



TPx Leaflet&OpenStreetMap

L'alternative open source à Google Maps®



Durée 4h

Indispensable d'un site web aujourd'hui l'affichage des cartes géographiques !
Google Maps® devenant de plus en plus restrictif et payant après seulement quelques requêtes, comment faire ??

Prenons en main l'alternative open source, Leaflet associé avec les fonds de cartes OpenStreetMap.

1. Quelques mots sur Leaflet et OpenStreetMap

1.1 OpenStreetMap

OpenStreetMap (OSM pour les intimes) est un fournisseur de fonds de carte¹. C'est exactement le même métier que ce que font les grands du web comme Google Maps, Bing Maps ou encore des acteurs plus locaux, comme IGN.

OSM se distingue cependant de ces différents organismes par sa démarche. En effet, toutes les cartes sont construites de manière collaborative et sont placées sous licence libre. En 2013 on recensait plus d'un million de personnes ayant contribué à l'élaboration des cartes que ce soit en répertoriant des chemins lors de leur promenade ou en corrigeant/ajoutant des noms de rues et bâtiments, par exemple.

Si vous voulez essayer, rendez-vous sur <http://www.openstreetmap.org/>



Quelques projets utilisant OSM

- Le site du même nom (la version *.fr parle de l'organisation, les versions *.com et *.org exposent quant à elles les cartes). Ce site vous permet d'explorer l'environnement et d'établir des itinéraires.
- De nombreux outils dédiés aux loisirs se servent aussi des cartes OSM, comme par exemple OpenSeaMap pour la navigation ou OpenCycleMap plutôt orienté cyclisme.
- Enfin, de véritables projets citoyens ont aussi vu le jour comme WheelMap qui recense les lieux étant considérés comme accessibles en fauteuil roulant.

1.2 Leaflet

OSM c'est bien, mais ce n'est pas forcément trivial à intégrer sur un site. Il faut en effet gérer le système de tuiles² pour afficher la carte et on ne peut pas vraiment interagir avec.

Afin de palier à cela, une bibliothèque javascript a vu le jour : « **Leaflet** ». Bibliothèque [open-source](#) et *customisable* via un mécanisme de plugin.



TPx Leaflet&OpenStreetMap

L'alternative open source à Google Maps®



Leaflet va donc se charger de faire le chargement des tuiles de la carte en fonction du lieu que l'utilisateur veut visualiser (et à quel niveau de zoom) mais aussi gérer des interactions comme l'affichage de marqueur, de zone, gérer les déplacements, etc.

Leaflet est indépendant du fournisseur de cartes, la bibliothèque peut tout à fait fonctionner aussi avec Google Maps® ou d'autres organismes.



1. Un fond de carte est une représentation planaire graphique d'un environnement géographique. Bref, c'est une carte qui n'affiche pas forcément des routes ou des reliefs.
2. On ne charge jamais une carte en entier, cela demanderait bien trop d'informations et générerait un trafic trop important. À la place, on charge la carte "par morceau" de plusieurs [km²] concernant uniquement la zone à visualiser.

2. Notre première carte

2.1 Structure de base (lien avec l'API Leaflet)

Pour cela, la première étape sera d'ajouter la bibliothèque Leaflet dans votre page web.

Deux solutions :

- Utiliser une version hébergée sur le cloud

```
<link rel="stylesheet" href="https://unpkg.com/leaflet@1.8.0/dist/leaflet.css"
integrity="sha512-
hoalWLoI8r4UszCkZ5kL8vayOGVae1oxXe/2A4AO6J9+580uKHDO3JdHb7NzwwzK5xr/
Fs0W40kiNHxM9vyTtQ==" crossorigin="" />
<script src="https://unpkg.com/leaflet@1.8.0/dist/leaflet.js" integrity="sha512-
BB3hKbKWOC9Ez/TAwyWxNXeoV9c1v6FieYiBieIWkpLjauysF18NzgR1MBNBXf8/KA
BdlkX68nAhlwcDFLGPCQ==" crossorigin=""></script>
```

- Utiliser une version hébergée localement

```
<link rel="stylesheet" href="/path/to/leaflet.css" />
<script src="/path/to/leaflet.js"></script>
```

Il faudra aussi ajouter un peu de CSS spécifique via la balise **<link>** suivante à mettre dans le **<head>** de votre page :

```
<link rel="stylesheet" href="style.css">
```



TPx Leaflet&OpenStreetMap

L'alternative open source à Google Maps®



① Vous avez sûrement remarqué qu'un numéro de version dans le lien. Vérifier que vous utilisez bien la dernière version de Leaflet en allant sur [leur page "Downloads"](#).

