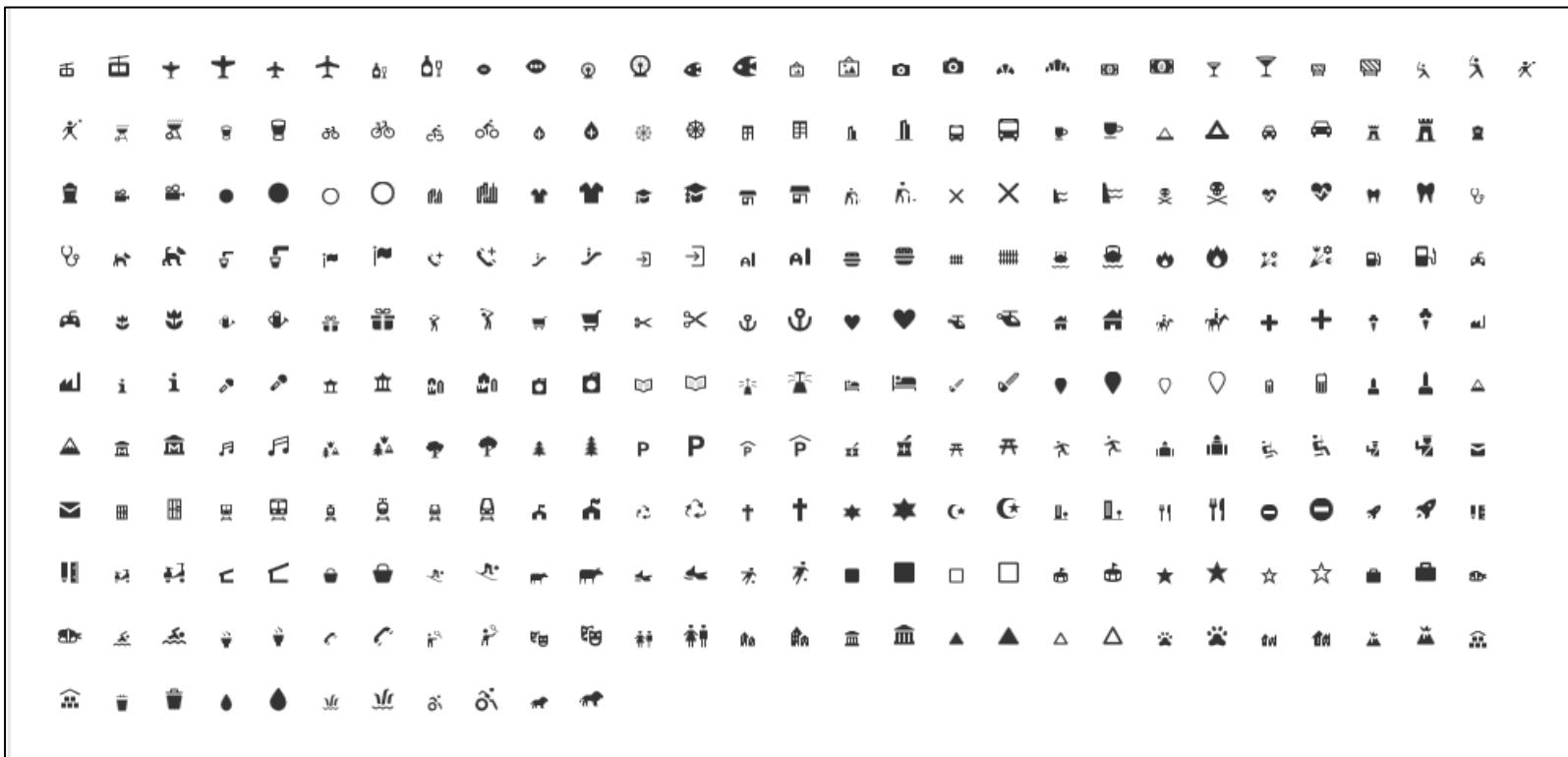
Mise en forme poussée de données spatiales

Pictogrammes, catégorisation, graduation, cercles gradués, extrusion 3D, combinaison de deux variables, cluster, heatmap,...

Mise en forme de données personnelles

- Mobiliser la bibliothèque vectorielle Maki

<https://www.mapbox.com/maki-icons/>



Utiliser des pictogrammes

- Utiliser la symbologie des symboles (pictos)

```
map.addSource('Arrets', {  
    type: 'vector',  
    url: 'mapbox://ninanoun.7mtp5buo'});  
  
map.addLayer({  
    "id": "Arrets",  
    "type": "symbol",  
    "source": "Arrets",  
    "source-layer": "topologie-des-points-darret-d-9ya955",  
    "layout": { "icon-image": "bus-15",  
               "icon-size": 1.5}  
});
```

Utiliser des pictogrammes

Mapbox_IconsMaki ↗
Boris Mericskay

HTML

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <meta charset='utf-8' />
5     <title>MapboxGL</title>
6
7 <script src='https://api.tiles.mapbox.com/mapbox-gl-js/v1.5.1/mapbox-gl.js'></script>
8 <link href='https://api.tiles.mapbox.com/mapbox-gl-js/v1.5.1/mapbox-gl.css' rel='stylesheet'
/>
9
10 </head>
11
12 <body>
13 <div id='map'> </div>
```

JS

```
18 // Ajout des arrêts de bus
19
20 map.addSource('Arrets', {
21     type: 'vector',
22     url: 'mapbox://ninanoun.58widelk'});
23
24 map.addLayer({
25     'id': 'Arrets',
26     'type': 'symbol',
27     'source': 'Arrets',
28     'source-layer': 'Bus-5ypx1k',
29     'layout': {'icon-image': 'bus-15', 'icon-size': 1.5},
30     minzoom:12
31});
```

<https://codepen.io/BorisMericskay/pen/OJobvWa>

Graduation couleur

- Configurer les options de mise en forme
 - Il faut spécifier la **variable mobilisée**, le **type d'échelle**, les **valeurs des bornes** et les **couleurs**

Cercles

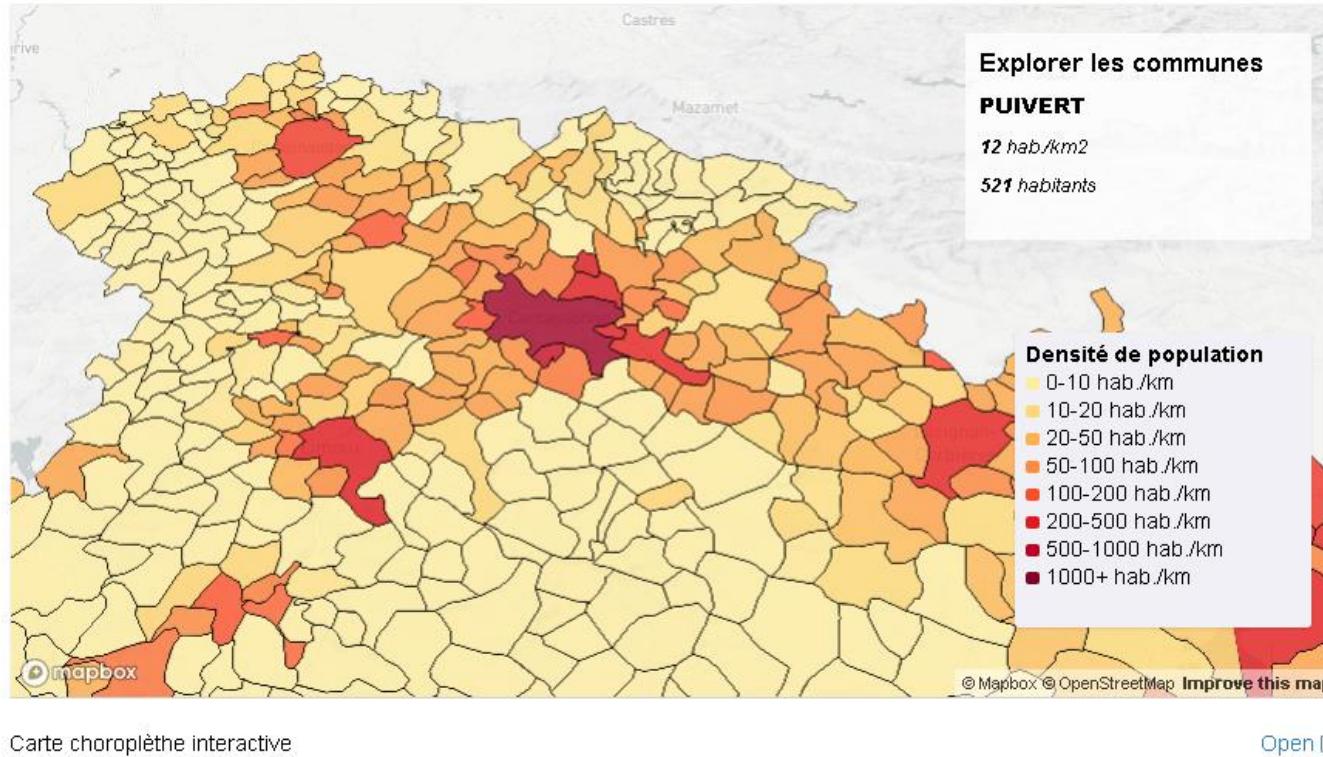
```
'paint': {'circle-radius': {'base': 1.5, 'stops': [[13, 2], [22, 60]]},  
          'circle-color': {'property': 'hauteur',  
                          'type': 'exponential',  
                          'stops': [[0, '#edf8e9'],  
                                    [5, '#c7e9c0'],  
                                    [10, '#a1d99b'],  
                                    [15, '#74c476'],  
                                    [20, '#006d2c']]}}}
```

Polygones

