# AIPP LAB 15

**Name: MOHAMMED NIZAMUDDIN**

**Hall Ticket No.: 2503B05144**

**Date: 25/11/2025 (Week 5)**

---

**Completed this Assignment using VS CODE AI Integration.**

## Task 1:

**Ask Use Python (or Node.js/JavaScript) to connect to a public API (e.g., OpenWeatherMap or JSONPlaceholder). Send a simple GET request to retrieve data. Display the response in a readable format (pretty JSON). Prompt: write a python function that display weather details of a city using weather api without error handling. Display weather details as JSON output.**

***CODE:***

TASK15.1.py 1

C: > Users > ACER > OneDrive > Documents > MAHVISH M.TECH AIPP LABS > ASSIGNMENT 15 >

```python

import requests
import json
from datetime import datetime
API_KEY = "e6c6083509fc4d450cde0ca4414b3a9f"
BASE_URL = "https://api.openweathermap.org/data/2.5/weather"
def get_weather(city):
    """
    Fetch weather details for a city using OpenWeather API.
    Includes error handling, JSON display and file storage.
    """
    params = {
        "q": city,
        "appid": API_KEY,
        "units": "metric"
    }
    try:
        response = requests.get(BASE_URL, params=params, timeout=5)
        # Raise error if HTTP response is not OK
        response.raise_for_status()
        data = response.json()
        # Extract fields
        temperature = data["main"]["temp"]
        humidity = data["main"]["humidity"]
        weather_desc = data["weather"][0]["description"].title()
        # User-friendly output for Task 3 & 4
        print("\n===== WEATHER REPORT =====")
        print(f"City: {city.title()}")
        print(f"Temperature: {temperature}°C")
        print(f"Humidity: {humidity}%")
        print(f"Weather: {weather_desc}")
        # Append results to results.txt (Task 5)
        entry = {
            "city": city.title(),
            "temperature": temperature,
            "humidity": humidity,
            "weather": weather_desc,
```

TASK15.1.py 1

C: > Users > ACER > OneDrive > Documents > MAHVISH M.TECH AIPP LABS > ASSIGNMENT 15 > TASK15.1.p

```
    def get_weather(city):
                    "weather": weather_desc,
                    "time": str(datetime.now())
                }
                with open("results.txt", "a") as file:
                    file.write(json.dumps(entry) + "\n")
                return entry    # For test cases
            except requests.exceptions.Timeout:
                print("Error: Request timed out. Check your network.")
            except requests.exceptions.ConnectionError:
                print("Error: Could not connect to API. Check internet connection.")
            except requests.exceptions.HTTPError:
                print("Error: Invalid city name or API key.")
            except Exception as e:
                print("Unexpected error:", str(e))
            return None
    # ✅ CALL THE FUNCTION HERE
    get_weather("Hyderabad")    # ← required!
    # ✅ CALL THE FUNCTION HERE
    get_weather("Delhi")    # ← required!
    # ✅ CALL THE FUNCTION HERE
    get_weather("Bangalore")    # ← required!
```

## *OUTPUT:*

```
===== FULL API JSON RESPONSE =====
{
    "coord": {
        "lon": 78.4744,
        "lat": 17.3753
    },
    "weather": [
        {
            "id": 721,
            "main": "Haze",
            "description": "haze",
            "icon": "50n"
        }
    ],
    "base": "stations",
    "main": {
        "temp": 23.23,
        "feels_like": 23.38,
        "temp_min": 21.73,
        "temp_max": 23.23,
        "pressure": 1016,
        "humidity": 68,
        "sea_level": 1016,
        "grnd_level": 952
    },
    "visibility": 2500,
    "wind": {
        "speed": 2.06,
        "deg": 110
    },
    "clouds": {
        "all": 40
    },
    "dt": 1764083372,
    "sys": {
        "type": 1,
        "id": 9214,
        "country": "IN",
            "sunrise": 1764032203,
            "sunset": 1764072586
        },
        "timezone": 19800,
        "id": 1269843,
        "name": "Hyderabad",
        "cod": 200
}

===== WEATHER REPORT =====
City: Hyderabad
Temperature: 23.23°C
Humidity: 68%
Weather: Haze
PS C:\Users\ACER\OneDrive\Documents\MAHVISH M.TECH AIPP LABS\ASSIGNMENT 15>
```

## Task 2:

**Prompt: write a python function that display weather details of a city using weather api with error handling. Display weather details as JSON output.**

