Python Code

Team Members:

M. Sanjay-AP23110010185

K. Samveeth_AP23110010176

T. Abhinav-AP23110010228

MD.Waseem-AP23110010174

CODE:

import requests import os import datetime

def fetch_weather_data(city_name, api_key):

base_url = "https://api.openweathermap.org/data/2.5/weather"

params = { "q": city_name, "appid": api_key, "units": "metric" }

```
try:
    response = requests.get(base_url, params=params, timeout=10)
    response.raise_for_status()
    return response.json()
except requests.exceptions.HTTPError as http_err:
    print(f"HTTP error occurred for {city_name}: {http_err}")
except requests.exceptions.ConnectionError:
    print("Connection error. Please check your internet connection.")
except requests.exceptions.Timeout:
    print("Request timed out.")
except requests.exceptions.RequestException as e:
    print("An error occurred:", e)

return None
```

def parse_weather_data(data):

if data is None or data.get("cod") != 200:

return None

```
weather_info = {
    "description": data["weather"][0]["description"],
    "temperature": data["main"]["temp"],
    "feels_like": data["main"]["feels_like"],
    "temp_min": data["main"]["temp_min"],
    "temp_max": data["main"]["temp_max"],
    "humidity": data["main"]["humidity"],
    "wind_speed": data["wind"]["speed"],
    "city": data["name"],
    "country": data["sys"]["country"],
    "timestamp":
datetime.datetime.utcfromtimestamp(data["dt"]).strftime('%Y-%m-%d %H:%
M:%S')
}
return weather_info
```

def display_weather(info): if info is None: print("Could not display weather information.") return

```
print(f"\nWeather in {info['city']}, {info['country']} at
{info['timestamp']} UTC:")
print(f"Condition   : {info['description'].capitalize()}")
print(f"Temperature : {info['temperature']}°C (Feels like:
{info['feels_like']}°C)")
print(f"Min/Max Temp: {info['temp_min']}°C / {info['temp_max']}°C")
print(f"Humidity    : {info['humidity']}%")
print(f"Wind Speed  : {info['wind_speed']} m/s")
```

def log_to_file(info, filename="weather_log.txt"): if info is None: return

```
with open(filename, "a") as file:
    file.write(f"Weather in {info['city']}, {info['country']} at
{info['timestamp']} UTC:\n")
    file.write(f"Condition   : {info['description'].capitalize()}\n")
    file.write(f"Temperature : {info['temperature']}°C (Feels like:
```

```
{info['feels_like']}°C)\n")
    file.write(f"Min/Max Temp: {info['temp_min']}°C /
{info['temp_max']}°C\n")
    file.write(f"Humidity    : {info['humidity']}%\n")
    file.write(f"Wind Speed  : {info['wind_speed']} m/s\n")
    file.write("-" * 40 + "\n")
```

def main(): api_key = "484bdd54ba5296ef242100b0de56a4a0" # Replace this with your real key

```
print("Welcome to the Weather App!")
cities = []

while True:
    city = input("Enter a city name (or type 'done' to finish):
").strip()
    if city.lower() == "done":
        break
    elif city:
        cities.append(city)

if not cities:
    print("No cities entered. Exiting.")
    return

for city in cities:
    print(f"\nFetching weather for {city}...")
    data = fetch_weather_data(city, api_key)
    weather_info = parse_weather_data(data)
    display_weather(weather_info)
    log_to_file(weather_info)

print("\nWeather information retrieved and saved to log (if
enabled).")
```

if **name** == "**main**": main()