

Mise en pratique - Découverte HTML5 - CSS3 - FlexBox - BootStrap - JavaScript - JQuery - JQuery Mobile - Cordova

Afpa © 2019 – Section Tertiaire Informatique – Filière « Etude et développement »

## TABLE DES MATIÈRES

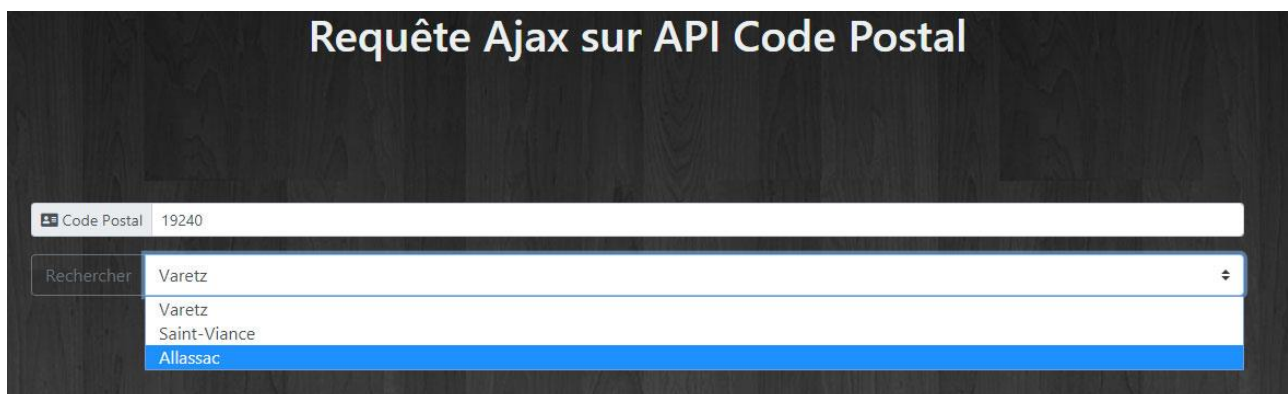
1. UTILISATION DE L'API WEB : ZIPPOPOTAM.....	3
2. LES PAYS DU MONDE - EXPLOITATION DE FLUX JSON AVEC AJAX ET JQUERY .....	3

## 1. UTILISATION DE L'API WEB : ZIPPOPOTAM

L'objectif de ce TP est de mettre en œuvre un système d'appel **Ajax** sur une **API Web** (<http://zippopotam.us/>) qui fournit les noms des communes en fonction du **Code Postal** que l'application utilisateur demande.

Dans un premier temps, vous étudierez cette API et vous essayerez de mettre en œuvre une page Web qui interrogera cette API à travers une **requête Ajax** écrite en code **JavaScript** natif (utilisation de l'objet **XMLHttpRequest**).

Vous vous concentrerez uniquement sur les **Communes Françaises** !!!



## 2. LES PAYS DU MONDE - EXPLOITATION DE FLUX JSON AVEC AJAX ET JQUERY

Afin de mener à bien ce TP, vous devrez placer vos scripts sur votre serveur **Apache**.

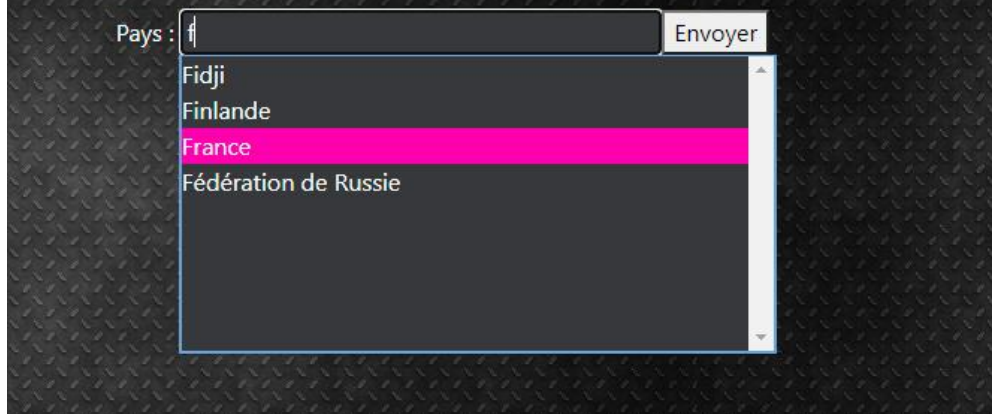
Si vous arrivez à ce TP c'est que vous avez déjà réalisé le TP sur l'**auto-complétion** en **JQuery** proposé dans le **livret d'apprentissage d'Ajax**. Si toutefois vous n'avez pas réussi à implémenter votre propre système d'auto-complétion vous pouvez passer directement à la **deuxième partie de ce TP**.

