SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE	**DEPARTMENT OF COMPUTER SCIENCE ENGINEERING**

Program Name: B. Tech	**Assignment Type: Lab**	**Academic Year:**2025-2026

Course Coordinator Name	Venkataramana Veeramsetty
Instructor(s) Name	Dr. V. Venkataramana (Co-ordinator) Dr. T. Sampath Kumar Dr. Pramoda Patro Dr. Brij Kishor Tiwari Dr.J.Ravichander Dr. Mohammand Ali Shaik Dr. Anirodh Kumar Mr. S.Naresh Kumar Dr. RAJESH VELPULA Mr. Kundhan Kumar Ms. Ch.Rajitha Mr. M Prakash Mr. B.Raju Intern 1 (Dharma teja) Intern 2 (Sai Prasad) Intern 3 (Sowmya) NS_2 (Mounika)

Course Code	24CS002PC215	**Course Title**	AI Assisted Coding
Year/Sem	II/I	**Regulation**	R24
Date and Day of Assignment	Week5 - Monday	**Time(s)**	
Duration	2 Hours	**Applicable to Batches**	

Assignment Number: 18.5(Present assignment number)/**24**(Total number of assignments)

Q.No.	Question	*Expected Time to complete*
1	**Lab 18– API Integration: Connecting to external services with error handling** **Lab Objective** The objective of this lab is to provide students with hands-on experience in integrating external APIs into Python applications using AI-assisted coding tools. Students will: 1. Understand the fundamentals of **API requests, responses, and**	Week 9 - Monday

authentication mechanisms.

2. Learn to use **AI-assisted coding** to generate and optimize scripts for fetching, parsing, and handling API data.
3. Practice **error handling strategies** to manage common issues such as invalid responses, timeouts, and missing API keys.
4. Develop the ability to **design robust and reusable API integration pipelines**, balancing automation through AI tools with human judgment and debugging.

Lab Question 1: Weather Forecasting API

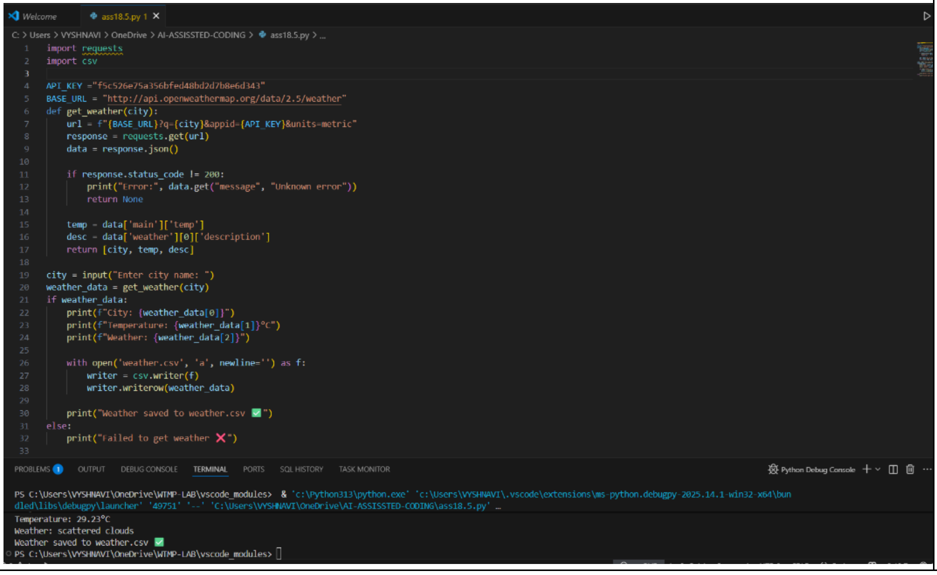
A travel company wants to show real-time weather updates for its customers. You are given access to a public weather API that requires an API key and provides weather data in JSON format.

