

API Map : manipulations simples

Une API (pour Application Programming Interface), que l'on pourrait aussi appeler bibliothèque logicielle, est un ensemble d'outils qui sont mis à disposition des programmeurs pour réaliser des fonctionnalités bien identifiées. Ici nous allons explorer une API pour manipuler des cartes (satellites ou routières).

Nous allons utiliser l'API ([MapQuest](#)) qui offre des outils pour manipuler des cartes. MapQuest permet d'obtenir gratuitement une clef d'accès qui permet 15000 chargements de cartes gratuits par mois. Vous pouvez demander à obtenir votre clef personnelle si vous souhaitez faire des essais avec cette API. Il faudrait alors dans ce qui suit remplacer la clef proposée.

La première chose à faire est de déclarer l'utilisation de l'API utilisée. Dans notre cas cela se fait en ajoutant les lignes suivantes dans la partie `head` de la page HTML, avant la déclaration de nos propres scripts :

```
<script src="https://api.mqcdn.com/sdk/mapquest-js/v1.3.2/mapquest.js"></script>
<link type="text/css" rel="stylesheet" href="https://api.mqcdn.com/sdk/mapquest-
js/v1.3.2/mapquest.css"/>
```

Il faudra également que notre page HTML contienne un élément identifié qui accueillera l'image de la carte désirée. Donc quelque chose comme :

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

Il est nécessaire de définir une largeur à cet élément sous peine d'avoir une carte de taille 0. Cela doit se faire dans une feuille de style annexe. Et on peut bien sûr ajouter à cet élément une ou des classes CSS `class="..."`

Pour avoir le droit de télécharger les images correspondant aux cartes il faut fournir une clef valide (voir ci-dessus). Pour fournir cette clef il faut exécuter le code suivant :

```
1 L.mapquest.key = 'dW6cmwiVeDCyIJXMGUcHte8xLru8zpgA';
```

Le chargement de l'API nous permet d'en utiliser les définitions. En particulier nous avons maintenant accès à un nouveau type de données représentant les cartes : `L.mapquest.map`. Pour créer une telle donnée, il faut donner deux informations. La première est l'id de l'élément HTML de la page qui contiendra la carte, le second est une donnée représentant des options pour la carte. Ces options définissent en particulier :

- les coordonnées (latitude, longitude) du centre de la carte : option `center`, de type `L.LatLng`
- le facteur de zoom : option `zoom`, un entier
- le type de carte utilisée : option `layers`. Les valeurs sont, par exemple : `L.mapquest.tileLayer('map')`, `L.mapquest.tileLayer('satellite')`, `L.mapquest.tileLayer('hybrid')`.

Pour créer une carte, il faut donc définir ces données ce qui donne quelque chose comme :

```
1
2 var leCentre = L.latLng(50.609731,3.137511);
3 var mapOptions = {
4   center: leCentre,
5   layers: L.mapquest.tileLayer('map'),
6   zoom: 15
7 };
8 // 'maCarte' correspond à l'id de l'élément <div> ci-dessus
9 var laCarte = L.mapquest.map('maCarte', mapOptions);
```

L'API propose en plus sur les données de type carte des fonctions telles que `setView`, `setZoom`, `addLayer` et `removeLayer` qui ont pour effet de modifier les options correspondantes pour la carte concernée. Ainsi

```
1 laCarte.setZoom(15);
```

fixe le facteur de zoom de la donnée `laCarte` à 15.

- 1- Le script de l'API et les informations sur les images sont obtenues à partir d'information récupérées par le réseau. Pour éviter des problèmes d'affichage du document, il est donc préférable de déclencher la création de la carte après le chargement complet de la page. On utilisera (comme pour la fonction de mise en place des événements `setupListeners`) l'événement `Load` sur l'élément `window` qui appellera une fonction dont le traitement correspondra au code ci-dessus.

- 2- Créer des marqueurs Text sur quelques lycées de Marseille

Rassemblez les différentes informations ci-dessus, placez-les dans une fonction et utilisez le tout pour construire une page html contenant votre première carte non statique.

Documentation <https://developer.mapquest.com/documentation/mapquest-js/v1.3/1-mapquest-map/>