

JQuery plugin / Google Maps API

...

Thomas PERRON
Pierre ROUZIC

Sommaire

- C'est quoi un plug-in ?
 - Définition
- Définir un plug-in sur jQuery
 - syntaxe
 - valeurs par défaut
 - each
 - *utilisation de la compression du code (2 versions à livrer)-> assets compressor sur WebStorm*
- Google Maps API JavaScript
 - Présentation
 - Plugins populaires
 - Docs associés
 - Présentation des API

Plugins JQuery

C'est quoi un plugin ?

Paquet / fichiers fournissant un ensemble de fonctions à appliquer permettant l'ajout de nouvelles fonctionnalités.

- modularité et intelligibilité
- easy-to-use (installation et documentations)

Ex: composant natif (Cordova), UI (jQuery-UI), animation, text-form ...

<https://plugins.jquery.com/>

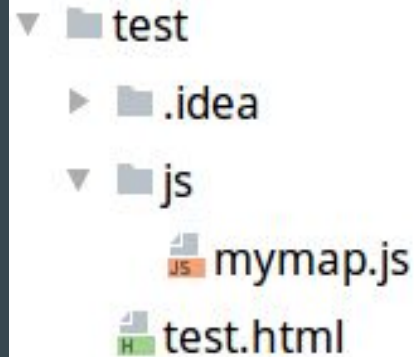
<https://cordova.apache.org/plugins/>

GITHUB

Comment écrire un plugin JQuery ?

Dans un fichier .js (de préférence dans le dossier js de votre projet).

```
<script src="nom_du_plugin.min.js"></script>
```



Créer un plug-in JQuery

<https://learn.jquery.com/plugins/basic-plugin-creation/>

- `$.fn.function_name` // required
- `$.fn.extend` // optionnal
- `.each()` // optionnal
- fonction de callback // optionnal

.fn.function_name()

```
$(function() {  
  
    var tab = [];  
    var lstorage = window.sessionStorage;  
    lstorage.localstorage = JSON.stringify(tab);  
  
    /**  
     * Creates map on web page  
     * @param options  
     * @returns {$}  
     */  
    $.fn.createMap = function (options) {  
        // default options for map initialization -> zoom , center and map  
        options = $.extend({  
            zoom: 8,  
            coords: [48.895651, 2.290569],  
            type: "ROADMAP"  
        }, options);  
  
        $("#map").createMap();  
        $("#map").putMarker({coords: [48,2]});  
        $("#map").putMarker({coords: [49,2]});  
    }  
});
```

Déclaration d'une fonctionnalité appartenant à un plug-in.

Pour utiliser une fonctionnalité d'un plug-in, il suffit de coupler la fonction à un objet JQuery.

\$.extend({})

Permet de définir des paramètres par défaut à une fonctionnalité d'un plug-in.

Possibilité de réécriture de ces différents paramètres.

Avantage : Permet une gestion par défaut des paramètres.

```
$(function() {  
    $.fn.colorize = function( options ){  
        options = $.extend({color:"green",  
            backgroundColor: "red",  
            onTextChanged : function(){}},  
            options);  
  
        // permet de renvoyer un objet jQuery pour le réutiliser  
        return this.css({  
            color: options.color,  
            backgroundColor: options.backgroundColor  
        });  
    }  
});
```

```
$('div').colorize({color:"yellow"});  
$('#test_text').colorize();
```

.each()

```
this.each(function () {  
    // adds marker on map at marker's position latitude and longitude  
    var marker = new google.maps.Marker({  
        map: $(this).data('googleMap'),  
        position: new google.maps.LatLng(options.coords[0], options.coords[1]),  
        draggable: options.draggable  
    });  
  
    // adds markers on local database
```

this.each(function () {

}// mettre le code dans cette fonction
permet d'exécuter sur la fonction sur
tout ce sur quoi elle a été appelé

\$('.a').second_plugin(); sans le each la
fonction s'applique qu'au premier lien
alors qu'avec le each elle s'applique sur
tous les liens du document

Compression du code

Il existe deux versions qui sont utilisables :

- 1) la version complète : commentée, indentée, espacée
- 2) la version compressée : tout sur une ligne sans espace ni commentaire

Pourquoi ?