***CODE:***

```
import requests # type: ignore
import json
API_KEY = "e6c6083509fc4d450cde0ca4414b3f"
def get_weather_with_errors(city):
    url = f"https://api.openweathermap.org/data/2.5/weather?q={city}&appid={API_KEY}&units=metric"
    try:
        response = requests.get(url, timeout=5)
        response.raise_for_status()
        data = response.json()

        print(json.dumps(data, indent=4))
        return data
    except requests.exceptions.Timeout:
        print("Error: API request timed out.")
    except requests.exceptions.ConnectionError:
        print("Error: Could not connect to API. Check your internet.")
    except requests.exceptions.HTTPError:
        print("Error: Invalid city or API key.")
    except Exception as e:
        print("Unexpected Error:", str(e))
    return None
get_weather_with_errors("Hyderabad")
```

***OUTPUT:***

```
PS C:\Users\ACER\OneDrive\Documents\MAHVISH M.TECH AIPP LABS\ASSIGNMENT 15>  c:; cd 'c:\Users
rs\ACER\AppData\Local\Microsoft\WindowsApps\python3.12.exe' 'c:\Users\ACER\.vscode\extensions
0' '--' 'C:\Users\ACER\OneDrive\Documents\MAHVISH M.TECH AIPP LABS\ASSIGNMENT 15\TASK15.2.py'
Error: Invalid city or API key.
PS C:\Users\ACER\OneDrive\Documents\MAHVISH M.TECH AIPP LABS\ASSIGNMENT 15>
```

## Task 3:

**Extract and Display Specific Data**

**• Instructions:**

**1. From the API response (e.g., weather API), extract specific fields (temperature, humidity, description).**

**2. Display them in a user-friendly format (not raw JSON)**

***CODE:***

: > Users > ACER > OneDrive > Documents > MAHVISH M.TECH AIPP LABS > ASSIGNMENT 15 > TASK15.3.py > get_weather_with_erro

```python
import json
API_KEY = "e6c6083509fc4d450cde0ca4414b3a9f"
def get_weather_with_errors(city):
    url = f"https://api.openweathermap.org/data/2.5/weather?q={city}&appid={API_KEY}&units=metric"
    try:
        response = requests.get(url, timeout=5)
        response.raise_for_status()
        data = response.json()
        print(json.dumps(data, indent=4))
        return data
    except requests.exceptions.Timeout:
        print("Error: API request timed out.")
    except requests.exceptions.ConnectionError:
        print("Error: Could not connect to API. Check your internet.")
    except requests.exceptions.HTTPError:
        print("Error: Invalid city or API key.")
    except Exception as e:
        print("Unexpected Error:", str(e))
    return None
get_weather_with_errors("Hyderabad")
def get_weather_pretty(city):
    data = get_weather_with_errors(city)
    if data is None:
        return None
    city_name = data["name"]
    temp = data["main"]["temp"]
    hum = data["main"]["humidity"]
    desc = data["weather"][0]["description"].title()
    print(f"City: {city_name}")
    print(f"Temperature: {temp}°C")
    print(f"Humidity: {hum}%")
    print(f"Weather: {desc}")
    return {"city": city_name, "temp": temp, "humidity": hum, "weather": desc}
res = get_weather_pretty("London")
assert "city" in res
assert "temp" in res
assert "weather" in res
```