Une fois que ces deux éléments sont prêts, il ne reste plus qu'à rajouter une balise `<div>` dans la section `<body>` de votre page. Cette dernière servira évidemment à accueillir la carte. Afin de la retrouver facilement, on lui donne un id reconnaissable comme **#macarte**. Spécifier la taille de la carte avec les balises `style`.

```
<div id="macarte"style="width:545px; height:490px">></div>
```

Bon, c'est pas mal, mais ça n'affiche rien ! Passons à la vitesse supérieure.

2.2 Afficher la carte

Ajouter un bloc script à la fin de la page, avant la balise `</body>`. Ce morceau de script va faire le lien entre la balise **`<div id="macarte">`** créée plus tôt et le code javascript.

Créer une variable **carte** qui va contenir un objet **map** de la bibliothèque Leaflet.

```
var carte = L.map('macarte').setView([46.3630104, 2.9846608], 6);
```

Pour centrer la carte sur la France **[46.3630104, 2.9846608]**

Quel est le rôle du dernier paramètre de `setview()` ?

Maintenant qu'on a un *moteur* de carte de prêt, on va lui rajouter un fond de carte OSM.

```
L.tileLayer('http://{s}.tile.osm.org/{z}/{x}/{y}.png', {  
  attribution: '&copy; <a href="http://osm.org/copyright">OpenStreetMap</a> contributors'  
}).addTo(carte);
```

La première partie désigne le fournisseur de cartes et place des *marqueurs de templates* dans son adresse. Ensuite, on ajoute ce fournisseur de carte à notre variable carte créée juste avant.

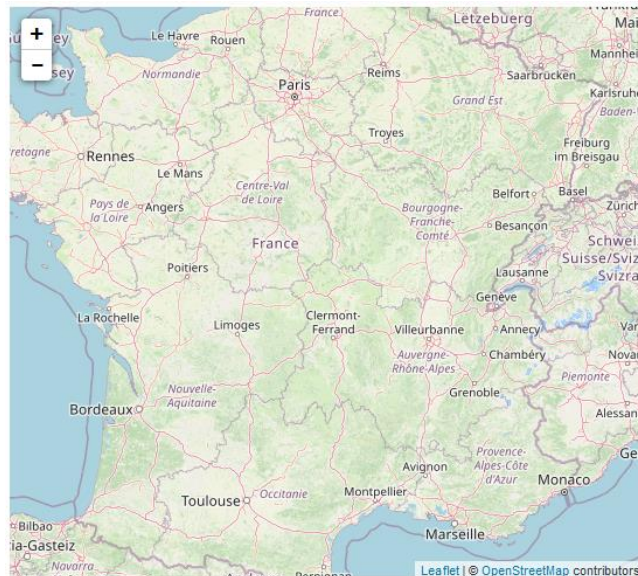
① Sans rentrer dans les détails, les marqueurs de template servent à décrire quel est le format de l'adresse à utiliser pour charger une tuile particulière. Ainsi, les infos `z`, `x`, `y` servent à décrire respectivement le niveau de zoom voulu ainsi que par exemple, la position en `x` et `y` voulue.

