# **Lab 18– API Integration: Connecting to external services with error handling**

**Name:** Suhana Rehan

**Enrollment Number**: 2503A51L36

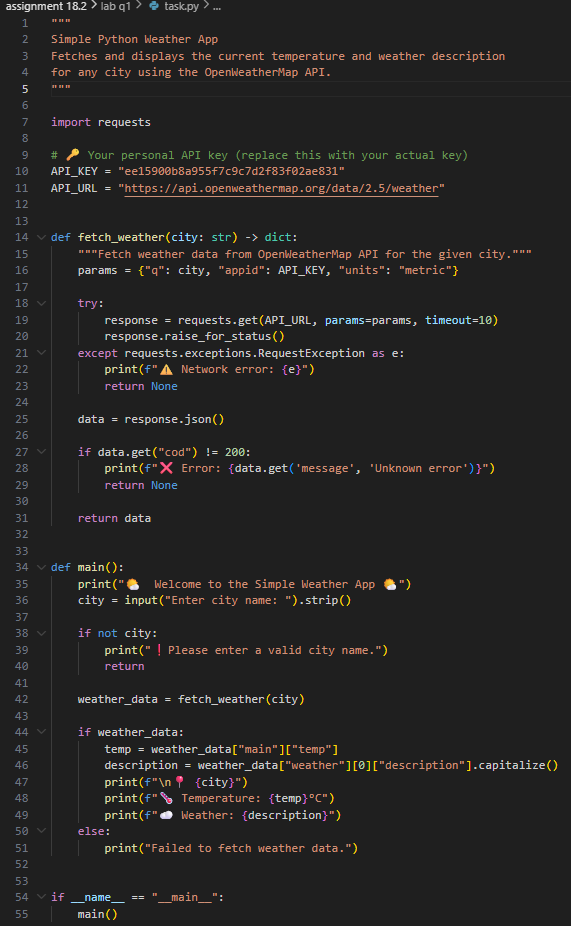
**Assignment Number:**12.2

**Lab Question 1 : Weather Forecasting API**

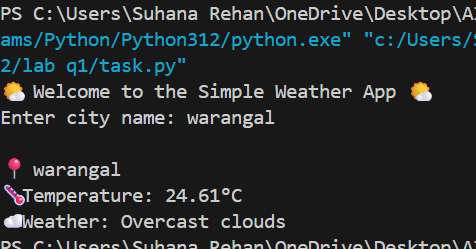
**Task 1**

**Prompt:**  
Write a Python script using the OpenWeatherMap API to fetch and display current temperature and weather description for a given city with error handling for missing or invalid API key.

**Code:**

****

**Output:**



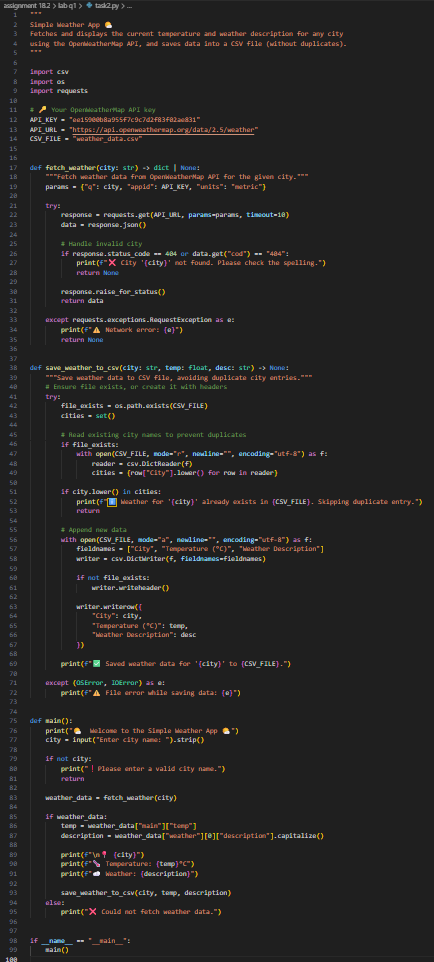
**Observations:**

* I used AI to help me understand how to send API requests using the requests module.
* It also showed me how to check if my API key is missing or wrong.
* I learned that JSON responses can be accessed like dictionaries in Python.

