

Agent Météo – LangChain (Option B)

Description du projet

Ce projet consiste à développer un **agent météo intelligent** en Python à l'aide de **LangChain**. L'agent est capable de répondre aux questions météo de l'utilisateur en **temps réel**, en utilisant de **vraies données météorologiques** issues d'une API publique.

Le projet s'appuie sur le **quickstart officiel de LangChain**, puis l'enrichit conformément à **l'Option B : Enrichir les outils existants**.

Prérequis

- Python **3.10 ou supérieur**
- Un compte **Anthropic (Claude)**
- Une clé API Anthropic valide

Ressources

Documentation Open-Meteo (API météo et géocodage) : <https://open-meteo.com/en/docs>

Documentation LangChain – Quickstart Python :
<https://docs.langchain.com/oss/python/langchain/quickstart>

Installation et configuration

1 Installation des dépendances

Installer LangChain et l'intégration Anthropic :

```
pip install -U langchain
pip install -U langchain-anthropic
pip install requests requests-cache retry-requests openmeteo-
requests
```

⚠ Python était déjà installé sur la machine avant le projet.

2 Configuration de la clé API

La clé API Anthropic doit être définie dans les **variables d'environnement utilisateur** :

- **Nom de la variable :**

ANTHROPIC_API_KEY

- **Valeur :**

votre_cle_api_claude

Cette configuration permet à LangChain d'accéder au modèle Claude sans inclure la clé directement dans le code (bonne pratique de sécurité).

🚀 Lancement du projet

Exécuter le fichier principal :

```
python agentmeteo.py
```

Une interface en ligne de commande s'ouvre. L'utilisateur peut poser des questions météo sur **différentes villes** jusqu'à taper `exit` pour quitter.

Fonctionnement de l'agent

◊ Prompt système

Le prompt a été adapté afin que l'agent :

- comprenne les demandes météo,
- détecte la ville mentionnée,
- utilise des outils pour récupérer des données **réelles**,
- réponde avec une touche humoristique tout en restant informative.

◊ Structures de données

Deux **dataclasses** ont été introduites pour structurer proprement les données :

LocationData

Contient les informations de localisation :

- nom de la ville
- latitude
- longitude

WeatherData

Contient les informations météo :

- température
- humidité
- vitesse du vent
- conditions météo

Le choix des **dataclasses** a été fait pour améliorer la lisibilité, la structure et la maintenabilité du code par rapport à un simple dictionnaire.

◊ Outil `get_user_location`

- Convertit le nom d'une ville en coordonnées géographiques.

- Utilise l'API de **géocodage Open-Meteo**.
- Vérifie si la ville existe.
- Extrait le **premier résultat** retourné par l'API.
- Retourne un objet LocationData.

◊ Outil get_weather_for_location

- Récupère la météo actuelle à partir des coordonnées.
- Utilise l'API **Open-Meteo** avec :
 - cache des requêtes
 - mécanisme de retry pour la fiabilité
- Retourne :
 - température
 - humidité
 - vitesse du vent
 - état du ciel

Les données sont encapsulées dans un objet WeatherData.

◊ Agent LangChain et mémoire

- Le modèle Claude est conservé tel que dans le quickstart.
- Une mémoire conversationnelle en RAM (InMemorySaver) est utilisée.
- L'agent peut répondre à **plusieurs questions successives**, sur différentes villes, dans une même session.

❖ Exemple d'utilisation

Entrée utilisateur :

What is the weather in Paris?

Sortie possible :

Agent: Looks like Paris is having a sun-sational day!

Données: Temperature: 22°C | Sunny | Humidity: 65% | Wind: 12 km/h

 Des captures d'écran de l'exécution sont disponibles en annexe.

Difficultés rencontrées et solutions

◊ Utilisation de l'API météo

- La documentation fournissait des exemples partiels.
- Il a fallu comprendre la structure des réponses API et adapter les appels REST.

◊ Géolocalisation

- La météo nécessite des coordonnées GPS.
- Une API de géocodage a donc été intégrée en amont pour convertir les villes en latitude/longitude.

◊ Structuration des données

- Le passage de chaînes de caractères simples à des données structurées a demandé une réflexion sur le design.
- Les dataclasses ont permis une solution propre et efficace.

L'utilisation d'outils comme la documentation officielle et des assistants IA (ChatGPT, Claude) a permis de débloquer certaines incompréhensions et d'approfondir la compréhension du fonctionnement des APIs.

Option choisie : Option B – Enrichir les outils existants

Justification

J'ai choisi l'Option B car elle permet de donner une **véritable utilité** à l'agent météo. Utiliser des **données réelles** rend l'agent plus pertinent et plus intéressant pour l'utilisateur qu'une simple simulation.