Si tout va bien, voici ce que vous devez maintenant obtenir :



TPx Leaflet & OpenStreetMap

L'alternative open source à Google Maps®



2.3 Rajoutons de l'informations

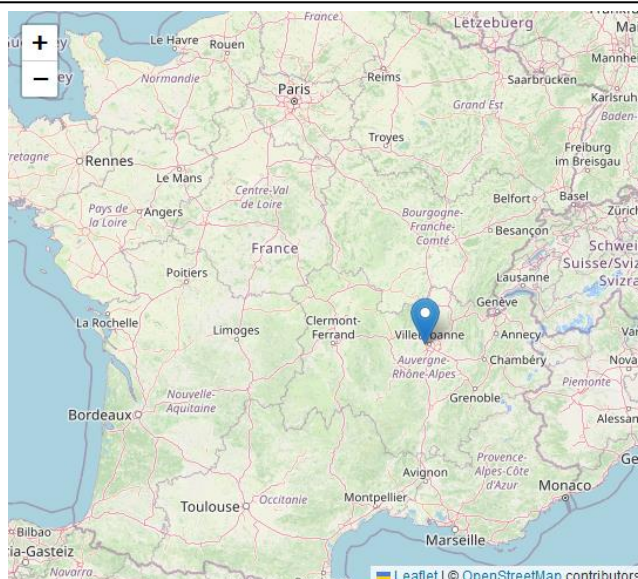
2.3.1 Des marqueurs

Le B-A-BA de la cartographie est bien souvent de savoir placer un marqueur sur une carte pour indiquer une position.

Un marqueur ([marker](#)) est défini par une position en latitude et en longitude. Pour l'exemple, placer un sur « le nombril du monde »... se situant aux coordonnées **[45.75336550211056, 4.791206030681179]**.

Pour placer un marqueur, dans le script, on va donc commencer par créer une variable de type **marker** en lui donnant des coordonnées, puis l'ajouter à la carte.

```
var marker = L.marker([45.75336550211056, 4.791206030681179]).addTo(carte);
```





TPx Leaflet&OpenStreetMap

L'alternative open source à
Google Maps®



2.3.2 Des lignes

Pour dessiner un chemin, on va créer un **Polyline**. Un **Polyline** va avoir besoin d'une succession de position à relier puis d'**options** comme sa couleur, son opacité, sa largeur...

On va donc faire un chemin qui relie les coordonnées suivantes :

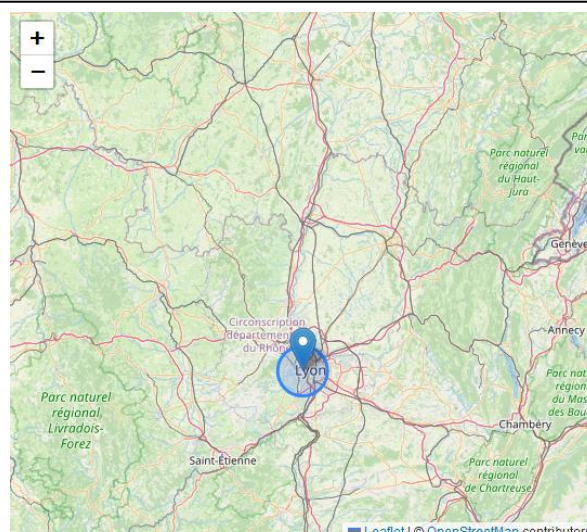
Coordonnées #	lieu	Latitude	Longitude
1	Branly	45.7533655	4.7912060
2	ND de la garde	43.2875718	5.3710335
3	Aéroport de nice	43.6602202	7.2143580
4	Ajaccio		

```
var lines = L.polyline([  
  [45.7533655, 4.7912060],  
  [43.2875718, 5.3710335],  
  [43.6602202, 7.2143580],  
  [....., ..... ]  
], {color: 'red'}).addTo(carte);
```

2.3.3 Un cercle

Comme en mathématiques, un cercle « **circle** » est caractérisé par un centre (coordonnées GPS) et un rayon (en mètres)

```
var influence = L.circle([45.7533655, 4.7912060], 10000).addTo(carte);
```





TPx Leaflet&OpenStreetMap

L'alternative open source à
Google Maps®



Par défaut le cercle est de couleur bleu semi-opaque avec un bord bleu foncé. Cela peut être changé via des options.

```
var influence = L.circle([46.6835956, -0.4137665], 210000, {  
  'color': '#FF7F00',  
  'fill': true,  
  'fillColor': '#FFFF00',  
  'fillOpacity': 0.2,  
}).addTo(carte);
```