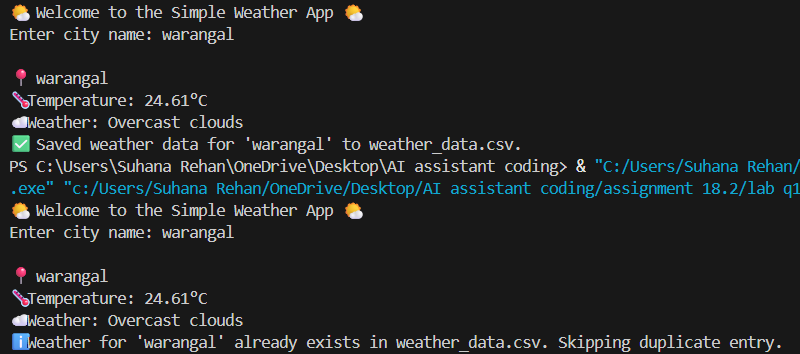
**Task 2**

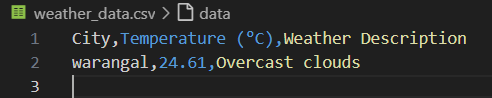
**Prompt:**  
Extend the weather script to save weather data into a CSV file without duplicates and handle file read/write errors.

**Code:**

****

**Output:**





**Observations:**

* I learned how to use csv.DictWriter to store data safely.
* The AI helped me avoid duplicate city entries by checking before writing.
* I also handled file errors using try and except blocks.

**Lab Question 2: Currency Exchange Rate API**

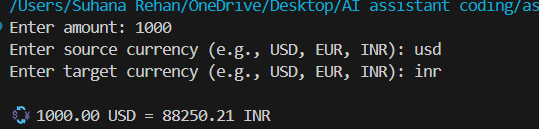
**Task 1**

**Prompt:**  
Write a Python script that takes user input (amount, source, target currency) and fetches the latest exchange rate with error handling for invalid currency codes.

**Code:**

****

**Output:**



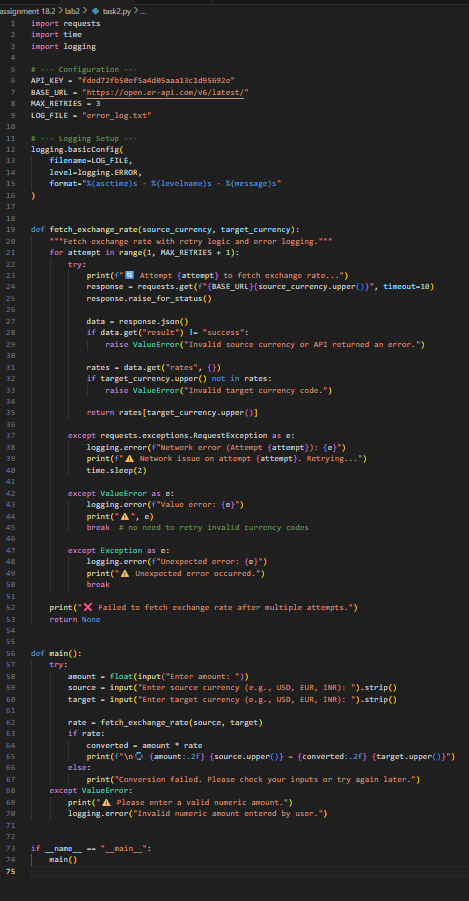
**Observations:**

* I used AI to help create input prompts and handle wrong currency codes.
* I understood how to check if the API response was valid before using it.
* The script now gives a clear error message when something goes wrong.

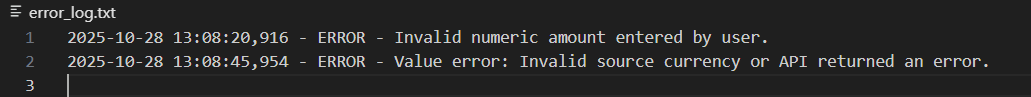
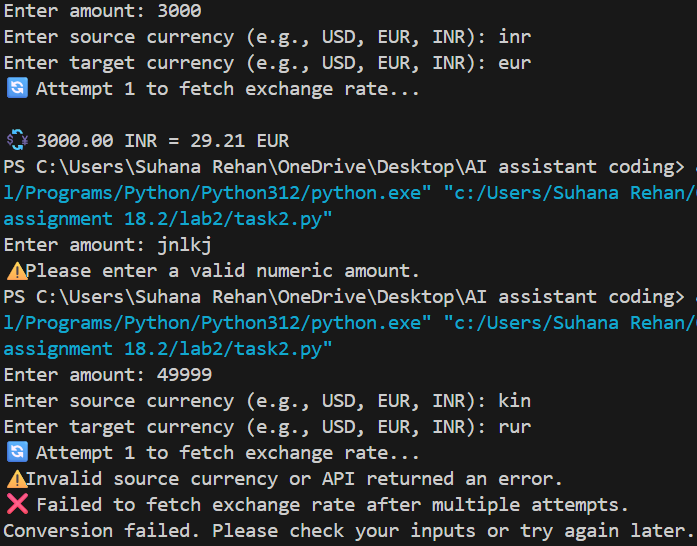
**Task 2**

**Prompt:**  
Add retry logic to the currency script to attempt the API call up to three times if it fails, and log all errors into a local file.