# *OUTPUT:*

```
PS C:\Users\ACER\OneDrive\Documents\MAHVISH M.TECH AIPP LABS\ASSIGNMENT
rs\ACER\AppData\Local\Microsoft\WindowsApps\python3.12.exe' 'c:\Users\A
8' '--' 'C:\Users\ACER\OneDrive\Documents\MAHVISH M.TECH AIPP LABS\ASSI
{
    "coord": {
        "lon": 78.4744,
        "lat": 17.3753
    },
    "weather": [
        {
            "id": 721,
            "main": "Haze",
            "description": "haze",
            "icon": "50n"
        }
    ],
    "base": "stations",
    "main": {
        "temp": 21.23,
        "feels_like": 21.45,
        "temp_min": 21.23,
        "temp_max": 21.73,
        "pressure": 1016,
        "humidity": 78,
        "sea_level": 1016,
        "grnd_level": 952
    },
    "visibility": 3000,
    "wind": {
        "speed": 2.06,
        "deg": 110
    },
    "clouds": {
        "all": 40
    },
    "dt": 1764084378,
    "sys": {
        "type": 1,
        "id": 9214,
        "country": "IN",
        "sunrise": 1764032203,
        "sunset": 1764072586
    },
    "timezone": 19800,
    "id": 1269843,
    "name": "Hyderabad",
    "cod": 200
}
{
    "coord": {
        "lon": -0.1257,
        "lat": 51.5085
    },
    "weather": [
        {
            "id": 803,
            "main": "Clouds",
            "description": "broken clouds",
            "icon": "04d"
        }
    ],
    "base": "stations",
    "main": {
        "temp": 6.58,
        "feels_like": 4.02,
        "temp_min": 5.95,
        "temp_max": 7.65,
        "pressure": 1015,
        "humidity": 81,
        "sea_level": 1015,
        "grnd_level": 1011
    },
    "visibility": 10000,
    "wind": {
        "speed": 3.6,
```

```
        "deg": 300
    },
    "clouds": {
        "all": 75
    },
    "dt": 1764084132,
    "sys": {
        "type": 2,
        "id": 2075535,
        "country": "GB",
        "sunrise": 1764056116,
        "sunset": 1764086409
    },
    "timezone": 0,
    "id": 2643743,
    "name": "London",
    "cod": 200
}
City: London
Temperature: 6.58°C
Humidity: 81%
Weather: Broken Clouds
PS C:\Users\ACER\OneDrive\Documents\MAHVISH M.TECH AIPP LABS\ASSIGNMENT 15>
```

# Task 4:

**Build a Function with a Parameter. Write a python function that displays weather details of a city using weather api with error handling. Display weather details in user friendly forma**

***CODE:***

> Users > ACER > OneDrive > Documents > MAHVISH M.TECH AIPP LABS > ASSIGNMENT 15 > TASK15.4.py > fetch_weather

```python
import requests # type: ignore
import json
API_KEY = "e6c6083509fc4d450cde0ca4414b3a9f"
def get_weather_with_errors(city):
    url = f"https://api.openweathermap.org/data/2.5/weather?q={city}&appid={API_KEY}&units=metric"
    try:
        response = requests.get(url, timeout=5)
        response.raise_for_status()
        data = response.json()
        print(json.dumps(data, indent=4))
        return data
    except requests.exceptions.Timeout:
        print("Error: API request timed out.")
    except requests.exceptions.ConnectionError:
        print("Error: Could not connect to API. Check your internet.")
    except requests.exceptions.HTTPError:
        print("Error: Invalid city or API key.")
    except Exception as e:
        print("Unexpected Error:", str(e))
    return None
def fetch_weather(city):
    data = get_weather_with_errors(city)
    if data is None:
        print("Error: City not found. Please enter a valid city.")
        return None
    result = {
        "city": data["name"],
        "temp": data["main"]["temp"],
        "humidity": data["main"]["humidity"],
        "weather": data["weather"][0]["description"].title()
    }
    print(result)
    return result
assert fetch_weather("New York")["city"] == "New York"
assert fetch_weather("xyz123") is None
assert isinstance(fetch_weather("Delhi"), dict)
```

## *OUTPUT:*


```
{
    "coord": {
        "lon": 77.2167,
        "lat": 28.6667
    },
    "weather": [
        {
            "id": 721,
            "main": "Haze",
            "description": "haze",
            "icon": "50n"
        }
    ],
    "base": "stations",
    "main": {
        "temp": 17.05,
        "feels_like": 16.35,
        "temp_min": 17.05,
        "temp_max": 17.05,
        "pressure": 1017,
        "humidity": 59,
        "sea_level": 1017,
        "grnd_level": 991
    },
    "visibility": 2200,
    "wind": {
        "speed": 1.03,
        "deg": 240
    },
    "clouds": {
        "all": 0
    },
    "dt": 1764084575,
    "sys": {
        "type": 1,
        "id": 9165,
        "country": "IN",
        "sunrise": 1764033716,
        "sunset": 1764071677
    },
    "timezone": 19800,
    "id": 1273294,
    "name": "Delhi",
    "cod": 200
}
{'city': 'Delhi', 'temp': 17.05, 'humidity': 59, 'weather': 'Haze'}
PS C:\Users\ACER\OneDrive\Documents\MAHVISH M.TECH AIPP LABS\ASSIGNMENT 15>
```