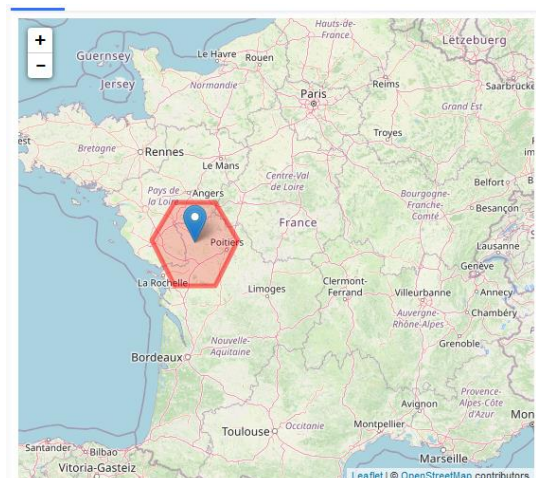
Se référer à la [page de documentation](#) pour tester les options.

2.3.4 Un polygone

Et bien rien de plus simple, un polygone ce n'est rien de plus qu'un polyline fermé dont on pourra colorier l'intérieur.

Sans plus attendre, voici un ensemble de coordonnées. Je vous laisse comme exercice le soin d'obtenir le résultat suivant :

Coordonnées #	Latitude	Longitude
1	47.318398	-0.886464
2	47.318398	0.069346
3	46.722971	0.5862335
4	46.000000	0.069346
5	46.000000	-0.886464
6	46.722971	-1.4137665



Pour supprimer une forme, rien de plus simple, il suffit d'appeler la méthode [removeLayer](#) de votre carte et de lui donner le marqueur (ou autre élément) à supprimer.

```
carte.removeLayer(marker);
```

3. Un peu plus d'interactivité

Les cartes s'affichent et on peut déjà transmettre des informations à l'utilisateur via l'affichage. L'étape suivante permettra d'augmenter les interactions via la gestion des clics et autres *drag'n drop*.



TPx Leaflet & OpenStreetMap

L'alternative open source à
Google Maps®



V1.0 Septembre
2022

3.1 Exemple simple, le clic

Il s'agit ici d'ajouter une fonction à appeler lors du déclenchement de l'événement. Cela se fera via la méthode « **on** » à laquelle on passera le nom de l'événement (**click**) puis la fonction à appeler au déclenchement.

Par exemple, pour appeler une fonction "placerMarqueur" on fera :

```
carte.on('click', placerMarqueur);

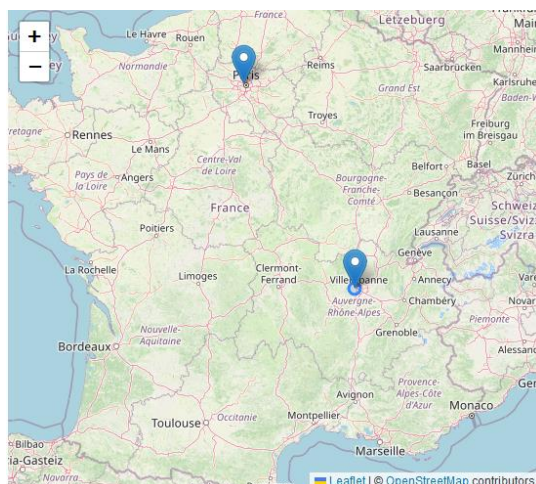
function placerMarqueur(e) {
    // Faire quelque chose suite à l'événement
}
```

Vous remarquerez que la fonction appelée par le clic possède un paramètre *e*. Ce dernier porte des infos sur l'événement, comme le lieu géographique du clic par exemple.