**Code:**

****

**Output:**



**Observations:**

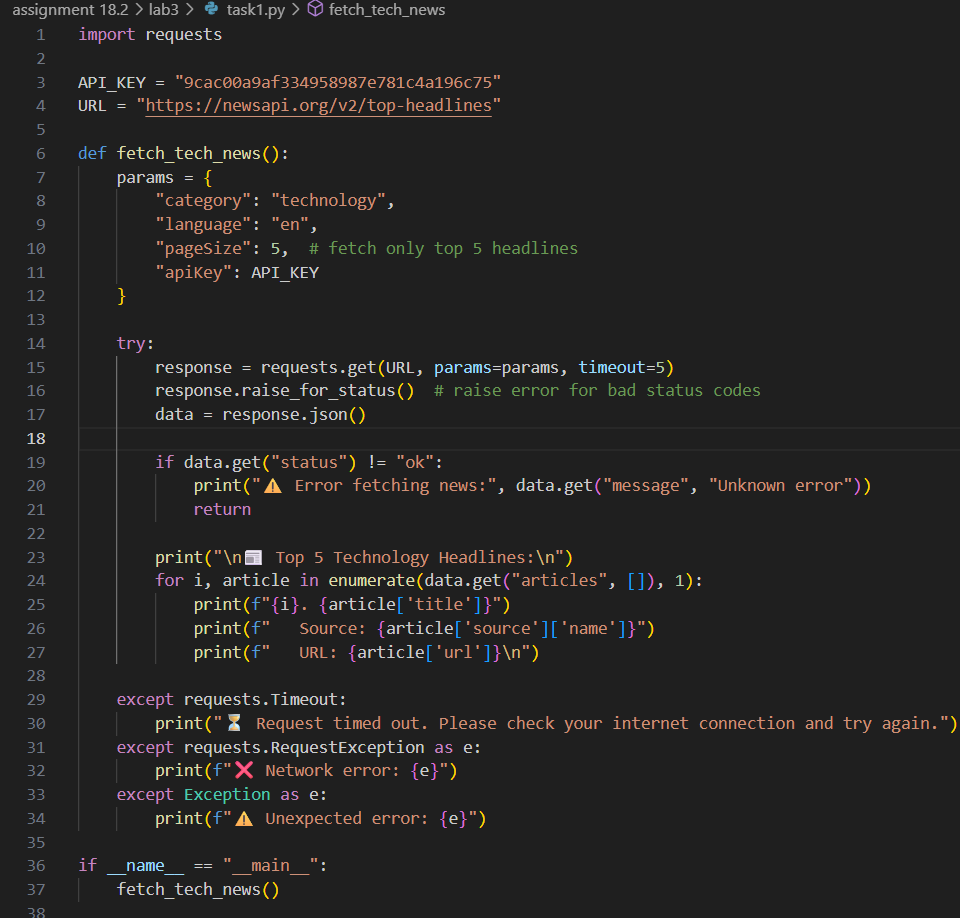
* AI showed me how to use a simple retry loop with for and try.
* I created an error log file using basic file handling.
* I learned that logging helps when APIs fail due to server issues.

**Lab Question 3 – News Headlines API**

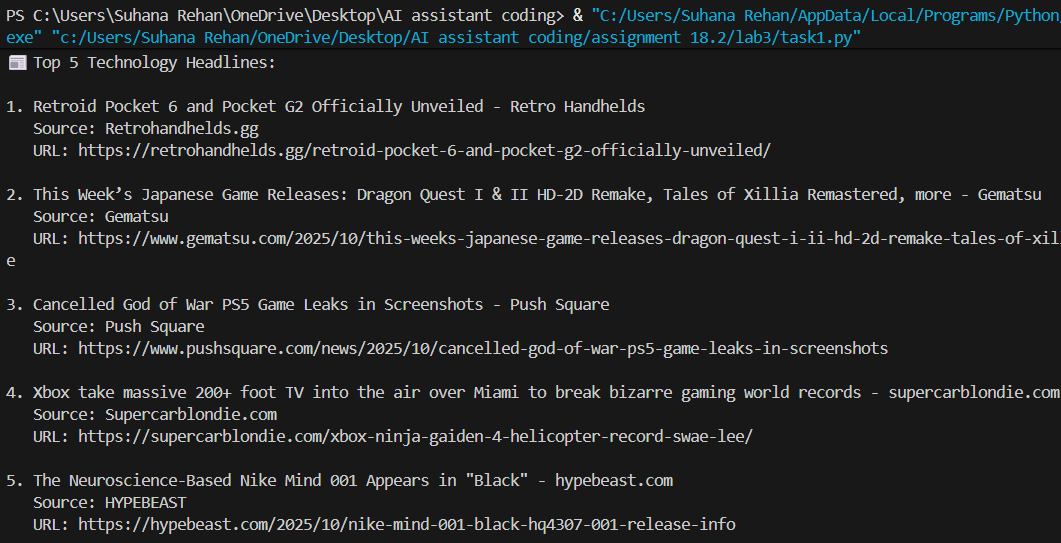
**Task 1**

**Prompt:**  
Write a Python script to fetch and print the top 5 technology headlines from a news API, with timeout error handling.