```
'paint': {'fill-color': {'property': 'densite',  
                         'stops': [[1, '#1a9850'],  
                                   [10, '#91cf60'],  
                                   [20, '#d9ef8b'],  
                                   [50, '#ffffbf'],  
                                   [100, '#fee08b'],  
                                   [150, '#fc8d59'],  
                                   [200, '#d73027']]},  
           'fill-opacity': 0.9}
```

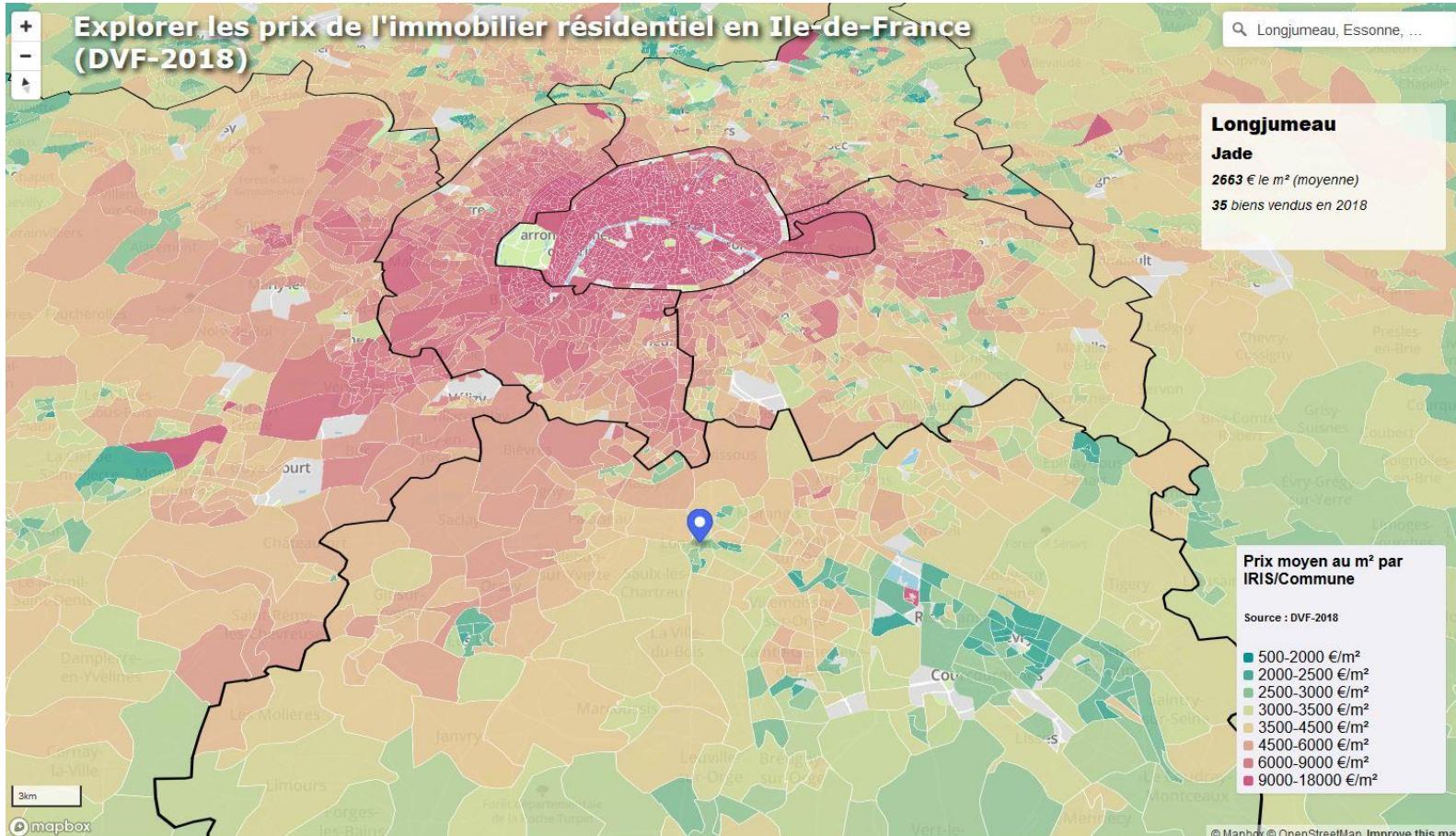
Exemple

#MapboxGL / Carte choroplète interactive



<https://codepen.io/BorisMericskay/pen/abjQydV>

Exemple



https://www.sites.univ-rennes2.fr/mastersigat/Webmapping/DVF_IDF.html

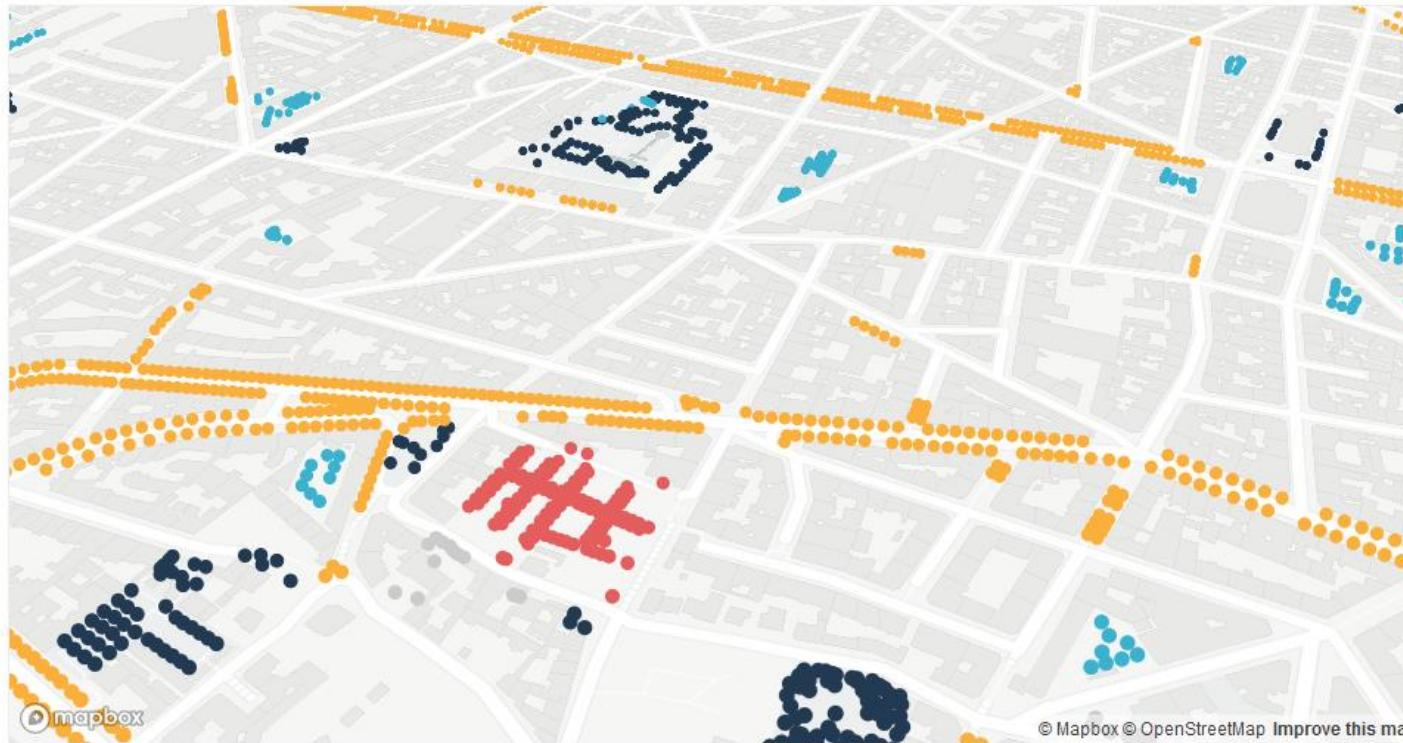
Catégorisation

- Configurer les options de mise en forme
 - Il faut spécifier la **variable mobiliser**, les valeurs des **catégories** et les **couleurs**