Voici un exemple un peu plus complet qui déplacera le marqueur à la position du clic.

```
var carte = L.map('macarte').setView([46.5630104, 1.4846608], 6);
L.tileLayer('http://{s}.tile.osm.org/{z}/{x}/{y}.png', {
    attribution: '&copy; <a href="http://osm.org/copyright">OpenStreetMap</a> contributors'
}).addTo(carte);
var marker = L.marker([48.86131823500415, 2.2941617066602973]).addTo(carte);
carte.on('click', placerMarqueur);

function placerMarqueur(e) {
    marker.setLatLng(e.latlng);
};
```





TPx Leaflet&OpenStreetMap

L'alternative open source à
Google Maps®



V1.0 Septembre
2022

3.2 Toujours plus d'infos, la popup

Ajouter une petite couche d'informations via une [infobulle](#). Cette infobulle pourra être peuplée de plein d'informations qui vous semblent pertinentes. L'infobulle de Leaflet s'appelle « **popup** ».

Pour attacher une « popup » à un élément (comme un marqueur), on utilise la fonction **bindPopup** que l'on peut éventuellement tout de suite personnaliser avec un texte (qui peut-être mis en forme avec des balises html). On peut ensuite récupérer cette popup pour la manipuler via la méthode **getPopup**.

```
var marker = L.marker([46.6835956, -0.4137665]).addTo(carte);  
marker.bindPopup(""); // Je ne mets pas de texte par défaut  
var mapopup = marker.getPopup();
```

Il faut maintenant appeler la fonction **openPopup** du « **marqueur** » (ou de l'objet à laquelle vous l'avez attachée).

```
mapopup.setContent('Salut, ça farte ?'); // je personnalise un peu avant d'afficher  
marker.openPopup();
```



3.3 Glisser et déposer

En anglais on parle de *draggable*, c'est donc sous ce terme qu'on le trouvera dans [la doc](#).

Il suffit simplement de rajouter l'option **draggable** à **true** pour que notre marqueur devienne déplaçable à la souris. Une fois cela fait, il pourra générer entre autres deux nouveaux événements intéressants : **dragstart** et **dragend**.



TPx Leaflet & OpenStreetMap

L'alternative open source à
Google Maps®



V1.0 Septembre
2022

```
var marker = L.marker([46.6835956, -0.4137665], {draggable:'true'}).bindPopup('').addTo(carte);  
  
marker.on('dragend', relachement);  
  
function relachement(e) {  
    marker.getPopup().setContent(""+marker.getLatLng());  
    marker.openPopup();  
}
```

① Bien entendu, ce ne sont pas là les seules actions possibles sur les objets ou sur la carte. [La documentation](#) vous tend les bras si vous voulez en savoir plus !

4. Le mécanisme des plugins

Le combo **Leaflet + OSM** de base permet déjà de faire beaucoup de choses. Mais grâce à son système de plugins, Leaflet peut proposer encore bien plus, proposant des outils pour afficher de nouveaux fond de cartes ou encore de nouveaux marqueurs et autres indicateurs visuels. Tout comme la bibliothèque Leaflet, les plugins sont open sources. Ils sont tous recensés sur la page "[Plugins](#)" du site original et regroupés par catégories de fonctions (fonds de cartes, affichages, interactions...).

Exemple, MarkerCluster

À titre d'exemple, je vous propose d'intégrer l'incontournable plugin [MarkerCluster](#) qui propose de regrouper les marqueurs géographiquement proches lorsque vous [dé]zoomer sur la carte ([dém](#)o).

Installation

Bon, pas de chance, en général l'étape d'installation est la plus pénible. Certains plugins passent par des outils comme npm, d'autres un simple lien à lier après téléchargement de la librairie minifiée.

MarkerCluset fait partie de la catégorie "installation via Bower". Cependant, en cherchant un peu on peut facilement trouver un CDN le fournissant :

```
<link rel="stylesheet" href="http://cdn.leafletjs.com/leaflet/v0.7.7/leaflet.css" />  
1 <link rel="stylesheet" type="text/css"  
2 href="http://cdnjs.cloudflare.com/ajax/libs/leaflet.markercluster/0.4.0/MarkerCluster.css" />  
3 <link rel="stylesheet" type="text/css"  
4 href="http://cdnjs.cloudflare.com/ajax/libs/leaflet.markercluster/0.4.0/MarkerCluster.Default.css" />  
5 <script src="http://cdn.leafletjs.com/leaflet/v0.7.7/leaflet.js"></script>  
6 <script  
src='http://cdnjs.cloudflare.com/ajax/libs/leaflet.markercluster/0.4.0/leaflet.markercluster.js'></script>
```



