

## EXERCICES SUR LES REQUÊTES API

### Exercice 1 : Explorateur de régions, départements et communes

On utilisera l'API Geo fournie par [api.gouv.fr](https://api.gouv.fr) pour explorer les différentes entités administratives de la France.

#### Objectif :

Créez une page web qui permet à l'utilisateur d'explorer les régions, départements et communes de la France. L'utilisateur doit pouvoir :

- Sélectionner une région depuis un menu déroulant.
- Après avoir sélectionné une région, affichez un autre menu déroulant avec la liste des départements de cette région.
- Après avoir sélectionné un département, affichez un dernier menu déroulant avec la liste des communes de ce département.

#### Testez l'application :

- Ouvrez la page HTML dans un navigateur.
- Sélectionnez une région, puis observez comment le menu déroulant des départements est mis à jour.
- Sélectionnez un département, puis observez comment le menu déroulant des communes est mis à jour.

### Exercice 2 : Trouveur de fabricant d'adresse MAC

#### Objectif :

Créez une page web avec un formulaire permettant à l'utilisateur de saisir une adresse MAC. Lorsque l'utilisateur soumet le formulaire, utilisez l'API [macvendors.com](https://macvendors.com) pour obtenir et afficher le nom du fabricant du périphérique correspondant à cette adresse MAC.

#### Étapes :

1. Créez le formulaire HTML.
2. Créez le JavaScript (script.js).

#### Testez l'application :

- Ouvrez la page HTML dans un navigateur.
- Saisissez une adresse MAC valide (par exemple, 00:1A:78:00:00:01).
- Cliquez sur "Trouver le fabricant" et observez le résultat affiché en dessous du formulaire.



### **Exercice 3** : Générateur de blagues

#### Objectif :

Créez une page web avec un bouton qui, lorsqu'il est cliqué, affiche une nouvelle blague à l'utilisateur en utilisant l'API `jokeapi.dev`.

#### Étapes :

1. Créez le squelette HTML :
2. Créez le JavaScript (`script.js`) :

#### Testez l'application :

- Ouvrez la page HTML dans un navigateur.
- Cliquez sur "Donne-moi une blague !" et observez une blague qui s'affiche en dessous du bouton.

#### En supplément :

- Vous pouvez également ajouter des options pour permettre à l'utilisateur de filtrer les blagues par catégorie (par exemple, programmer, général, etc.).
- Prenez en compte la gestion des erreurs. Si l'API est indisponible ou si une autre erreur survient, assurez-vous d'afficher un message d'erreur approprié à l'utilisateur.

### **Exercice 4** : Création d'une API pour gérer des notes

#### Objectif :

Créez une API qui permet de gérer une collection de notes. L'API doit permettre de créer, lire, mettre à jour et supprimer des notes.

#### Pré-requis :

- Connaissance de base de JavaScript
- Node.js installé
- npm (Node Package Manager) installé

#### Étapes :

Initialisation du projet :

- *Créez un nouveau dossier pour votre projet.*

Installation des dépendances :

- *Installez Express et le body-parser (pour lire le corps des requêtes POST) :*

Création de l'API :

- *Créez un fichier `server.js`*



### Testez votre API :

Utiliser curl pour tester les différentes routes de votre API.

- Créer une nouvelle note (POST):

```
curl -X POST -H "Content-Type: application/json" -d  
'{"text":"Ma première note"}' http://localhost:3000/notes
```

Cette requête ajoute une nouvelle note avec le texte "Ma première note" à la liste des notes.

- Lire toutes les notes (GET):

```
curl http://localhost:3000/notes
```

Cette requête retourne toutes les notes disponibles dans la liste.

- Lire une note spécifique par son ID (GET): Prenons par exemple l'ID 1.

```
curl http://localhost:3000/notes/1
```

Cette requête retourne la note avec l'ID 1.

- Mettre à jour une note (PUT): Pour mettre à jour le texte de la note ayant l'ID 1.

```
curl -X PUT -H "Content-Type: application/json" -d  
'{"text":"Ma note mise à jour"}' http://localhost:3000/notes/1
```

Cette requête modifie le texte de la note avec l'ID 1 pour le remplacer par "Ma note mise à jour".

- Supprimer une note (DELETE): Pour supprimer la note ayant l'ID 1.

```
curl -X DELETE http://localhost:3000/notes/1
```

Cette requête supprime la note avec l'ID 1 de la liste.