

Test technique

Ce test comporte deux exercices : le premier consiste à développer un API, le second concerne une fonction à écrire.

Le délai pour réaliser ces deux tests est d'une semaine

1. API Météo

Contexte

Cette API permet d'obtenir la météo actuelle en fonction d'une **ville** ou de **coordonnées géographiques (latitude/longitude)**.

Prérequis

- Node.js ≥ 18
- Un gestionnaire de paquets (**npm** ou **yarn**)
- Une clé d'API d'un fournisseur météo (<https://openweathermap.org/current>)

Endpoints de l'API

1. Obtenir la météo par ville

Requête

GET /weather?city=Paris

Réponse (JSON)

```
{
  "city": "Paris",
  "temperature": 15,
  "humidity": 82,
  "description": "Ciel partiellement nuageux"
}
```

2. Obtenir la météo par latitude et longitude

Requête

GET /weather?lat=48.8566&lng=2.3522

Réponse (JSON)

```
{
  "latitude": 48.8566,
  "longitude": 2.3522,
  "temperature": 15,
  "humidity": 82,
  "description": "Ciel partiellement nuageux"
}
```

Evaluation

L'objectif premier de cet exercice est d'avoir un aperçu sur la façon dont vous allez structurer votre API. Essayez de respecter au maximum les standards de qualité de code et concentrez-vous sur les points suivants :

- La lisibilité du code ;
- L'architecture de l'application.
- Les tests automatisés.

2. Somme des nombres dans un JSON

Écrire une fonction qui parcourt une structure JSON imbriquée et retourne la somme de toutes les valeurs numériques qu'elle contient.

Exemple d'entrée:

```
{
  "a": 5,
  "b": {
    "c": 3,
    "d": {
      "e": 8,
      "f": "hello world !"
    }
  },
  "g": [1, 2, {"h": 4}]
}
```

Sortie attendue :

23

Evaluation

L'objectif premier de cet exercice est d'avoir un aperçu sur la façon dont vous allez procéder pour résoudre ce problème. Essayez de respecter au maximum les standards de qualité du code et concentrez-vous sur les points suivants:

- La lisibilité du code.
- L'algorithme utilisé.
- Les tests automatisés.

Rendu

Le rendu de ces exercices sera publié sur Github. Afin de nous aider à comprendre vos axes de réflexion, pensez à ajouter un README pour nous guider sur les points suivants :

- Les différents algorithmes et design patterns utilisés.
- Les choix techniques.
- Les pistes d'amélioration de votre code.

Bon courage !