CCI Nîmes

Travaux pratique 

Mise en pratique et réalisation d’une API au travers de JS



Réalisé et enseigné par

Jérôme Ferratier



Table des matières

[Introduction 3](#_Toc87809128)

[Pré requis 4](#_Toc87809129)

[Javascript – Initialisation (Web service) 5](#_Toc87809130)

Introduction

Utilisation du javascript dans un environnement client au travers de la console d’un navigateur et d’une page web.

Pré requis

**Connaissance :**

Javascript

***Outils :***

VSCode

Javascript – Initialisation (Web service)

* **Chapitre 1. Consommation de web service Synchrone / Asynchrone**

Commençons par les basiques. Un web service via une URL retourne des données.

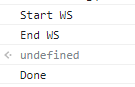
Ces données peuvent être des informations au format texte, du HTML, des images, des videos…

* **Etape 1 :**

Depuis le site : <https://javascript.info/xmlhttprequest>, bien lire la documentation.

Ensuite, consommer le webservice *« /article/xmlhttprequest/example/load »* avec la méthode « XMLHttpRequest » depuis l’onglet console de votre navigateur.

Avec l’aide de console.log, obtenez le résultat suivant :



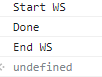
(Insérer ici, une impression d’écran de votre résultat).

Votre consommation de web service se fait elle de manière synchrone ou asynchrone ?

* **Etape 2 :**

Un paramètre dans l’appel de la fonction XMLHttpRequest permet de forcer le mode Synchrone.

Obtenez le résultat suivant.



(Insérer ici, une impression d’écran de votre résultat).

* **Etape 3 :**

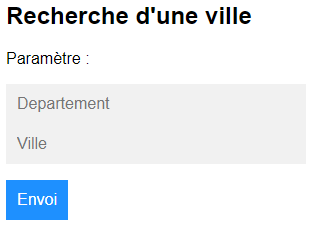
Depuis l’onglet Network de votre navigateur, quel type de donnée recevez-vous ?

* **Chapitre 2. Utilisation d’une API.**

Le site api.gouv.fr contient de nombreuse API gratuites et libres. Nous allons voir ensemble comment les exploiter.

Pour cela, nous allons réaliser un formulaire permettant de trouver et afficher les informations de n’importe quelle ville de France.

Formulaire attendu :



**Partie 1. Recherche d’une ville :**

Accédez à la page <https://api.gouv.fr/les-api/api-geo>.

* **Etape 1.1 :**

Grace aux formulaires disponible sur la page permettant de tester les web services, et grâce à vos onglets console ou network, trouvez le service qui recherche et retourne la liste des villes(communes) en fournissant un numéro département.

* **Etape 1.2 :**

Quel est le format du retour de ce web service ?

* **Etape 1.3 :**

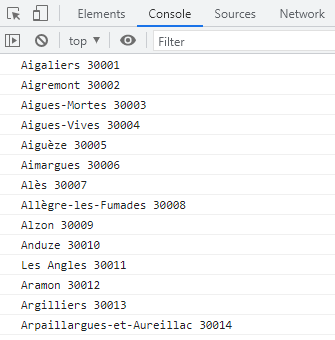
Depuis la page0.html insérer le code javascript permettant de :

1.3.1 : Consommer le web service avec un appel XMLHttpRequest (utilisez le département de votre choix).

1.3.2 : Convertissez la réponse du Web service avec la méthode JSON.parse.

Cette méthode convertit le format JSON en tableau.

1.3.3 : Parcourez le tableau récupéré et affichez les informations suivantes en console :

 (Utilisez des boucles for et les console.log).

**Partie 2. Formulaire web et Autocomplétions :**

Que se passerait -il si l’on demandait la liste complète de l’ensemble des articles d’amazon, l’ensemble des numéros de sécurité sociale sur Ameli, ou bien l’ensemble de te toutes les communes de France ? Nos programmes ou pages Web s’exposeraient à des crashs ou bien à de fortes lenteurs. En plus de faire planter les serveurs eux même ;).