```
'paint': { 'circle-radius': {'base': 1.75,'stops': [[12, 2], [22, 180]]},  
          'circle-color': [  
              'match',[ 'get', 'ethnicity'],  
              'White', '#fbb03b',  
              'Black', '#223b53',  
              'Hispanic', '#e55e5e',  
              'Asian', '#3bb2d0',  
              '#ccc'  
          ]  
      }
```

Exemple

#MapboxGL / Catégorisation de données



Built with [blockbuilder.org](#)

[Open](#)

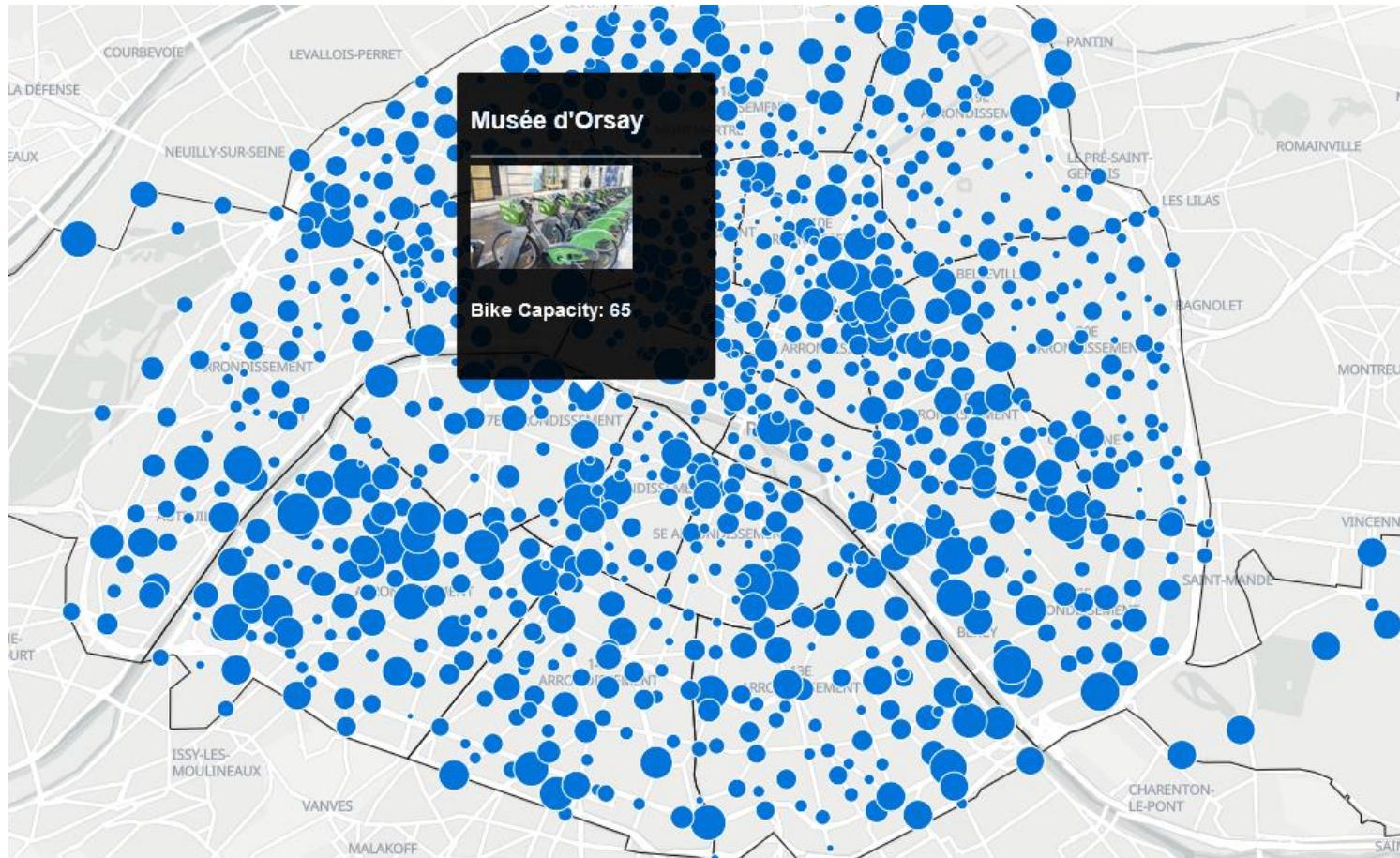
<https://codepen.io/BorisMericskay/pen/VwBqKjx>

Cercles gradués

- Configurer les options de mise en forme
 - Il faut spécifier la **variable mobiliser** et les valeurs des **bornes (valeur, taille du cercle)**

```
paint: {'circle-color': '#D49A66',
        'circle-radius': {property: 'population',
                         type: 'exponential',
                         stops: [[10, 1],[2000, 20]]},
        'circle-opacity': 0.8}
```

Exemple



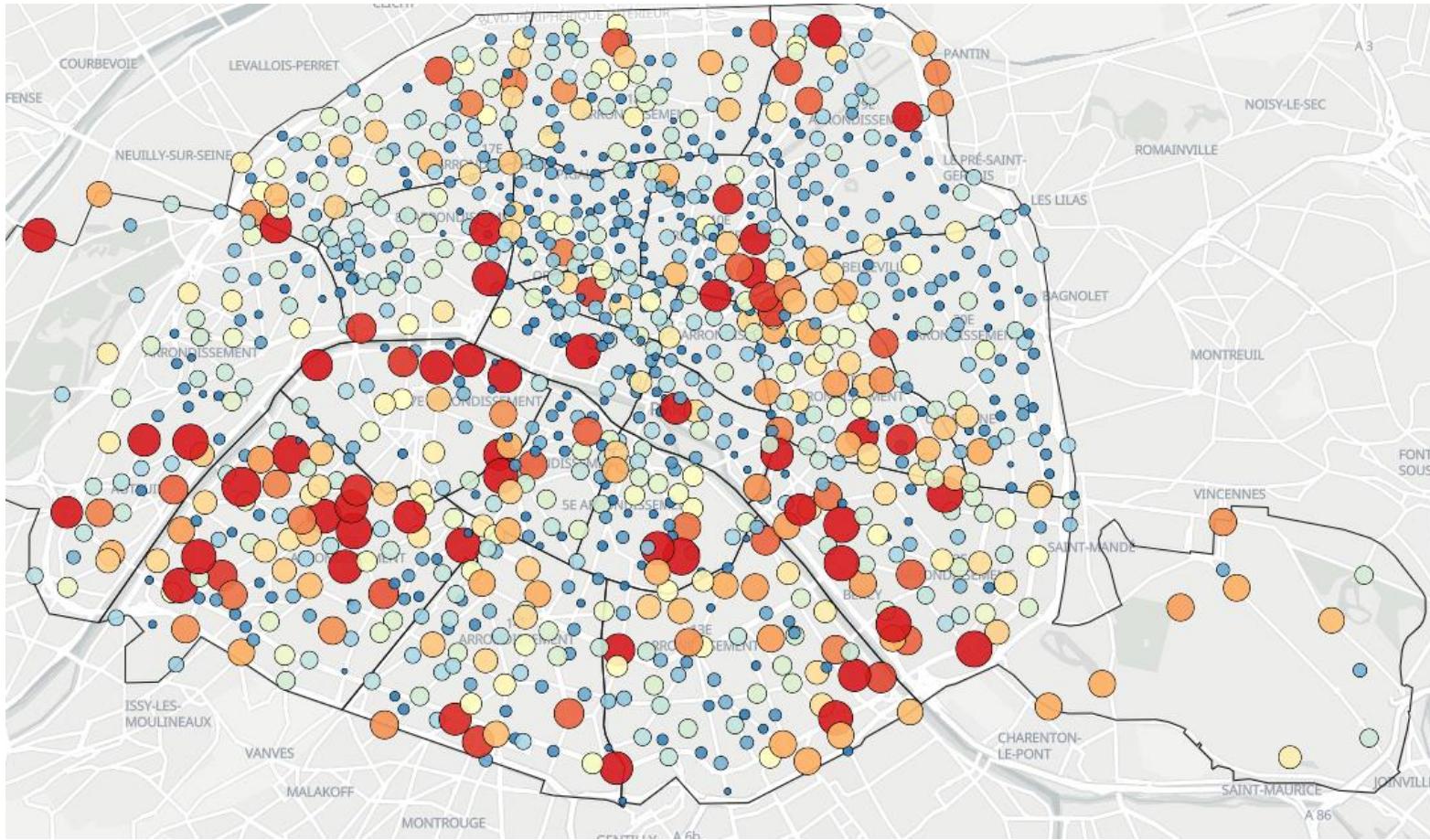
<https://codepen.io/BorisMericskay/pen/MWBZbVX>

Cercles gradués avec graduation de couleur

- Configurer les options de mise en forme
 - Il faut spécifier la **variable mobiliser** et les valeurs des **bornes (valeur, taille du cercle)**

