```python
# Flood Risk Prediction Assistant
# SDG 13: Climate Action

def predict_flood_risk(rainfall_mm, river_level_m,
soil_moisture_percent):
    """
    Predict flood risk based on environmental parameters
    """

    if rainfall_mm > 200 and river_level_m > 8 and
soil_moisture_percent > 80:
        return "HIGH Flood Risk"
    elif rainfall_mm > 120 and river_level_m > 6:
        return "MODERATE Flood Risk "
    else:
        return "LOW Flood Risk"


# Sample Input
rainfall = float(input("Enter rainfall (mm): "))
river_level = float(input("Enter river level (m): "))
soil_moisture = float(input("Enter soil moisture (%): "))

# Prediction
risk = predict_flood_risk(rainfall, river_level, soil_moisture)
print("\nFlood Risk Prediction:", risk)

Enter rainfall (mm):  245
Enter river level (m):  15
Enter soil moisture (%):  50


Flood Risk Prediction: MODERATE Flood Risk

# Carbon Footprint Calculator
# SDG 13: Climate Action

def calculate_carbon_footprint(electricity_units, travel_km,
waste_kg):
    """
    Estimate carbon footprint in kg CO2
    """
    electricity_emission = electricity_units * 0.82    # kg CO2 per
unit
    travel_emission = travel_km * 0.21                 # kg CO2 per km
    waste_emission = waste_kg * 0.45                   # kg CO2 per kg

    total_emission = electricity_emission + travel_emission +
waste_emission
    return round(total_emission, 2)
```

```python
# User Inputs
electricity = float(input("Electricity used (units/month): "))
travel = float(input("Travel distance (km/month): "))
waste = float(input("Waste generated (kg/month): "))

# Output
carbon_footprint = calculate_carbon_footprint(electricity, travel,
waste)
print("\nEstimated Monthly Carbon Footprint:", carbon_footprint, "kg
CO2")
```

```
Electricity used (units/month):  34
Travel distance (km/month):  4
Waste generated (kg/month):  2


Estimated Monthly Carbon Footprint: 29.62 kg CO2
```

```python
# Climate Change Awareness Chatbot
# SDG 13: Climate Action

def climate_chatbot(user_input):
    user_input = user_input.lower()

    if "climate change" in user_input:
        return "Climate change refers to long-term changes in
temperature and weather patterns."
    elif "carbon footprint" in user_input:
        return "Carbon footprint is the total greenhouse gas emissions
caused by human activities."
    elif "how to reduce" in user_input:
        return "You can reduce emissions by saving energy, using
public transport, and reducing waste."
    elif "renewable energy" in user_input:
        return "Renewable energy comes from natural sources like
solar, wind, and hydro power."
    else:
        return "Sorry, I can help only with climate-related
questions."


print("Climate Change Awareness Chatbot")
print("Type 'exit' to stop\n")

while True:
    user = input("You: ")
    if user.lower() == "exit":
        print("Bot: Thank you for supporting climate action")
        break
    print("Bot:", climate_chatbot(user))
```

```
Climate Change Awareness Chatbot
Type 'exit' to stop


You:  climate change

Bot: Climate change refers to long-term changes in temperature and
weather patterns.

You:  carbon footprint

Bot: Carbon footprint is the total greenhouse gas emissions caused by
human activities.

You:  how to reduce

Bot: You can reduce emissions by saving energy, using public
transport, and reducing waste.

You:  renewable energy

Bot: Renewable energy comes from natural sources like solar, wind, and
hydro power.

You:  exit

Bot: Thank you for supporting climate action
```