## <u>Task 5</u>:

**Prompt: write a python function that display weather details of a city using weather api with error handling. Display weather details as JSON output. Store the weather details in current directory as text file, every run output will append.**

## *CODE:*

TASK15.1.py | TASK15.2.py | TASK15.3.py | TASK15.4.py | TASK15.5.py

C: > Users > ACER > OneDrive > Documents > MAHVISH M.TECH AIPP LABS > ASSIGNMENT 15 > TASK15.5.py > save_weather

```python
import requests # type: ignore
import json
API_KEY = "e6c6083509fc4d450cde0ca4414b3a9f"
def get_weather_with_errors(city):
    url = f"https://api.openweathermap.org/data/2.5/weather?q={city}&appid={API_KEY}&units=metric"
    try:
        response = requests.get(url, timeout=5)
        response.raise_for_status()
        data = response.json()
        print(json.dumps(data, indent=4))
        return data
    except requests.exceptions.Timeout:
        print("Error: API request timed out.")
    except requests.exceptions.ConnectionError:
        print("Error: Could not connect to API. Check your internet.")
    except requests.exceptions.HTTPError:
        print("Error: Invalid city or API key.")
    except Exception as e:
        print("Unexpected Error:", str(e))
    return None
def fetch_weather(city):
    data = get_weather_with_errors(city)
    if data is None:
        print("Error: City not found. Please enter a valid city.")
        return None
    result = {
        "city": data["name"],
        "temp": data["main"]["temp"],
        "humidity": data["main"]["humidity"],
        "weather": data["weather"][0]["description"].title()
    }
    return result
def save_weather(city):
    result = fetch_weather(city)
    if result is None:
        return None
    # Append to file
    # Append to file
    with open("results.txt", "a") as f:
        f.write(json.dumps(result) + "\n")
    print("Weather details saved to results.txt")
    return result
before = len(open("results.txt").readlines())
save_weather("London")
after = len(open("results.txt").readlines())
assert after == before + 1
assert save_weather("xyz123") is None
assert "city" in save_weather("Delhi")
```

***OUTPUT:***

```
6' '--' 'C:\Users\ACER\OneDrive\Documents\MAHVISH M.TECH AIPP LABS\ASSIGNMENT 15\TASK15.5.py'
{
    "coord": {
        "lon": -0.1257,
        "lat": 51.5085
    },
    "weather": [
        {
            "id": 803,
            "main": "Clouds",
            "description": "broken clouds",
            "icon": "04d"
        }
    ],
    "base": "stations",
    "main": {
        "temp": 6.52,
        "feels_like": 4.65,
        "temp_min": 5.84,
        "temp_max": 7.17,
        "pressure": 1016,
        "humidity": 81,
        "sea_level": 1016,
        "grnd_level": 1012
    },
    "visibility": 10000,
    "wind": {
        "speed": 2.57,
        "deg": 310
    },
    "clouds": {
        "all": 75
    },
    "dt": 1764085385,
    "sys": {
        "type": 2,
        "id": 2075535,
        "country": "GB",
```

```
        "sunrise": 1764056116,
        "sunset": 1764086409
    },
    "timezone": 0,
    "id": 2643743,
    "name": "London",
    "cod": 200
}
Weather details saved to results.txt
Error: Invalid city or API key.
Error: City not found. Please enter a valid city.
{
    "coord": {
        "lon": 77.2167,
        "lat": 28.6667
    },
    "weather": [
        {
            "id": 721,
            "main": "Haze",
            "description": "haze",
            "icon": "50n"
        }
    ],
    "base": "stations",
    "main": {
        "temp": 17.05,
        "feels_like": 16.35,
        "temp_min": 17.05,
        "temp_max": 17.05,
        "pressure": 1017,
        "humidity": 59,
        "sea_level": 1017,
        "grnd_level": 991
    },
    "visibility": 2200,
    "wind": {
        "speed": 1.03,
```

```
            "deg": 240
        },
        "clouds": {
            "all": 0
        },
        "dt": 1764085205,
        "sys": {
            "type": 1,
            "id": 9165,
            "country": "IN",
            "sunrise": 1764033716,
            "sunset": 1764071677
        },
        "timezone": 19800,
        "id": 1273294,
        "name": "Delhi",
        "cod": 200
}
Weather details saved to results.txt
PS C:\Users\ACER\OneDrive\Documents\MAHVISH M.TECH AIPP LABS\ASSIGNMENT 15>
```