De plus :

- l'agent peut répondre pour **n'importe quelle ville**
- les informations retournées sont **structurées**
- l'expérience utilisateur est plus naturelle (saisie libre, boucle interactive)

Cette approche correspond davantage à un **cas d'usage réel** et exploite pleinement les capacités de LangChain.

Conclusion

Ce projet m'a permis :

- de suivre et comprendre un tutoriel LangChain,
- d'adapter un code existant,
- d'intégrer des APIs externes,
- et de concevoir un agent IA plus utile et réaliste.

Note : Le contenu de ce README a été rédigé à partir de mes propres explications et de mon travail personnel ; ChatGPT a uniquement servi à reformuler et améliorer la qualité rédactionnelle.

ANNEXE :

You voulez connaître la météo ? Demandez-moi !

You: tu vas bien parle français stp
Agent: Oui, je vais bien, merci ! Je suis "weather" ou pas (whether or not) tu veux parler français, je suis là pour t'aider ! ☺

Mais attention, je suis un expert météo qui adore les jeux de mots ! Si tu veux connaître la météo quelque part, dis-moi simplement quelle ville t'intéresse et je te donnerai un "forecast" avec beaucoup de *pun* !

Alors, est-ce que tu veux savoir le temps qu'il fait quelque part ? Je suis prêt à "rain" ou "shine" pour toi ! ☀️

You: je veux bien paris et geneve
Agent: Ah, excellente demande ! Laisse-moi te donner la météo pour ces deux villes magnifiques ! 🌈

Paris 🇫🇷
- Température : 11,3°C
- Humidité : 75%
- Vent : 13,4 km/h

C'est "Seine"-sationnel à Paris ! Avec 11 degrés, c'est parfait pour une promenade, mais attention au vent qui souffle assez fort - tu risques de perdre ton écharpe ! ❄️

Genève CH
- Température : 5,3°C
- Humidité : 88%
- Vent : 1,8 km/h

À Genève, il fait plus "swiss"-frais avec seulement 5 degrés ! L'humidité est très élevée à 88% - on pourrait dire que l'air est aussi riche que les banques suisses ! Heureusement, le vent est très calme, c'est "Geneva"-llement agréable ! 😊

Bref, Paris est plus chaud mais plus venteux, tandis que Genève est plus froide et humide. Choisis ta destination selon ton "tempéra-ment" ! 😊

Données: Paris: 11,3°C, 75% humidity, 13,4 km/h wind | Geneva: 5,3°C, 88% humidity, 1,8 km/h wind

You: []

Ln 132, Col 27 Spaces: 4 UTF-8 CRLF () Python 3.13.3

```

File Edit Selection View Go Run Terminal Help ← → ⌘ OPTION-B
EXPLORER PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Python + ×
> OPTION-B
> OUTLINE
> TIMELINE
PS C:\Users\saiss\OneDrive\Bureau\TESTCOMPETENCES\PARTIE2-ADAPTATIONETAMELIORATION\OPTION-B> & C:/Users/saiss/AppData/Local/Programs/Python/Python31\3/python.exe c:/Users/saiss/OneDrive/Bureau/TESTCOMPETENCES/PARTIE2-ADAPTATIONETAMELIORATION/OPTION-B/agentmeteo.py
Weather Agent (type 'exit' to quit)

Vous voulez connaître la météo ? Demandez-moi !

You: coucou
Agent: Well hello there! I'm your friendly weather forecaster, and I'm here to help you weather any storm! Whether you need a forecast or just want to breeze through some conversation, I'm all ears. What can I do for you today? Don't be shy - there's no pressure here! 😊

You: c'est possible d'avoir la météo à tunis
Agent: Bien sûr! Let me Tunisia in to the weather for you! ☀️ In Tunis right now, it's a cool 12.4°C - not too hot, not too cold, just Tunisia-perfect! The humidity is sitting at 67%, so it's a bit on the moist side (you could say the air is feeling a little "Tunis-ional"!). Winds are blowing at a gentle 3.7 km/h - just a light breeze, nothing to get blown away about! I hope this forecast helps you plan your day in Tunisia's capital. Stay weather-wise! ☺

Données: Temperature: 12.4°C, Humidity: 67%, Wind Speed: 3.7 km/h

You: mercii
Agent: De rien! You're very welcome! I'm always happy to help - rain or shine! If you need another forecast, just give me a shout. I'll be here, weathering the day with you! Have a wonderful time, and may the skies be ever in your favor! ☀️

You: []

```