Full stack Developer Technical Test

Challenge 1 – Weather reporting

Create a RESTful API using Python Flask that simulates a simplified weather reporting system for various cities. The API should provide the following features:

1. Add a weather report for a city (POST). A report should include:
   - City name
   - Temperature
   - Condition (e.g., Sunny, Cloudy, Rainy)
   - Timestamp of the report
2. Retrieve the latest weather report for a given city (GET).
3. Retrieve a summary of all cities currently in the system with their latest weather conditions (GET).
4. Update the latest weather report for a specific city (PUT).
5. Delete all weather reports for a specific city (DELETE).

Requirements:

- Use Python for the backend.
- Store weather report data in-memory (e.g., using Python dictionaries or lists).
- Implement proper error handling, such as city not found or invalid weather data.

Notes

- Please do not spend more than 2 hours on this. We're interested to see where you spend the time and don't expect a complete solution. If you do have extra time, feel free to add extra functionality.
- You can use any framework you like.
- Please provide a git repo with your solution.
- Whilst using AI tools to help you is totally fine, please ensure you understand the approach and be prepared to talk us through it.
- Tests are very welcome!
- Please feel free to leave some notes in your readme about what you would have tackled next.