**The following demonstrate the suggestions provided by GitHub Copilot.**

1. Error handling: GitHub Copilot suggests using a **try-except** block to catch potential exceptions when making the API request. This helps to handle possible network errors or issues with the API endpoint.
2. Error message: In case an exception occurs during the API request, GitHub Copilot suggests printing an informative error message that includes the exception details. This helps in providing useful feedback to the user.
3. Data parsing: GitHub Copilot provides suggestions for parsing the response JSON data. It suggests using the **.get()** method to safely access nested dictionary values and provides default values in case the data is missing or not in the expected format.
4. Printing weather information: GitHub Copilot suggests using conditional statements to print the temperature and weather description only if the data is available. This improves the code's readability and handles cases where certain data fields may be missing from the API response.

import requests

API\_KEY = "b2486a8d0e1cdb5d8821cc2eff034c3e"

def get\_weather(city\_name):

    url = f"http://api.openweathermap.org/data/2.5/weather?q={city\_name}&appid={API\_KEY}"

    # response = requests.get(url)

    try:

        response = requests.get(url)

        response.raise\_for\_status()

    except requests.exceptions.RequestException as e:

        #  In case an exception occurs during the API request,

        # GitHub Copilot suggests printing an informative error

        # message that includes the exception details.

        # This helps in providing useful feedback to the user.

        print(f"Failed to fetch weather data: {e}")

        return

    if response.status\_code == 200:

        data = response.json()

        temperature = data["main"]["temp"]

        humidity = data["main"]["humidity"]

        weather\_description = data["weather"][0]["description"]

        print(f"Weather in {city\_name}:")

        print(f"Temperature: {temperature} K")

        print(f"Humidity: {humidity}%")

        print(f"Description: {weather\_description}")

def main():

    city\_name = input("Enter city name: ")

    get\_weather(city\_name)

if \_\_name\_\_ == "\_\_main\_\_":

    main()