**Code:**

****

**Output:**



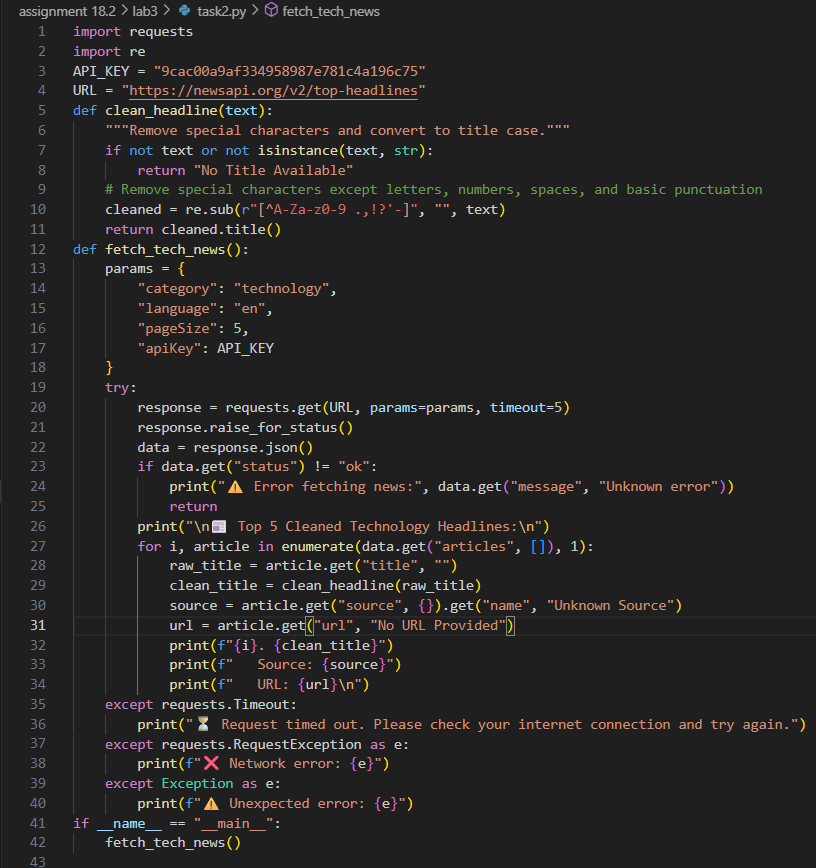
**Observations:**

* AI helped me understand how to use the timeout parameter in requests.
* The script now prints headlines clearly in the console.
* I learned how to handle slow or failed responses without crashing.

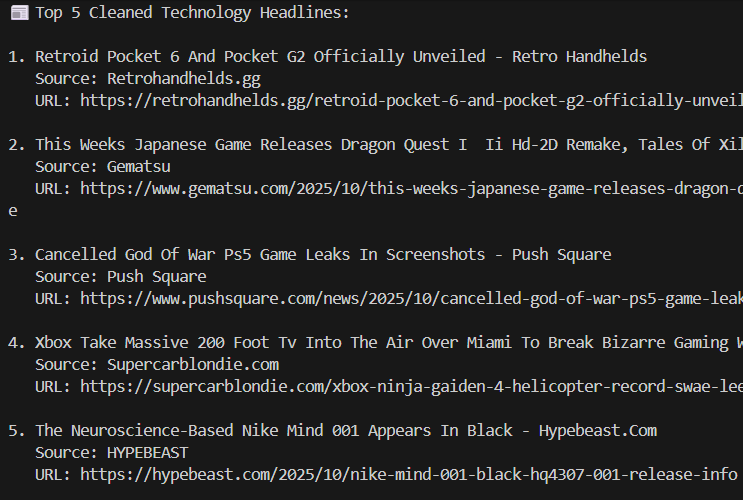
**Task 2**

**Prompt:**  
Clean and preprocess the headlines by removing special characters and converting them to title case, handling empty or null values.

**Code:**

****

**Output:**

****

**Observations:**

* I used AI to clean text using regular expressions and title() function.
* It helped me skip empty or None headlines safely.
* The cleaned headlines look much more readable now.