```
'paint': {'circle-stroke-color': '#000000',
          'circle-stroke-width': 0.6,
          'circle-radius': {property: 'capacity', type: 'exponential', stops:
[[0, 0],[100, 20]]},
          'circle-color': {property: 'capacity',stops: [
[20, '#2c7bb6'],
[30, '#abd9e9'],
[40, '#ffffbf'],
[50, '#fdbe61'],
[60, '#d7191c'],]}, 
          'circle-opacity':0.9}}
);
```

Exemple

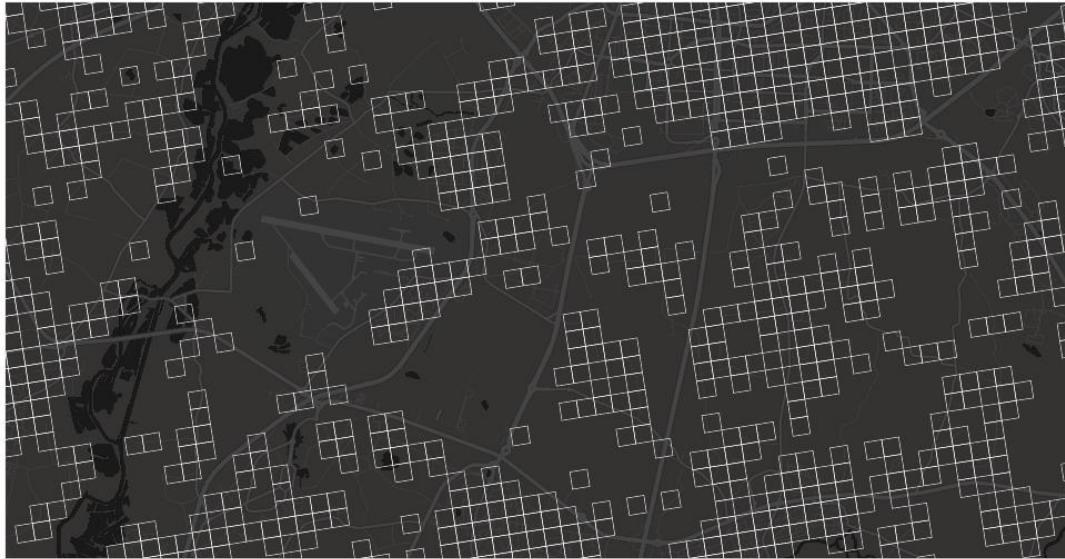


<https://codepen.io/BorisMericskay/pen/MWBZbRP>

Extrusion 3D

- Récupérer le template

#Template / Extrusion MapboxGL



Built with [blockbuilder.org](#)

[Open](#)

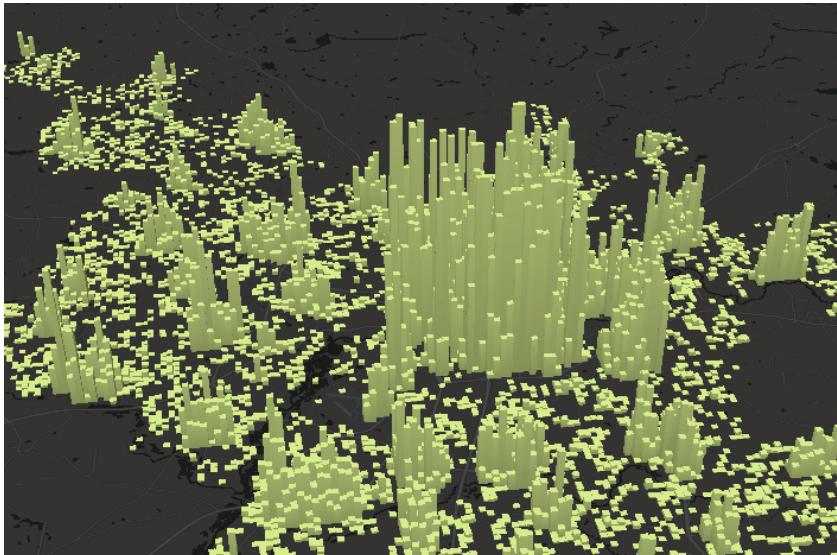
index.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset='utf-8' />
<title>Display a map</title>
<meta name='viewport' content='initial-scale=1, maximum-scale=1, user-scalable=no' />
```

<https://bl.ocks.org/mastersigat/64af1a273f155037214d96406cb4777a/7c3f5c36742ec2dcf231059e903b144118c79d42>

Extrusion 3D

- Configurer les options de mise en forme
 - Il faut spécifier la **variable mobilisée** et les modalités de **l'extrusion (valeur, taille de l'extrusion)**



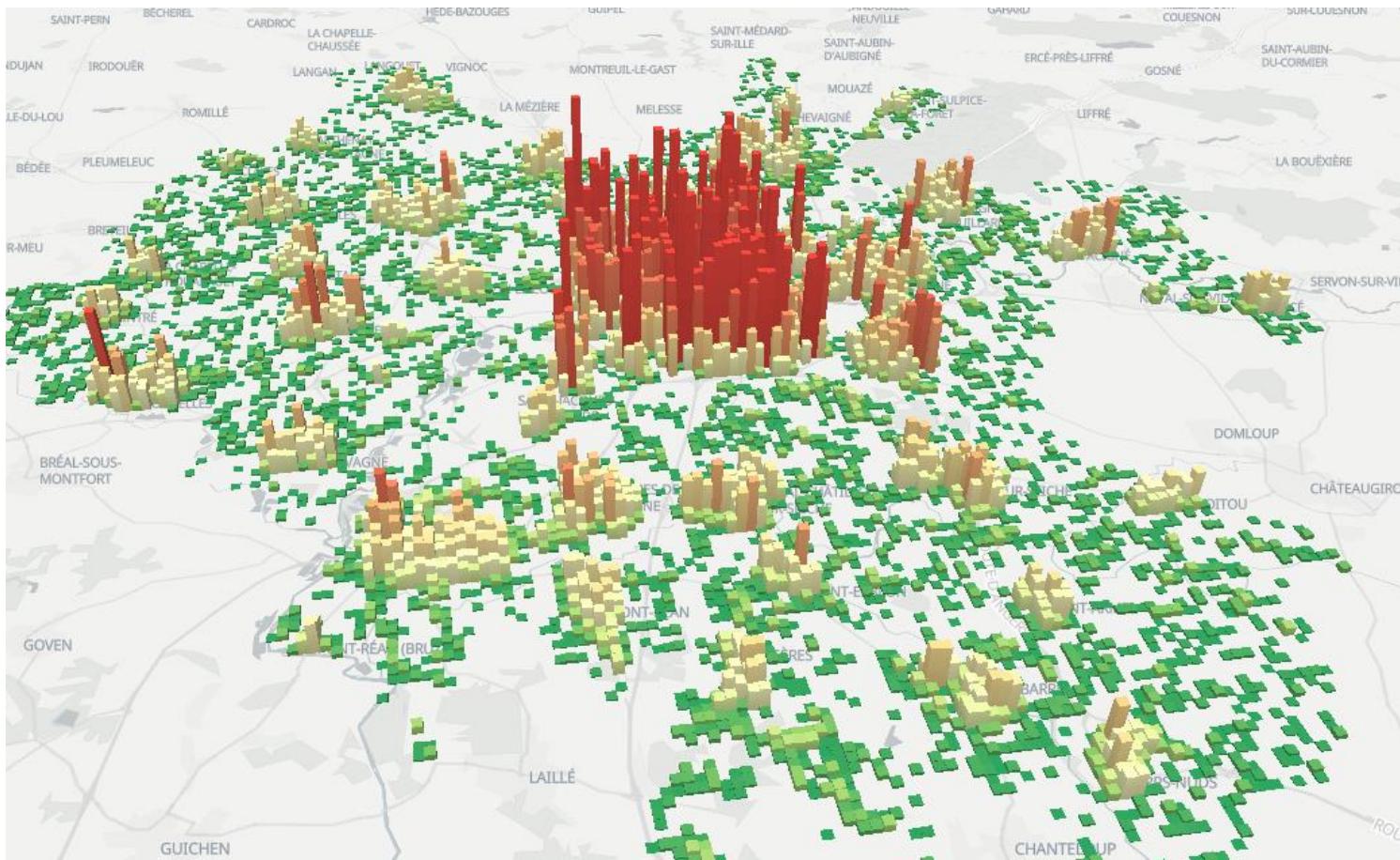
```
map.addLayer({
  'id': 'extrude',
  'type': 'fill-extrusion',
  'source': 'Carro',
  'source-layer': 'karook-dcnhdj',
  'layout': {'visibility': 'visible'},
  'paint': { 'fill-extrusion-color': '#d9ef8b',
    'fill-extrusion-height': {
      'property': 'Individus',
      'stops': [[1, 0],
                [10, 100],
                [700, 7000]]},
    'fill-extrusion-opacity': 0.95,
    'fill-extrusion-base': 0 }
});
```

Extrusion 3D

- Ajouter des couleurs

```
'paint': {
  'fill-extrusion-color': {
    'property': 'Individus',
    'stops': [
      [1, '#1a9850'],
      [10, '#91cf60'],
      [20, '#d9ef8b'],
      [50, '#ffffbf'],
      [100, '#fee08b'],
      [150, '#fc8d59'],
      [200, '#d73027']]
  },
  'fill-extrusion-height': {
    'property': 'Individus',
    'stops': [[1, 0],
              [10, 100],
              ...]
  }
}
```

