Consuming 3rd Party API and Designing RESTful API with CRUD Operations

# *Part 1: Consuming 3rd Party API and Data Validation*

## Sign Up and Get API Key from OpenWeather API

I Created a free account on <https://openweathermap.org/api> by using my email address then created a username and password. After that I received an email from OpenWeather that contains the initial api. On the cite I am able to create a new api to use for api calls. Save the key in .env file.

A screenshot of a computer

AI-generated content may be incorrect.

## Choose the Right API Endpoint For current weather data use

Visit:   
https://openweathermap.org/current?utm\_source=chatgpt.com#geo

https://openweathermap.org/api/statistics-api#day

and copy API call from the Documentation

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

I replaced the city name London with Philadelphia and {API Key } with real API key.

## Test It in Browser Pheonix AZ’s Weather

A screenshot of a computer

AI-generated content may be incorrect.

Philadelphia PA’s Weather

A screenshot of a computer

AI-generated content may be incorrect.

## Test It in Postman

A screen shot of a computer code

AI-generated content may be incorrect.

A screenshot of a computer code

AI-generated content may be incorrect.

## 🔧 1. Install Dependencies

pip install requests pydantic pymongo python-dotenv

🌦️ 2. main.py — Fetch + Validate + Save

A screenshot of a computer

AI-generated content may be incorrect.

* It loads the API key from the environment variable OPENWEATHER\_API\_KEY using [os.getenv](vscode-file://vscode-app/c:/Users/tahir/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html).
* The API key is inserted into the [URL](vscode-file://vscode-app/c:/Users/tahir/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html) string as a query parameter:

A screenshot of a computer

AI-generated content may be incorrect.

*Part 2: Designing and Implementing RESTful API with CRUD Operations*

Build REST API (FastAPI + MongoDB)

Install Required Libraries  
pip install fastapi uvicorn

1. **Create a new main.py** file that will be your FastAPI server for the CRUD API.