Le fichier JSON qui contient la liste de tous les pays du Monde : **pays.json** correspond à vos données internes de l'entreprise. C'est sur ce script JSON que vous requêter de façon **asynchrone** pour obtenir l'**auto-complétion**.

Mise en pratique – Découverte HTML5 - CSS3 – FlexBox - BootStrap - JavaScript –  
JQuery - JQuery Mobile - Cordova

Afpa © 2020 – Section Tertiaire Informatique – Filière « Etude et développement »

# Introduction à Ajax (jQuery)



Pays : f Envoyer

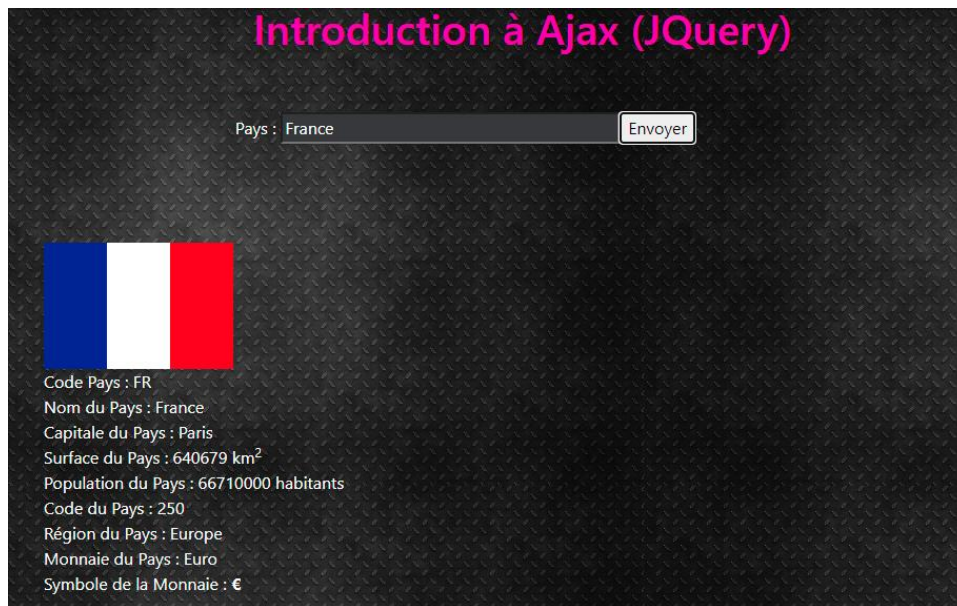
- Fidji
- Finlande
- France
- Fédération de Russie

Les données contenues dans le script **JSON** initial étant trop limitées, je vous propose donc, afin d'enrichir votre interface utilisateur, d'aller maintenant chercher des **informations supplémentaires** sur **l'API REST FULL** à l'adresse suivante : <http://restcountries.eu/>

Vous devrez prendre le temps d'étudier cette **API** avant de continuer ce TP.

Dans cette **deuxième phase**, lorsque l'utilisateur de l'application aura sélectionné un pays dans la **pseudo-liste déroulante**, après la soumission du formulaire, une **nouvelle requête Ajax** sera envoyée pour interroger les données proposées par le **Web Service**.

Vous devrez exposer cette fois les informations suivantes (**données externes** à l'application) : **Capitale du Pays**, **Surface du Pays**, **Population du Pays**, **Code du Pays**, **Région du Pays**, **Monnaie du Pays** et **Symbole** de la monnaie. Votre interface utilisateur devra ressembler à ce qui suit : (Après la soumission du formulaire).



**Deuxième partie :** (Pour ceux d'entre vous qui n'ont pas réussi à implémenter leur propre système d'auto-complétion)

Pour cette deuxième phase, vous devrez présenter à l'utilisateur un système d'**auto-complétion** portant sur le champ *input* du nom du pays recherché.

Vous cibleriez toujours le fichier JSON initial (**pays.json**) et il devra renvoyer la liste des pays en fonction de la saisie utilisateur (vous présenterez à l'utilisateur le **nom du pays** ainsi que son **code alpha 2** (code sur deux lettres, exemple : France - FR)).

Afin de faciliter votre tâche dans vos développements, vous pourrez mettre en place le système d'**auto-complétion** : **EasyAutocomplete** proposé par la bibliothèque **JQuery**. Pour sa mise en œuvre référez-vous à la documentation en ligne : <http://easyautocomplete.com/guide> .