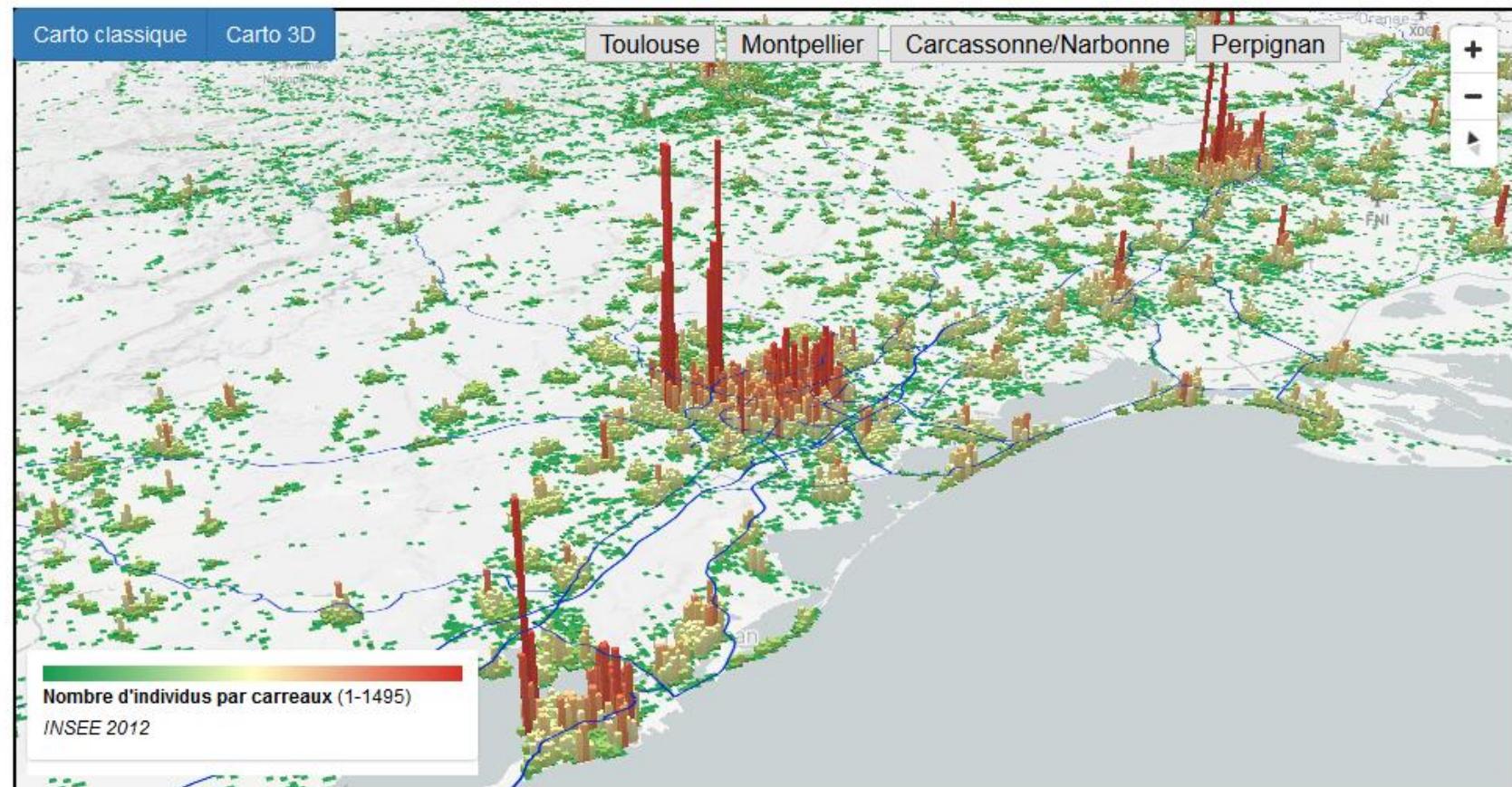
Exemple



<https://codepen.io/BorisMericskay/pen/wvxRgjE>

Exemple

#MapboxGL / Extrusion carreaux



Built with [blockbuilder.org](#)

[Open](#)

https://sites-formations.univ-rennes2.fr/mastersigat/WebMaps/Extrusion_Occitanie.html

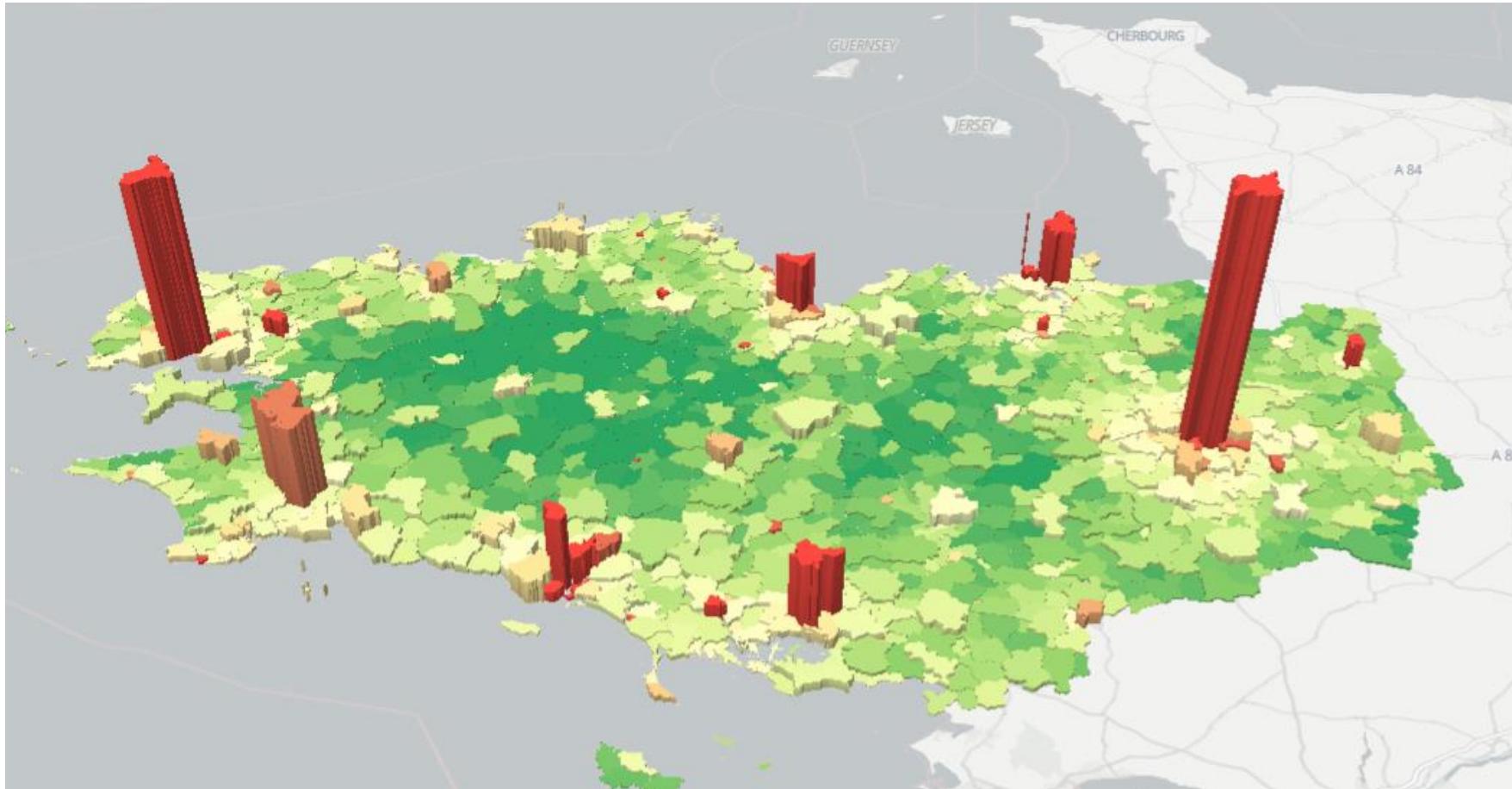
Deux variables

```
map.addLayer({
  'id': 'extrudecommunes',
  'type': 'fill-extrusion',
  'source': 'communes',
  'source-layer': 'TD1_Data-3kid81',
  'layout': {'visibility': 'visible'},
  'paint': {
    'fill-extrusion-color': {
      'property': 'densite',
      'stops': [[20, '#1a9850'],
                 [50, '#91cf60'],
                 [100, '#d9ef8b'],
                 [200, '#ffffbf'],
                 [500, '#fee08b'],
                 [1000, '#d73027']]],
      'fill-extrusion-height': {'property': 'popOK',
                                'stops': [[100, 10],
                                          [1000, 100],
                                          [200000, 100000]]},
      'fill-extrusion-opacity': 0.8,
      'fill-extrusion-base': 0
    }
});
```

Symbologie graduation de couleur
(densité)

