

# Restaurants Chinois d'Antananarivo

Dans la base de connaissance on constate de nombreuses restaurants chinois qui se situe à Antananarivo mais leur probabilité qu'ils sont vraiment un restaurant chinois ne sont pas vraiment spécifié.

Premièrement nous utilisons un navigateur n8n que nous avons installé via la vidéo . Dans mon n8n j'ai spécifié deux workflow , le premier workflow consiste à récupérer les données de la liste de tous les restaurants Chinois qui se situe à Antananarivo .Le deuxième consiste à afficher le résultats en un fichier .HTML

- **\*\*Nœuds\*\*** : Tâches .
- **\*\*Arêtes\*\*** : Flux de données .
- 

## Dans le Premier WorkFlow de n8n .

Avant de commencer je mets un noeud pour qu'on peut tester le workflow manuellement puis après cela je mets un noeud de code ou je crée une boucle pour récupérer tout les restaurants chinois Antananarivo jusqu'à ce qu'il n'y en a plus . Après je crée un HTTP request pour appeler un url qui contient tous les informations pour les restaurations avec une clé `api_key` pour pouvoir récupérer les données .

Je crée un notre noeud de code pour extraire les données dont on a besoin . Je convertie on un fichier .JSON puis je le mets en binaire .

Voici les nombreuses étapes que j'ai franchi dans ce workflow :

### Étape 1 : Déclenchement manuel

Noeud : "When clicking 'Test workflow'"

Le workflow démarre manuellement lorsque l'utilisateur clique sur le bouton **"Test workflow"** dans l'interface n8n. Cela sert à tester le flux sans automatisation externe.

---

### Étape 2 : Génération de requêtes paginées

Noeud : "Code"

Ce noeud génère une liste de paramètres `start` correspondant à la pagination (0, 20, 40, ..., jusqu'à 100), pour interroger plusieurs pages de résultats via l'API de SerpAPI.

---

## Étape 3 : Requête vers l'API SerpAPI

### Noeud : "HTTP Request"

Envoie une requête `GET` à l'URL `https://serpapi.com/search.json` avec les paramètres suivants :

- `q=restaurant chinois antananarivo`
- `start={{ $json.start }}` (pagination)
- ``api_key=..`
- `ll=@-18.792,47.5079,15z` (coordonnées d'Antananarivo)

Ce noeud récupère des données Google Maps sur les restaurants chinois à Antananarivo.

---

## Étape 4 : Extraction et formatage des résultats

### Noeud : "Code1"

Ce script analyse les résultats JSON retournés, extrait les informations utiles pour chaque restaurant :

- `nom`, `contact`, `adresse`, `latitude`, `longitude`, `spécialité`, `photo`, `site`

Chaque restaurant est structuré en tant qu'objet JSON propre.

---

## Étape 5 : Conversion en fichier JSON

### Noeud : "Convert to File"

Prend les objets JSON formatés dans l'étape précédente et les transforme en un **fichier JSON** prêt à être sauvegardé.

---

## Étape 6 : Sauvegarde du fichier sur le disque

### Noeud : "Read/Write Files from Disk"

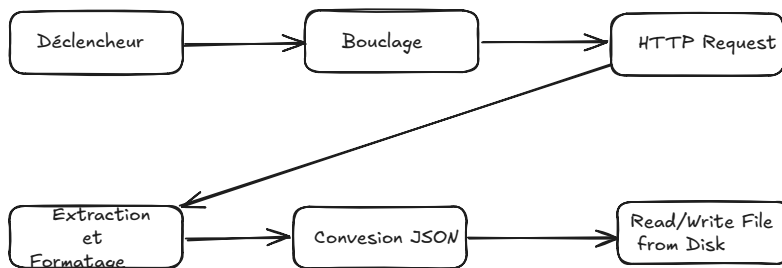
Écrit le fichier nommé `restaurantsChinoise.json` sur le disque local, contenant tous les restaurants trouvés et formatés.

---

## Résultat final :

Un fichier JSON contenant une **liste structurée des restaurants chinois à Antananarivo** récupérés via SerpAPI, prêt à être utilisé ou partagé.