Lorsque la requête **AJAX** renverra une **erreur 404** vous informerez l'utilisateur qu'il pourra relancer sa recherche en utilisant le **code alpha 2** (sur deux lettres) du pays, normalement, il y aura toujours un renvoi de données. C'est pour cette raison que je vous ai incité à afficher ce code **alpha2** dans la liste de réponse de l'**auto-complétion** ;- )

**Indice** : Vous devrez solliciter l'**API REST Countries** avec le point de terminaison REST : **Full Name** pour pouvoir avancer ... et pour pouvoir exposer les données supplémentaires relatives au pays recherché.



### Troisième partie : (Affichage de la carte du pays)

Pour cette dernière partie, vous devrez présenter la carte du pays actif à droite de la zone des informations sur le pays.

Pour ce faire, vous devrez étudier l'**API Google Maps** et ainsi vous pourrez implémenter cette nouvelle fonctionnalité.

**Indice** : Vous devrez fournir à l'**API GM** les **coordonnées de géolocalisation** du pays considéré (**Latitude** et **Longitude**). En choisissant un **zoom actif** de départ à **5**, vous obtiendrez une représentation de la carte du pays qui sera convenable.

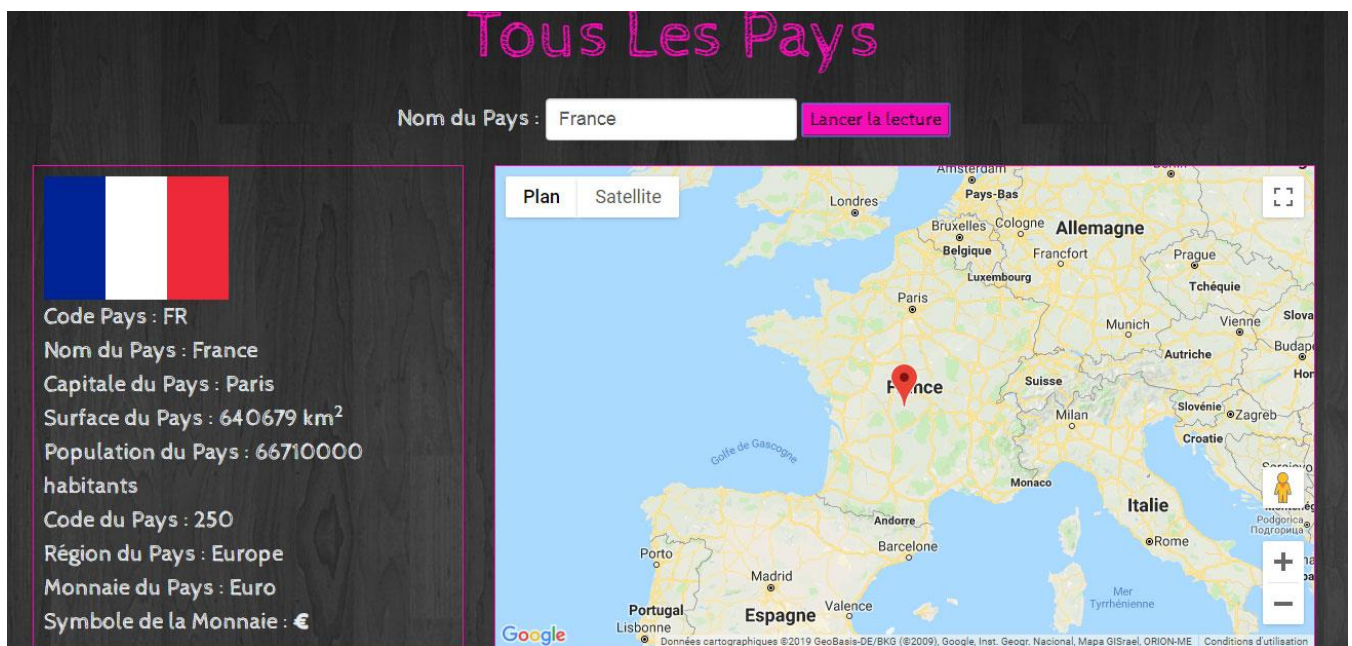
**Remarque** : Si vous êtes réticent à donner vos codes de carte bleue à Google vous pouvez aussi explorer la solution : **Leaflet** (<https://leafletjs.com/>) qui est une bibliothèque JavaScript qui permet d'insérer une carte interactive sur votre application ;-)

Le résultat attendu sera conforme à ce qui suit :