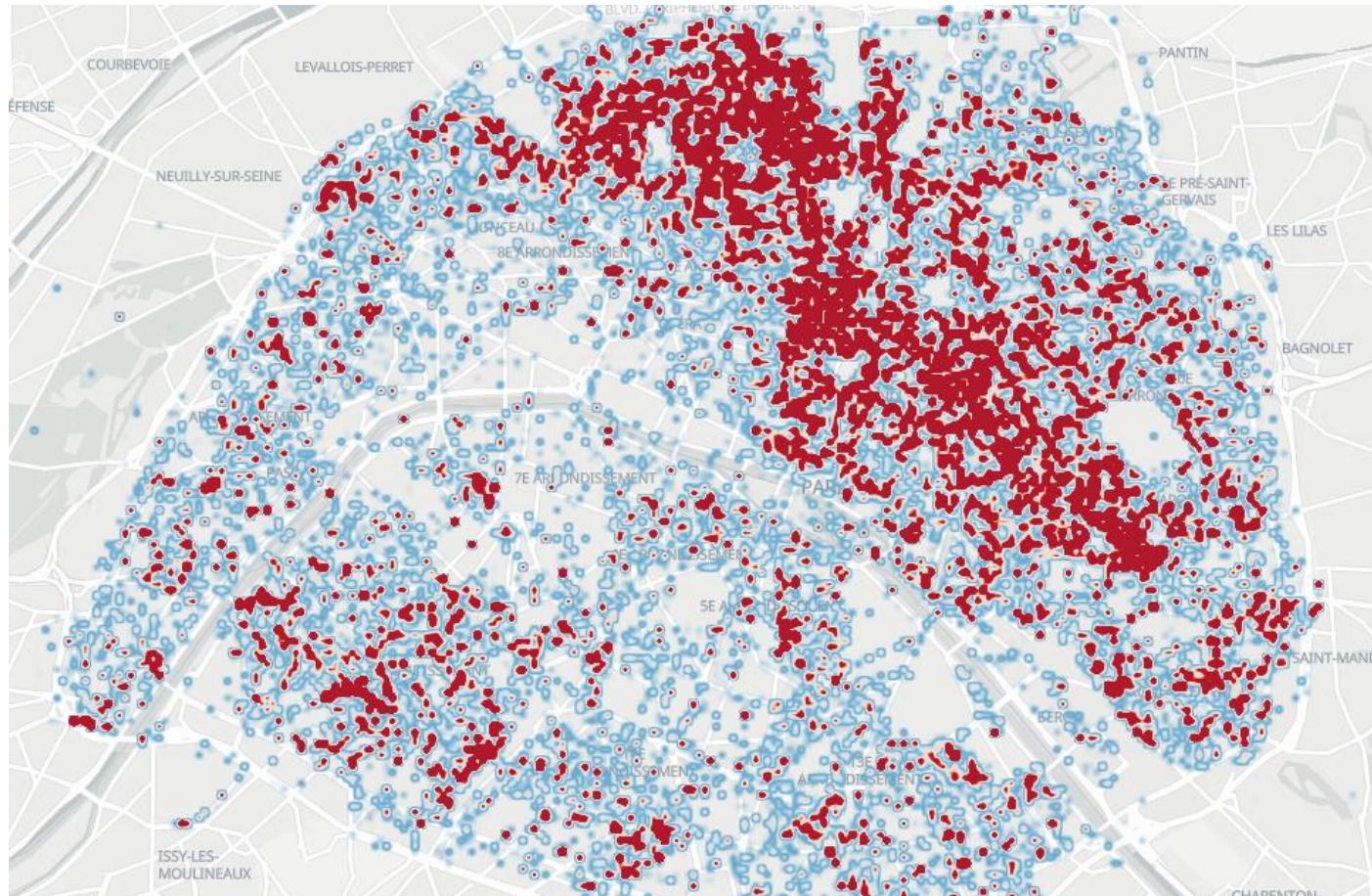
Symbologie extrusion 3D de la population

Exemple



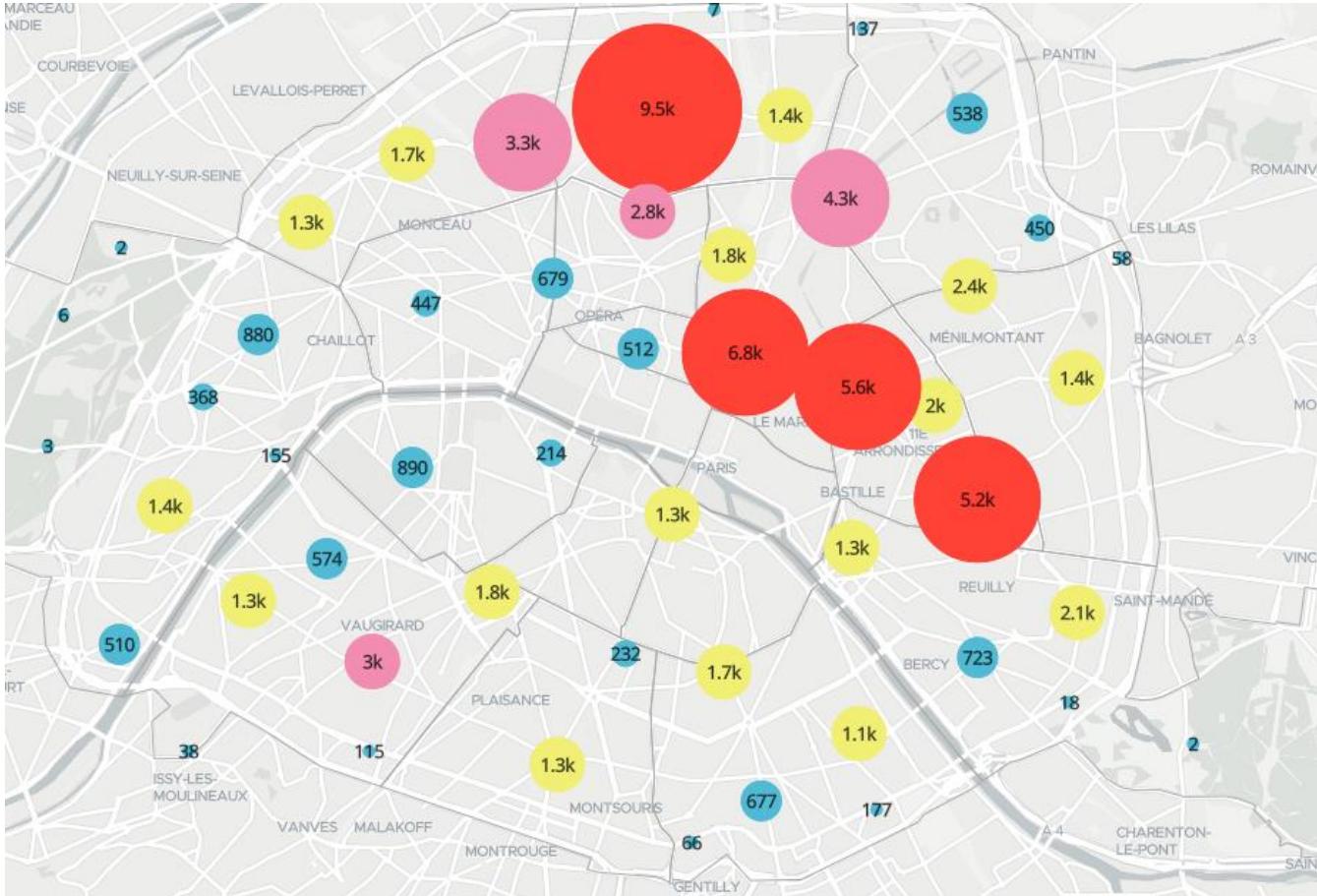
<https://codepen.io/BorisMericskay/pen/eYLBMjB>