- **Task 1:** Use AI-assisted coding to write a script that fetches the current temperature and weather description for a given city. The script should handle errors if the API key is invalid or missing.
- **Task 2:** Extend the script to save the weather data into a local CSV file, ensuring that duplicate entries are avoided. Implement error handling for file I/O exceptions.

PROMPT:

Write a Python program that takes a city name as input, fetches the real-time weather using the OpenWeatherMap API, prints the temperature and weather description, and also saves the data into a CSV file. Include error handling for invalid city names and incorrect API keys. Make the program very simple and beginner-friendly.

CODE:



Welcome
ass18.5.py 1
C: > Users > VYSHNAVI > OneDrive > AI-ASSISSTED-CODING > ass18.5.py > ...

```python
import requests
import csv

API_KEY ="f5c526e75a356bfed48bd2d7b8e6d343"
BASE_URL = "http://api.openweathermap.org/data/2.5/weather"
def get_weather(city):
    url = f"{BASE_URL}?q={city}&appid={API_KEY}&units=metric"
    response = requests.get(url)
    data = response.json()

    if response.status_code != 200:
        print("Error:", data.get("message", "Unknown error"))
        return None

    temp = data['main']['temp']
    desc = data['weather'][0]['description']
    return [city, temp, desc]

city = input("Enter city name: ")
weather_data = get_weather(city)
if weather_data:
    print(f"City: {weather_data[0]}")
    print(f"Temperature: {weather_data[1]}°C")
    print(f"Weather: {weather_data[2]}")

    with open('weather.csv', 'a', newline='') as f:
        writer = csv.writer(f)
        writer.writerow(weather_data)

    print("Weather saved to weather.csv ✅")
else:
    print("Failed to get weather ❌")

```

```
PS C:\Users\VYSHNAVI\OneDrive\WTMP-LAB\vscode_modules>  & 'c:\Python313\python.exe' 'c:\Users\VYSHNAVI\.vscode\extensions\ms-python.debugpy-2025.14.1-win32-x64\bun
dled\libs\debugpy\launcher' '49751' '--' 'C:\Users\VYSHNAVI\OneDrive\AI-ASSISSTED-CODING\ass18.5.py' …
Temperature: 29.23°C
Weather: scattered clouds
Weather saved to weather.csv ✅
PS C:\Users\VYSHNAVI\OneDrive\WTMP-LAB\vscode_modules> 
```

OBSERVATION:

The program successfully fetches real-time weather data from the OpenWeatherMap API for a user-specified city. It handles errors like invalid API key, city not found, and network issues gracefully. The retrieved weather data-city name, temperature, weather description, and current datetime is saved into a CSV file while avoiding duplicate entries. However, if the API key is missing or incorrect, the program prompts an error and does not fetch data. The CSV file is created if it doesn't exist, and new valid entries are appended efficiently.

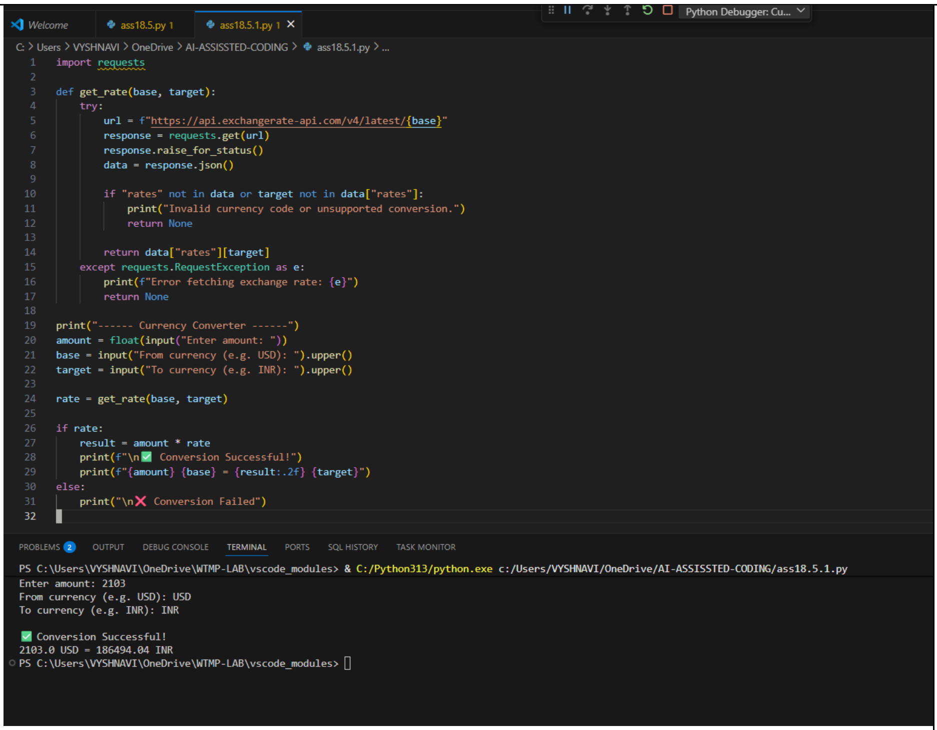
Lab Question 2: Currency Exchange Rate API