Inutile pour la majorité des utilisateurs de récupérer le code intégral, seule une version compressée et légère est suffisante.

Google Maps API Web Services

Présentation

Plusieurs API disponible pour différentes plateformes :

Android, JavaScript ...

Pour JavaScript : Maps JavaScript API

<https://developers.google.com/maps/documentation/javascript/?hl=fr>

Utiliser Maps JavaScript API

```
<script type="text/javascript"  
src="https://maps.googleapis.com/maps/api/js?key=YOUR_KEY"></script>
```

Dans l'entête HTML, avec la clé générée via votre compte Google.

Indispensable pour faire le TP !!

Comment fonctionne Maps JavaScript API ?

Requêtes JSON via protocole HTTPS, garantissant la sécurité des données.

Format requête :

Service : service demandé (géolocalisation, geocoder ...)

Output : XML ou JSON

```
https://maps.googleapis.com/maps/api/service/output?parameter
```

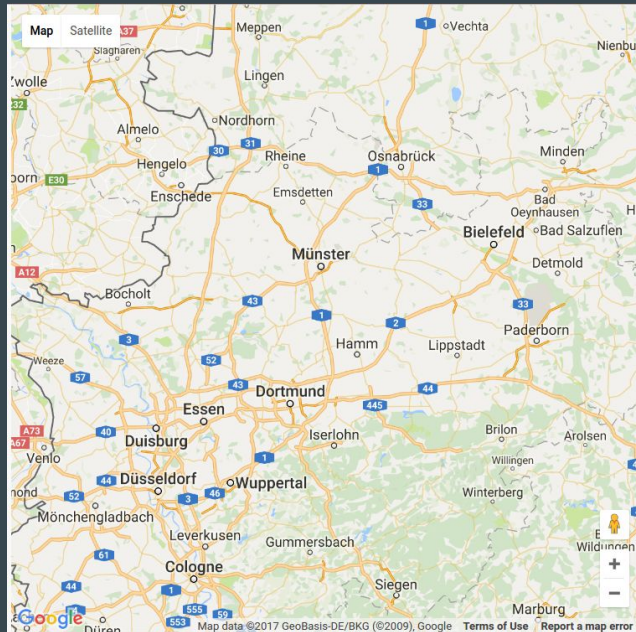
s

Créer une carte

```
// map initialization
var map = new google.maps.Map(this, {
  zoom: options.zoom,
  center: new google.maps.LatLng(options.coords[0], options.coords[1]),
  mapTypeId: options.type,
});
```

Trois paramètres indispensables :

- Niveau de zoom
- Centre de la carte
- Typologie de la carte



Créer un marqueur

```
// adds marker on map at marker's position latitude and longitude  
var marker = new google.maps.Marker({  
  map: $(this).data('googleMap'),  
  position: new google.maps.LatLng(options.coords[0], options.coords[1]),  
  draggable: options.draggable  
});
```

Deux paramètres obligatoires :

- Carte associée au marqueur
- Position du marqueur sur la carte

Autres paramètres :

- Type d'icône, titre, photos, taille marqueur ...

Geocoding Service

Service permettant de localiser une ville, adresse, et de retourner ses coordonnées.

Le service peut également s'utiliser comme Geocoding Reverse (obtention de différentes informations à partir des coordonnées (LatLng)).

```
var geocoder = new google.maps.Geocoder();
var address = $("#addressTo").val();
geocoder.geocode( { 'address': address}, function(results, status) {
    if (status == google.maps.GeocoderStatus.OK) {
        var coords = results[0].geometry.location;
        $("#map").putMarker({coords: [coords.lat(), coords.lng()]});
        $("#map").setCenter(coords);
    } else {
        alert("Le geocodage n'a pu etre effectue pour la raison suivante: " + status);
    }
});
```

Directions API

Permet de définir un itinéraire entre différentes coordonnées, ainsi que de sélectionner le mode de transport, les routes à éviter ...