Heatmap



<https://codepen.io/BorisMericskay/pen/gOjZmYB>

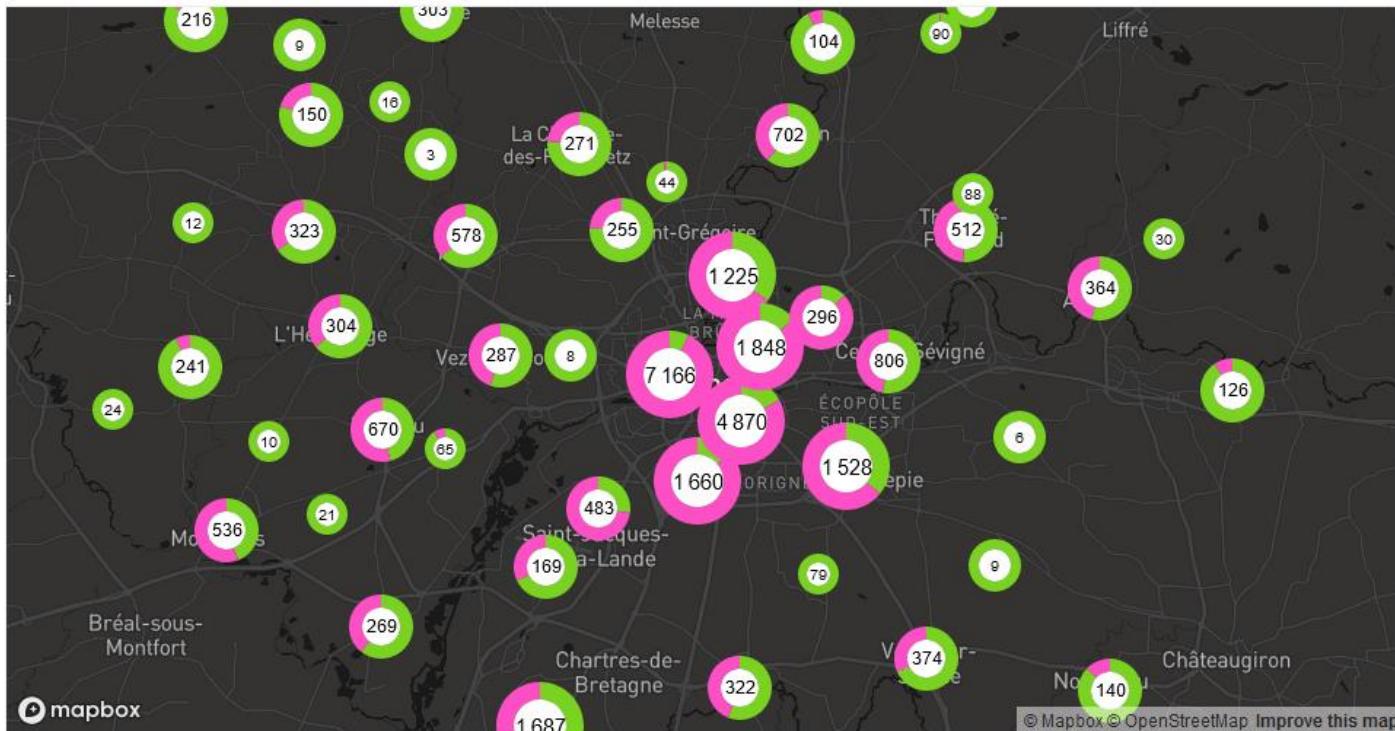
Cluster



<https://codepen.io/BorisMericskay/pen/OJwrpXq>

Cluster thématique

#MapboxGL / Clusters thématiques



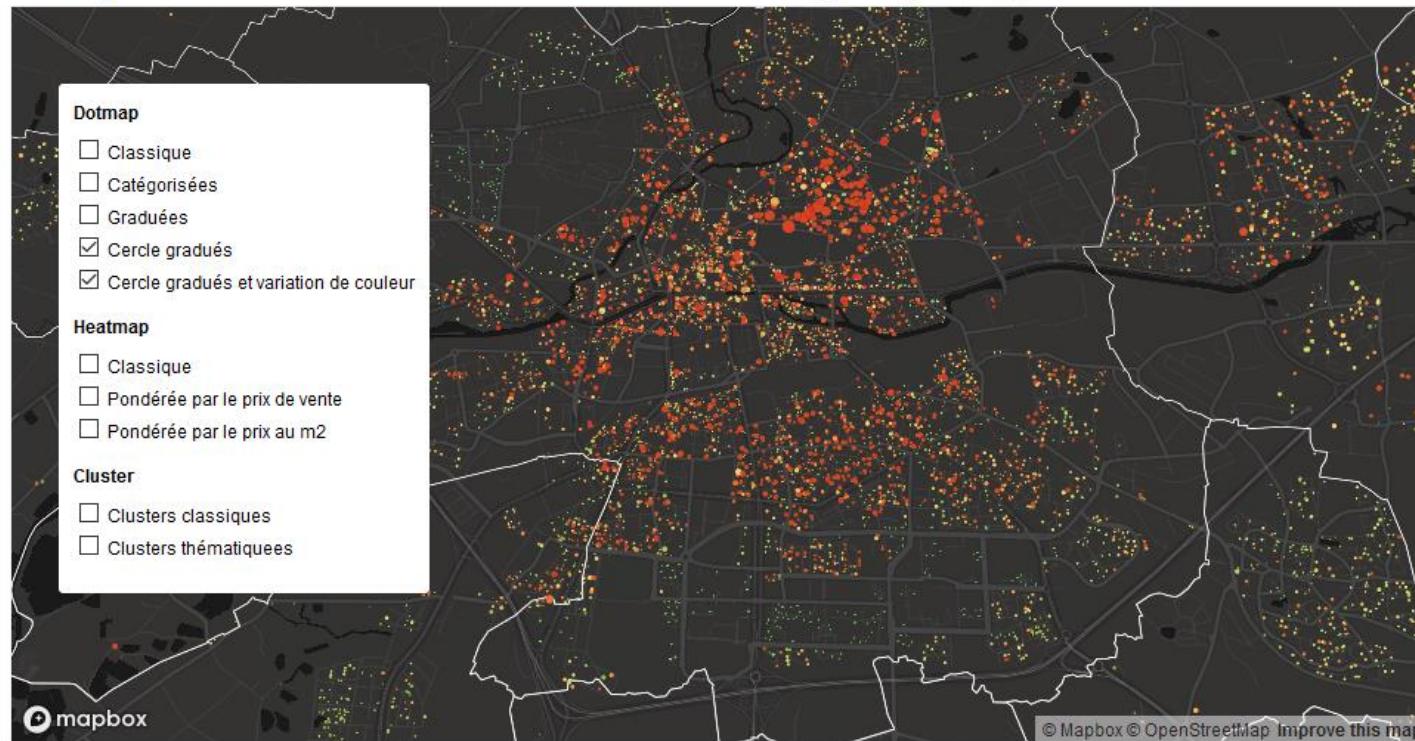
Built with [blockbuilder.org](#)

[Open](#)

<https://codepen.io/BorisMericskay/pen/qByLrp>

Synthèse des modes de visualisation

MapboxGL / Variation Cartographique



Built with [blockbuilder.org](#)

[Open](#)

https://sites-formations.univ-rennes2.fr/mastersigat/WebMaps/Variation_Mapbox.html

MNT en 3D

- Depuis la version 2.x il est possible de mobiliser un MNT mondial et de l'extruder en 3D + texture du ciel + caméra (animation)

Examples

Found 5 results. [Reset filters](#)



Add 3D terrain to a map
Use `setTerrain` to add 3D terrain to a map using a raster terrain source.



Add a gradient sky layer to a map
Add a customizable, gradient sky layer.



Add an atmospheric sky layer to a map
Add a customizable sky layer that simulates the natural scattering of light in the atmosphere.



Animate the camera along a path
Use the `FreeCamera` API to follow a path over 3D terrain.



Animate the camera around a point in 3D terrain
Use the `FreeCamera` API to create a fly-over animation focused on a point.

<https://docs.mapbox.com/mapbox-gl-js/example/?topic=3D>

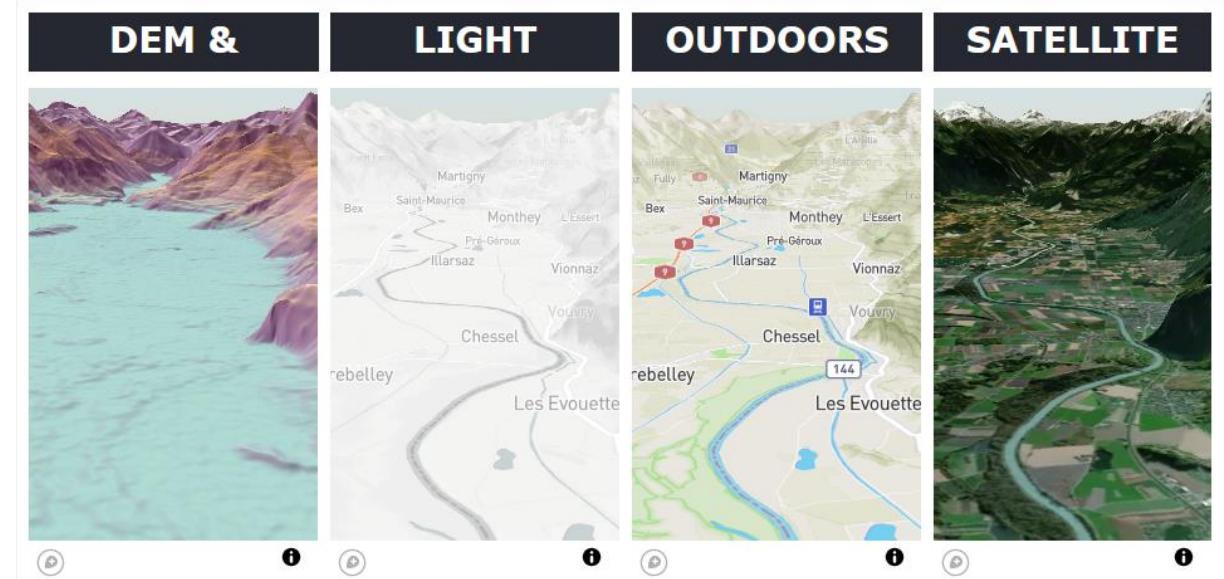
MNT en 3D

#MapboxGL / DEM 3D (MapboxGL 2.x)



