Python Weather App – Project Brief

Introduction

Create a Python program that simulates a weather forecast for a given city. The application will use hardcoded weather data to provide weather forecasts to the user.

Project Requirements

1. **Welcome Message**: Display a welcome message to the user.
2. **User Input**: Ask for the city name for which the weather forecast is needed.
3. **Fetch Weather Data**: Use hardcoded weather data for several cities to simulate fetching weather information.
4. **Display Weather Data**:
    - Current temperature
    - Weather conditions (e.g., sunny, rainy)
    - Wind speed
    - Humidity
5. **Data Validation**: Ensure valid input by checking that the user enters a valid city name from the hardcoded list.
6. **Thank You Message**: Thank the user for using the weather forecast application.

Design Considerations

- Structure your program to include clear main functions and subroutines.
- Ensure your code is modular and readable.
- Handle invalid city names by informing the user and allowing them to try again.

Example Data:

```
weather_data = { "London": {"temperature": "15°C", "conditions": "Cloudy",
"wind_speed": "5 km/h", "humidity": "80%"}, "New York": {"temperature":
"20°C", "conditions": "Sunny", "wind_speed": "10 km/h", "humidity":
"50%"}, "Tokyo": {"temperature": "18°C", "conditions": "Rainy",
"wind_speed": "7 km/h", "humidity": "90%"}, "Sydney": {"temperature":
"22°C", "conditions": "Windy", "wind_speed": "15 km/h", "humidity":
"60%"}, "Paris": {"temperature": "17°C", "conditions": "Foggy",
"wind_speed": "3 km/h", "humidity": "85%"} }
```