Même si cela n’est pas autorisé, pour éviter cela il faut filtrer un minimum nos recherches au travers de nos programmes ou pages.

Nous utiliserons dans cet exercice un formulaire avec autocomplétions.

Pour cela, un exemple de bonnes pratiques de création de formulaire avec autocomplétion est disponible sur w3school via l’URL https://www.w3schools.com/howto/howto\_js\_autocomplete.asp

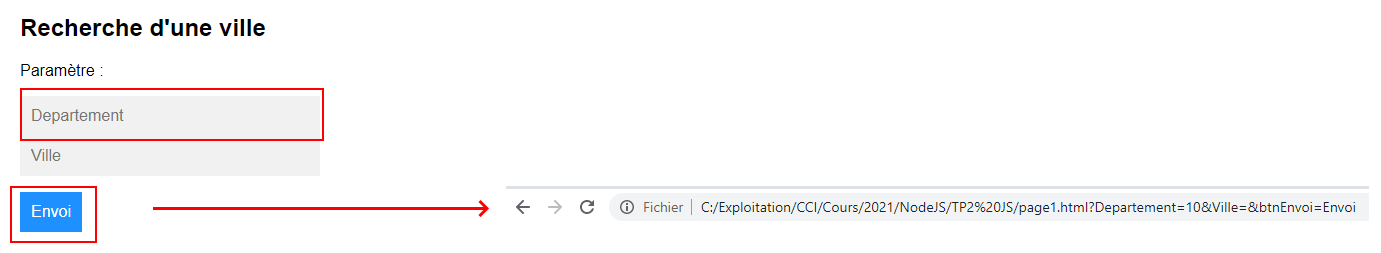
Ce formulaire a été très légèrement reprit et adapté à notre contexte sur la page page1.html.

* **Etape 1 :**

Depuis la page page1.html. Seule la recherche par département fonctionne.

La liste des départements a été écrite en dur dans le code, et est utilisée avec de l’auto complétion grâce à la méthode autocomplete ( voir code ).

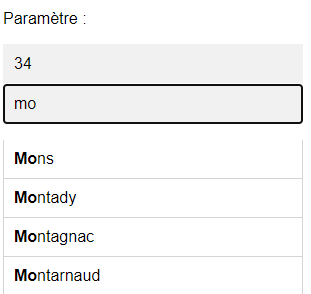
Lorsque l’on fait une recherche, le bouton « Envoi » émet une requête HTTP GET en passant des paramètres dans l’URL . Et c’est tout, aucune autre action n’est réalisée.

****

Implémentez la fonction loadCityList(dep) avec le web service de recherche de ville.

Grace à la méthode autocomplete, faite en sorte que le champs Ville propose une autocomplétions sur le département choisi.

Résultat attendu :

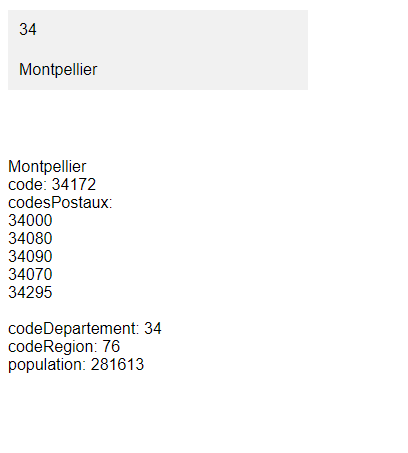


* **Etape 2 :**

Implémentez la fonction loadCity(ville) avec un web service approprié de recherche de ville.

Grace à la méthode autocomplete, faites en sorte que le champs Ville propose une autocomplétions sur le département choisi.

Résultat attendu :



* **Etape 3 :**

Lorsque les web services sont en cours de chargement, afficher un loader sur la page.

Pour cela, utilisez un gif, ou un css (Exemple <https://www.w3schools.com/howto/howto_css_loader.asp> ).

* **Etape 4 :**

Inscrivez-vous sur le site <https://api.meteo-concept.com/>.

Si site dispose de web service retournant la météo à partir d’un code insee de ville.

Implémentez ce web service pour avoir le résultat suivant :