TPx Leaflet & OpenStreetMap

L'alternative open source à
Google Maps®



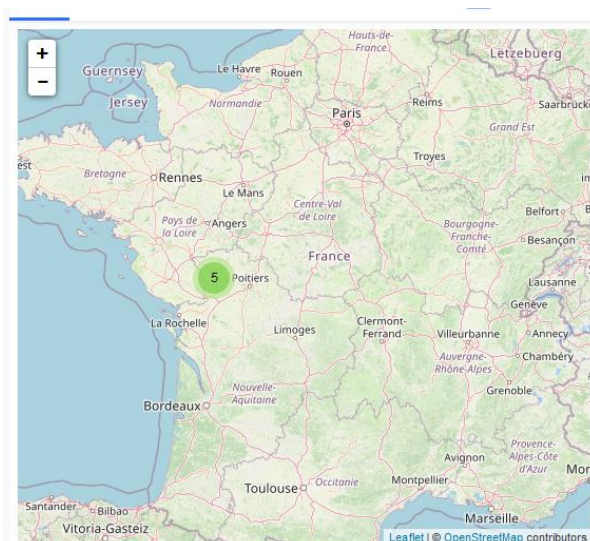
V1.0 Septembre
2022

Un petit rajout dans notre <head> et c'est parti !

Utilisation

Maintenant, pour l'utilisation c'est comme pour tout plugin de tout logiciel, on lit [la doc](#) et on démarre avec [un exemple](#) !

On découvre alors que ce plugin est tout simple puisqu'il suffit simplement d'ajouter un **markerClusterGroup** à notre carte puis ajouter des marqueurs classiques à ce dernier.



Voyez plutôt :

Sources :

Article « des cartes sur votre site » du 05/07/2016 sur ZesteDeSavoir.com, écrit par Eskimon (<https://zestedesavoir.com/tutoriels/1365/des-cartes-sur-votre-site/>)





TPx Leaflet&OpenStreetMap

L'alternative open source à
Google Maps®



Exercice : Calculateur de distance

Pour ce petit exercice, on se propose de faire un calculateur de distances. Ce calculateur devra être en mesure d'afficher la distance entre deux marqueurs et de tracer un joli segment entre les deux. Chaque marqueur doit pouvoir être déplacé et lorsque le déplacement se termine, une popup doit apparaître sur le marqueur déplacé pour afficher la distance entre les deux.

Pour avoir un aperçu du résultat, rendez-vous ci-dessous dans la rubrique correction.

Indices

Il y a pas mal de choses que l'on n'a pas vu encore, mais tout est dit dans [la documentation](#) en allant chercher les bons mots-clés.

Indice #1 : Comment calculer une distance ?

Indice #2 : J'ai déplacé un marqueur, comment je le récupère dans l'événement ?

Amélioration

Pour les plus courageux d'entre vous, voici quelques idées d'améliorations pour faire un calculateur de distance du tonnerre !

1. Première amélioration, réussir à placer plusieurs marqueurs lors d'un clic. À chaque clic, un marqueur se rajoute et le chemin s'étend jusqu'à ce nouveau marqueur. On affiche alors la distance totale sur ce nouveau marqueur (somme de tout les tronçons). Bien entendu la distance doit se recalculer lors du déplacement d'un marqueur.
2. Ensuite, il serait sympa d'afficher une popup par dessus les marqueurs, pour avoir les distances intermédiaires (en gros le point A n'affiche rien, le B affiche la distance de [AB], le C affiche [AB]+[BC] puis le D affiche la somme de tout, [AB]+[BC]+[CD]). Comme je suis sympa, je vous laisse choisir entre afficher les distances intermédiaires soit au clic sur le marqueur, soit à son survol.
3. Enfin, petit comble du confort, un double clic sur un point le supprime (et comme on veut une belle application, les distances doivent se recalculer proprement et le dessin du chemin aussi).

Bon courage !