Trois paramètres indispensables :

- Les coordonnées d'origines
- Les coordonnées d'arrivée
- Le mode de transport

DistanceMatrix



```
var origin1 = {lat: 55.930, lng: -3.118};  
var origin2 = 'Greenwich, England';  
var destinationA = 'Stockholm, Sweden';  
var destinationB = {lat: 50.087, lng: 14.421};
```

Results

14 Great Carleton Pl, Edinburgh EH16 4TP, UK to Stockholm, Sweden: 2,520 km in 1 day 2 hours

14 Great Carleton Pl, Edinburgh EH16 4TP, UK to Kožná 12, 110 00 Praha 1-Staré Město, Czechia: 1,892 km in 19 hours 8 mins

Royal Borough of Greenwich, UK to Stockholm, Sweden: 1,881 km in 20 hours 0 mins

Royal Borough of Greenwich, UK to Kožná 12, 110 00 Praha 1-Staré Město.

Permet d'obtenir différentes informations entre deux coordonnées (état du trafic, durée du voyage, distance entre coordonnées, éviter/sélectionner type de routes, sélectionner un type de transport spécifique (BUS, TRAIN...)).

Autres API

Geolocation API

Elevation API

TimeZone API

Roads API

...

Plugins populaires

- <https://hpneo.github.io/gmaps/>
- <https://tilotiti.github.io/jQuery-Google-Map/>
- <https://maplacejs.com/>
- <https://github.com/mapsplugin/cordova-plugin-googlemaps/wiki/map>

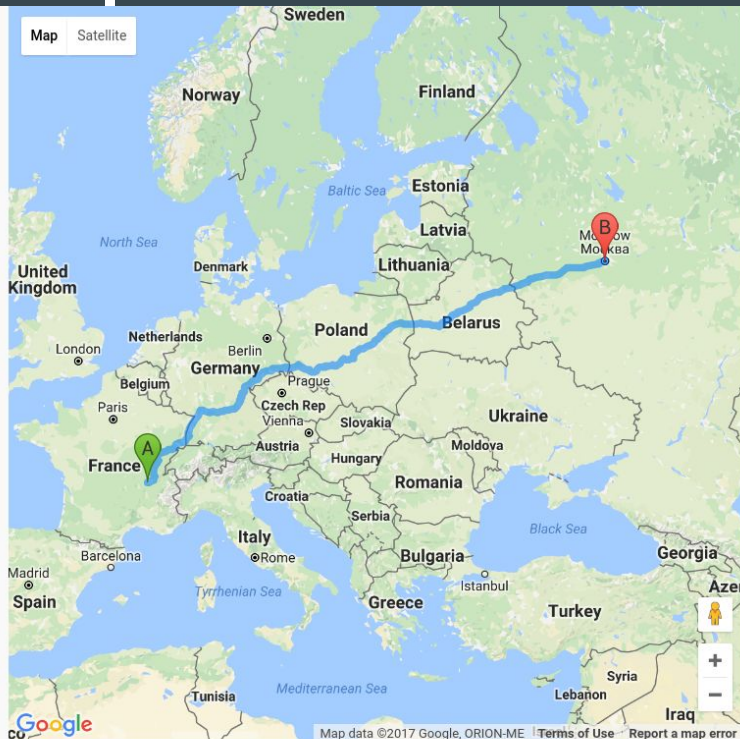
**Ce que l'on attend de
vous ...**

Transformation d'une fonction en plugin

...

**comprendre les mécanisme des
paramètres, objet littéral,
callback**

Attentes pour le TP



Moscow, Russia

Lyon, France

DRIVING MODE

POSITION VILLE DE DEPART

POSITION VILLE D'ARRIVEE

AVEC AUTOROUTES

AVEC PEAGES

TRACER L'ITINERAIRE

Lieu de départ.

Lieu d'arrivée.

2,956 km

Distance.

1 day 5 hours

Durée.

Documentations et exemples

Pour récupérer une clé :

<https://developers.google.com/maps/documentation/javascript/get-api-key?hl=fr>

Rendez-vous sur

<https://developers.google.com/maps/documentation/javascript/reference>

pour la documentation officielle. (Assez indigeste)

Pour quelques exemples :

<https://developers.google.com/maps/documentation/javascript/examples/>

Au boulot !

(Merci de votre attention :))