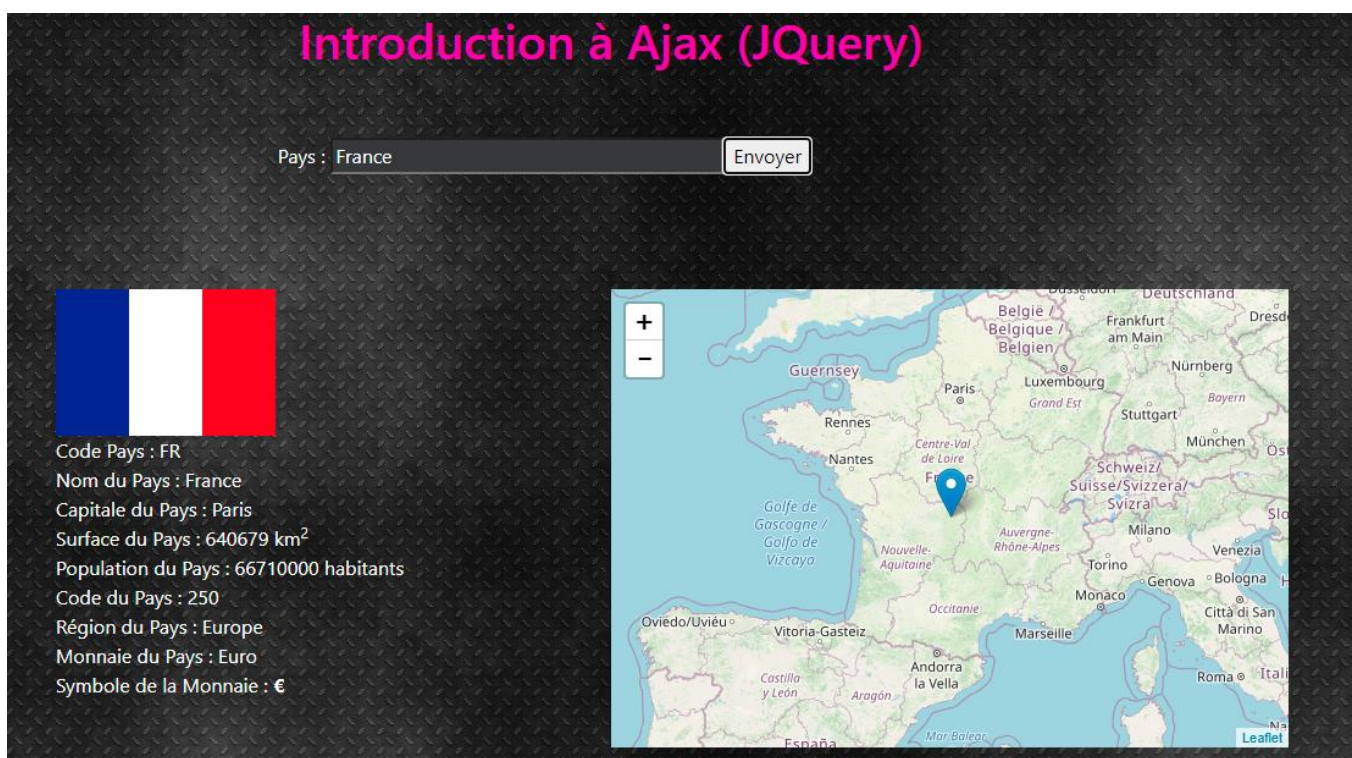
Mise en pratique – Découverte HTML5 - CSS3 – FlexBox - BootStrap - JavaScript –  
jQuery - JQuery Mobile - Cordova

Afpa © 2020 – Section Tertiaire Informatique – Filière « Etude et développement »





Avec l'API de [Google Maps](#).



Avec la Bibliothèque [Leaflet](#).

Mise en pratique – Découverte HTML5 - CSS3 – FlexBox - BootStrap - JavaScript –  
jQuery - JQuery Mobile - Cordova

Afpa © 2020 – Section Tertiaire Informatique – Filière « Etude et développement »

## **Reproduction interdite**

Article L 122-4 du code de la propriété intellectuelle.

« Toute représentation ou reproduction intégrale ou partielle faite sans le consentement de l'auteur ou de ses ayants droits ou ayants cause est illicite. Il en est de même pour la traduction, l'adaptation ou la reproduction par un art ou un procédé quelconque. »

### **CREDITS**

#### **ŒUVRE COLLECTIVE DE L'AFPA**

**Sous le pilotage de la DIIP et du centre d'ingénierie sectoriel Tertiaire-Services**

#### **Equipe de conception (IF, formateur, mediatiseur)**

Formateur : Alexandre RESTOUEIX

**Date de mise à jour : 29/05/2019**

Mise en pratique – Découverte HTML5 - CSS3 – FlexBox - BootStrap - JavaScript –  
JQuery - JQuery Mobile - Cordova

Afpa © 2020 – Section Tertiaire Informatique – Filière « Etude et développement »