<https://codepen.io/BorisMericskay/pen/vYaVger>

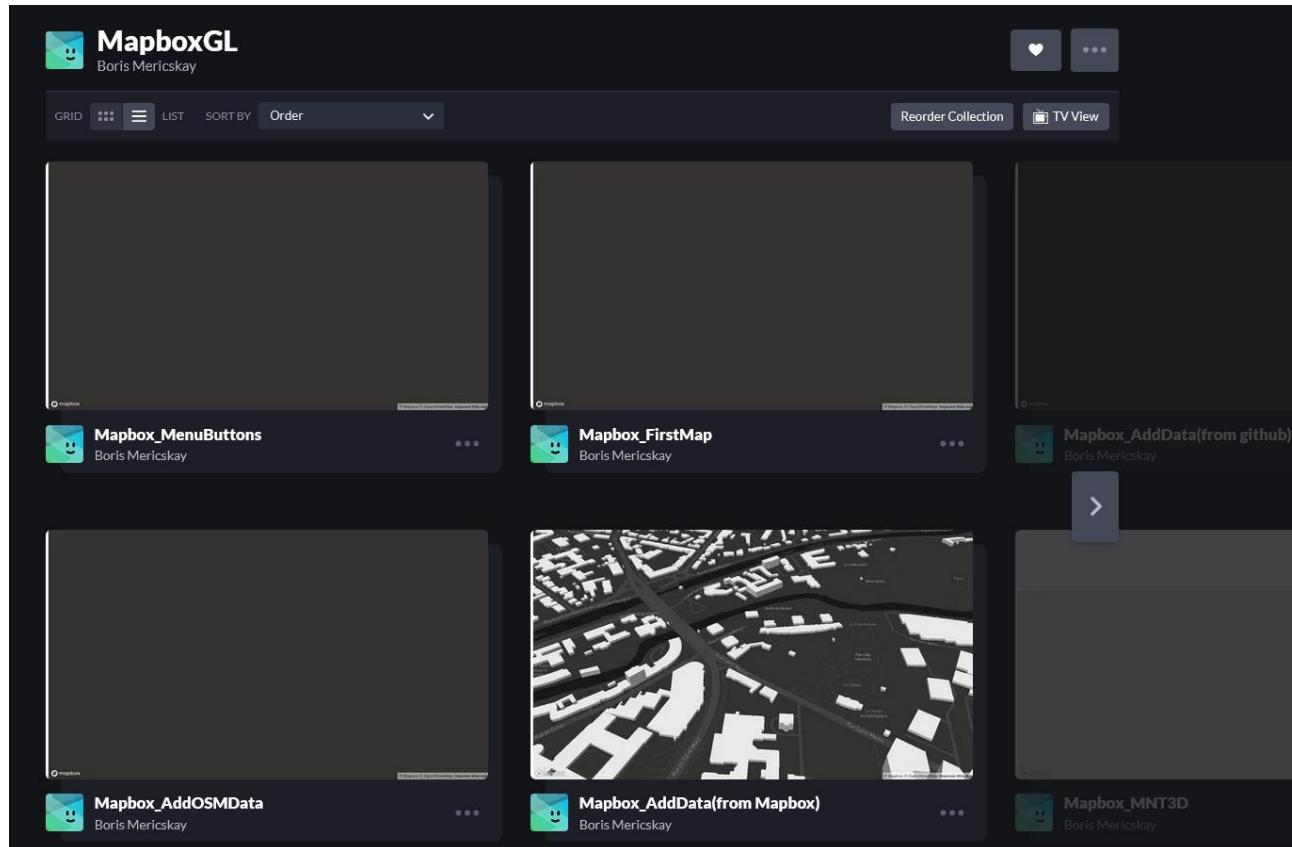
MapboxGL / Sync 3D maps



<https://codepen.io/BorisMericskay/pen/xxJyqjX>

Exemples divers

- <https://codepen.io/collection/wayZNV>



Exemples Mapbox

[mapbox | Docs](#)

All docs > Mapbox GL JS > Examples

Examples

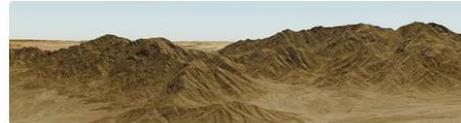
Display a map

Initialize a map in an HTML element with Mapbox GL JS.



Add 3D terrain to a map

Use `setTerrain` to add 3D terrain to a map using a raster terrain source.



Add a canvas source

Add a canvas source to the map.



Accept coordinates as input to a geocoder

Use the `mapbox-gl-geocoder` control to search for places using Mapbox Geocoding API.



Add a 3D model

Use a custom style layer with `three.js` to add a 3D model to the map.



Add a custom style layer

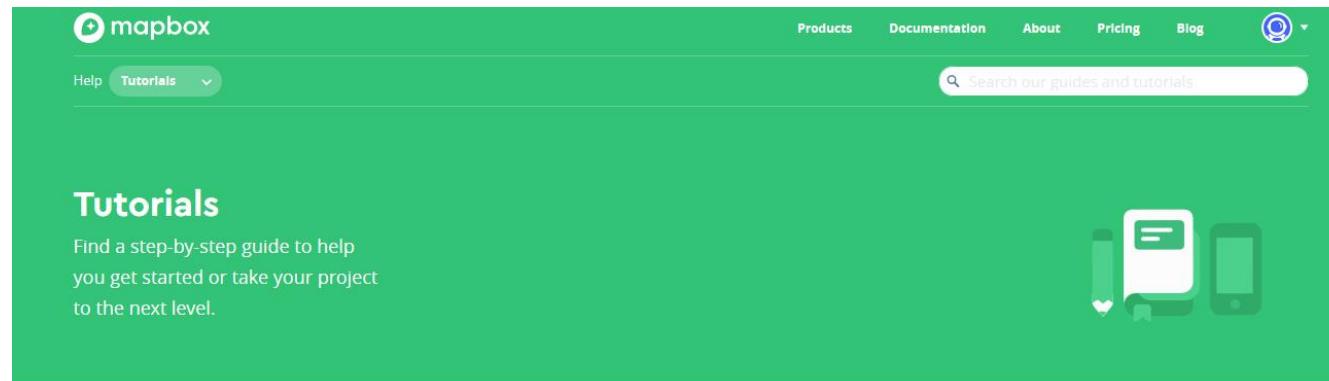
Use a custom style layer to render custom WebGL content.



<https://docs.mapbox.com/mapbox-gl-js/example/>

Tutoriels

<https://docs.mapbox.com/help/tutorials/?product=Mapbox+GL+JS>



Categories

All

Web apps (18)

Map design (13)

Uploads (12)

Mobile apps (11)

Third party integration (7)

Data (5)

Unity (4)

Directions (4)

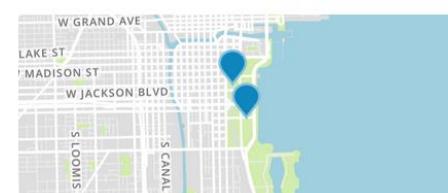
Analysis (3)

Satellite (2)

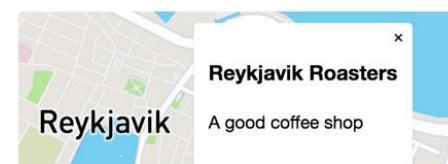
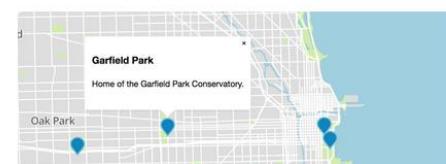
Web apps (18)



Make a choropleth map, Part 1: create a style

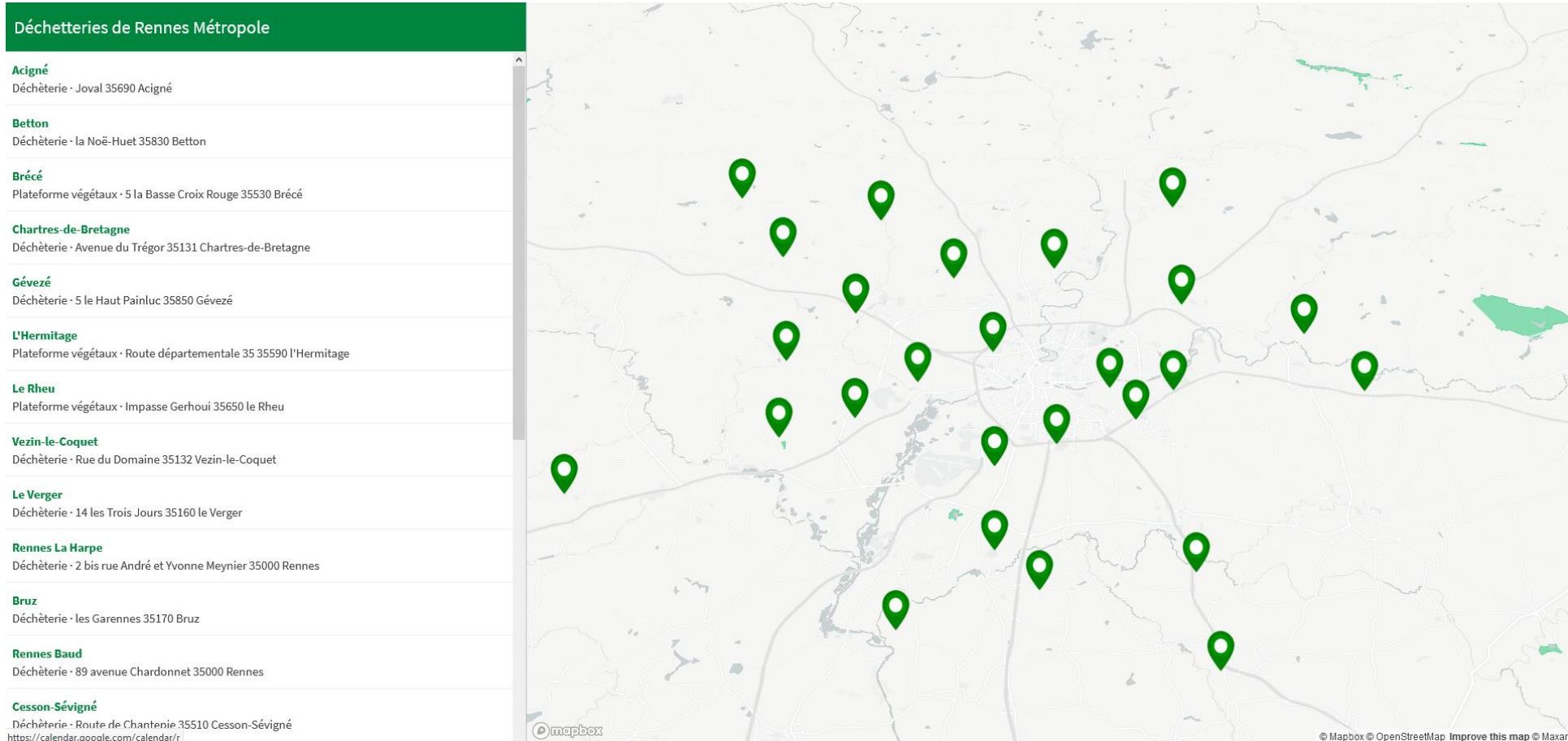


Add custom icons or markers



Aller plus loin

<https://docs.mapbox.com/help/tutorials/building-a-store-locator/>



<https://codepen.io/BorisMericskay/pen/dyqVoBQ>