### Apreçu du graphe dans workflow 1 .



## Dans le Deuxième WorkFlow de n8n .

Dans ce second worflow j'appelle le fichier .JSON que j'ai utilisé dans mon premier n8n , on spécifiant un chemin exacte dans mon noeud appeler execute command puis je lis avec un noeud read/write file from disk ce qui est spécifié à lire seulement .

Après cela j'extract le fichier .JSON de mon premier workflow , puis j'extract le fichier et je fais une classification afin de bien définir les critères qui prouve que c'est vraiment un restaurant chinois .

Avant de l'afficher je fais passer la classification avec un test qui s'appelle test de khi<sup>2</sup> qui permet de donner un pourcentage afin de bien définir le restaurant chinois avec ces critères et dans le mien j'ai fais transmettre 6 critères .

Voici les nombreuses étapes que j'ai franchi dans ce workflow :

1. **Déclenchement manuel** ( `manualTrigger` )
  - Permet de tester le workflow à la demande.
2. **Lecture du fichier JSON** ( `Read/Write Files from Disk` )

- Fichier source : `ListeRestaurant.json` .

### 3. Extraction des données JSON ( Extract from File )

- Transforme le fichier JSON en objets exploitables pour n8n.

### 4. Analyse / Scoring des restaurants ( Extraction et Classification )

- Attribue un score de "probabilité d'être un vrai restaurant chinois" à chaque entrée en fonction de critères :
  - Nom chinois détecté
  - Spécialité chinoise
  - Emplacement dans des lieux fréquentés par la communauté chinoise
  - Décor évoquant la culture chinoise
  - Note  $\geq 4$
  - Nombre d'avis  $\geq 5$

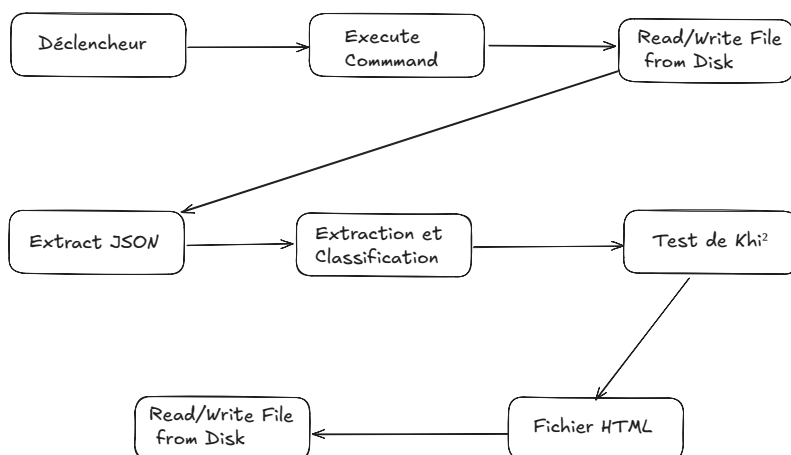
### 5. Filtrage avancé avec test $\chi^2$ ( $\chi^2$ )

- Calcule la significativité statistique de chaque critère avec le test du  $\chi^2$ .
- Sélectionne les restaurants probables selon un seuil ( $>0.5$ ) **et** un critère significatif.
- En cas de trop peu de résultats, complète avec les meilleurs scores.

### 6. Génération d'un fichier HTML ( Read/Write Files from Disk )

- Le contenu HTML est généré dans un script ( `jsCode` ) non totalement visible ici.
- Fichier écrit : `/home/ayan/restaurant_chinois.html`

## Aprçu du graphe dans workflow 2 .



## Conclusion

Dans se projet nous utilisons le concept de la théorie de graphe afin de bien détaillé ce que nous voulons faire . Si on affiche seulement le restaurant chinois d'Antananarivo c'est que c'est vague parce qu'elle ne prouve pas exactement la théorie qu'o'n cherche . Donc une mise en

place de graphe est permise afin d'obtenir exactement les vrais restaurants chinois d'Antananarivo et d'ou l'utilisation du workflow dans n8n .