

TP Mini app météo avec API

TP : Mini App Météo en Vanilla JS

Objectifs pédagogiques

- Manipuler les promesses en JavaScript.
 - Utiliser `async/await` et `try/catch` pour gérer le code asynchrone.
 - Lancer plusieurs requêtes en parallèle avec `Promise.all`.
 - Construire une petite application web en **vanilla JS** avec `fetch`.
 - (Partie 2) Combiner deux APIs pour enrichir l'affichage.
-

Prérequis

- Un navigateur moderne (Chrome, Firefox, Edge).
 - Un éditeur de code (VSCode, WebStorm, etc.).
 - Connaissances de base en HTML et JavaScript.
-

Partie 1 : Appeler une API météo

Étape 1 — Setup simple (30 min)

1. Crée un fichier `index.html` avec une structure de base.
2. Ajoute un champ texte et un bouton pour entrer une ville.
3. Utilise `fetch` pour appeler l'API gratuite wttr.in.

Exemple d'URL :

```
https://wttr.in/Paris?format=j1
```

4. Affiche la réponse brute (JSON) dans la page.
→ Tu dois utiliser `.then()` et `.catch()`
-

Étape 2 — Passage à `async/await` (30 min)

1. Réécris la fonction pour utiliser `async/await`.
 2. Ajoute un **bloc** `try/catch` pour gérer les erreurs réseau.
 3. Affiche des informations lisibles :
 - Ville
 - Température actuelle (°C)
 - Description de la météo
 4. (Bonus) Ajoute un **message personnalisé** (par exemple : "Prenez un parapluie" si la météo annonce de la pluie).
-

Étape 3 — Afficher la météo de plusieurs villes (45 min)

1. Permet à l'utilisateur d'entrer **plusieurs villes séparées par des virgules** (exemple : `Paris, Londres, Tokyo`).
 2. Utilise `Promise.all` pour lancer plusieurs requêtes en parallèle.
 3. Affiche les résultats sous forme de tableau ou de liste.
-

Étape 4 — (Optionnel) Interface enrichie

1. Améliore ton HTML avec un peu de mise en forme (titres, paragraphes, listes).
 2. Utilise un `<div>` ou `` pour afficher les résultats proprement.
-

Partie 2 : Ajouter une deuxième API

Étape 5 — Ajouter l'API Wikipedia

1. Utilise l'API de Wikipedia pour récupérer un court résumé de la ville.

Exemple d'URL :

```
https://en.wikipedia.org/api/rest_v1/page/summary/Paris
```

2. Après avoir affiché la météo, appelle également cette API.
3. Affiche sous la météo un court texte descriptif de la ville.

→ Ici tu combines **deux appels API en série** (`await getWeather()` , puis `await getCityInfo()`).

Étape 6 — Combiner les deux APIs avec `Promise.all` (bonus)

1. Quand tu récupères les informations d'une ville, lance en **parallèle** :
 - L'appel à l'API météo.
 - L'appel à l'API Wikipedia.
 2. Combine les résultats et affiche :
 - Nom de la ville
 - Température et description météo
 - Résumé Wikipedia
-

Ce que tu auras pratiqué

- Gestion de promesses avec `.then()` et `.catch()` .
- Simplification avec `async/await` .
- Gestion d'erreurs avec `try/catch` .
- Appels multiples en parallèle avec `Promise.all` .
- Consommation de deux APIs différentes et combinaison des résultats.
- Mise en place d'une petite application en vanilla JS.