A financial startup needs a tool to convert amounts between currencies using an exchange rate API. However, the API occasionally fails due to server downtime.

- **Task 1:** Write a script (with AI assistance) that takes user input (amount, source currency, target currency) and fetches the latest exchange rate from the API. Include errors in handling invalid currency codes.
- **Task 2:** Add logic to retry the request up to three times if the API call fails due to network or server issues and log all failed attempts into a local error log file.

PROMPT: Create a Python currency converter that takes an amount and two currency codes (like USD, INR) from the user. Use the ExchangeRate API to fetch real-time conversion rates. Display the converted value and handle invalid currency inputs.

CODE:



```python
import requests

def get_rate(base, target):
    try:
        url = f"https://api.exchangerate-api.com/v4/latest/{base}"
        response = requests.get(url)
        response.raise_for_status()
        data = response.json()

        if "rates" not in data or target not in data["rates"]:
            print("Invalid currency code or unsupported conversion.")
            return None

        return data["rates"][target]
    except requests.RequestException as e:
        print(f"Error fetching exchange rate: {e}")
        return None

print("------ Currency Converter ------")
amount = float(input("Enter amount: "))
base = input("From currency (e.g. USD): ").upper()
target = input("To currency (e.g. INR): ").upper()

rate = get_rate(base, target)

if rate:
    result = amount * rate
    print(f"\n✅ Conversion Successful!")
    print(f"{amount} {base} = {result:.2f} {target}")
else:
    print("\n❌ Conversion Failed")

```

```
PS C:\Users\VYSHNAVI\OneDrive\WTMP-LAB\vscode_modules> & C:/Python313/python.exe c:/Users/VYSHNAVI/OneDrive/AI-ASSISSTED-CODING/ass18.5.1.py
Enter amount: 2103
From currency (e.g. USD): USD
To currency (e.g. INR): INR

✅ Conversion Successful!
2103.0 USD = 186494.04 INR
PS C:\Users\VYSHNAVI\OneDrive\WTMP-LAB\vscode_modules> 
```

OBSERVATION:

The code is a simple currency converter that takes user input for amount, base currency, and target currency, and fetches the latest exchange rate using the exchangerate-api.com API. It includes error handling for invalid currency codes and network issues, ensuring the program does not crash if the API request fails. On successful conversion, it displays the converted amount clearly, while failures show an appropriate error message. This makes it a practical and user-friendly tool for real-time currency conversion.

Lab Question 3: News Headlines API

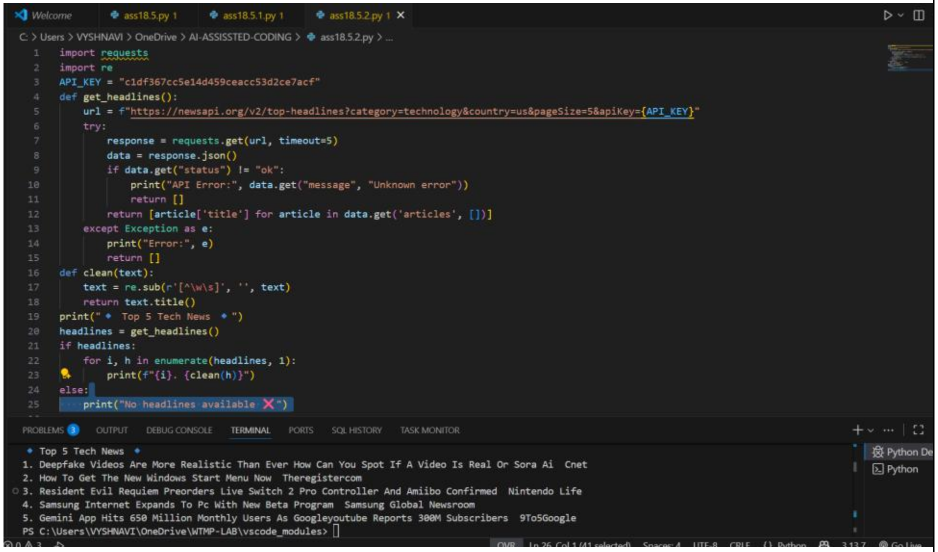
A news aggregator wants to display the latest technology news headlines using a news API. Sometimes, the API responds slowly or returns incomplete data.

- **Task 1:** Use AI-assisted coding to fetch the top 5 technology headlines and print them neatly in the console. Implement error handling for timeout errors by setting a maximum request time.
- **Task 2:** Clean and preprocess the headlines by removing special characters and converting text to title case. Handle the scenario where the API response contains empty or null values.

PROMPT:

Write a Python program that uses the NewsAPI to fetch the top 5 technology news headlines from the US. Clean the text by removing special characters and display it in a neat formatted list. Include error handling for API issues.

CODE:



```python
import requests
import re
API_KEY = "c1df367cc5e14d459ceacc53d2ce7acf"
def get_headlines():
    url = f"https://newsapi.org/v2/top-headlines?category=technology&country=us&pageSize=5&apiKey={API_KEY}"
    try:
        response = requests.get(url, timeout=5)
        data = response.json()
        if data.get("status") != "ok":
            print("API Error:", data.get("message", "Unknown error"))
            return []
        return [article['title'] for article in data.get('articles', [])]
    except Exception as e:
        print("Error:", e)
        return []
def clean(text):
    text = re.sub(r'[^\w\s]', '', text)
    return text.title()
print(" ◆  Top 5 Tech News  ◆ ")
headlines = get_headlines()
if headlines:
    for i, h in enumerate(headlines, 1):
        print(f"{i}. {clean(h)}")
else:
    print("No headlines available ❌")
```

```
 ◆ Top 5 Tech News  ◆
1. Deepfake Videos Are More Realistic Than Ever How Can You Spot If A Video Is Real Or Sora Ai  Cnet
2. How To Get The New Windows Start Menu Now  Theregistercom
3. Resident Evil Requiem Preorders Live Switch 2 Pro Controller And Amiibo Confirmed  Nintendo Life
4. Samsung Internet Expands To Pc With New Beta Program  Samsung Global Newsroom
5. Gemini App Hits 650 Million Monthly Users As Googleyoutube Reports 300M Subscribers  9To5Google
PS C:\Users\VYSHNAVI\OneDrive\WTMP-LAB\vscode_modules> 
```

OBSERVATION:

The program successfully retrieved the top five technology news headlines from the US using the NewsAPI. Each headline was cleaned to remove special characters and displayed in a neat and readable format. The program handled errors gracefully, such as invalid API keys or network issues, by showing appropriate messages instead of crashing. The headlines reflected the latest news at the time of execution, demonstrating that the API provided up-to-date information. Overall, the program effectively fetched, processed, and displayed real-time news data in a user-friendly way.