```
import pandas as pd

# Sample soil data
soil_data = pd.DataFrame({
    'Soil_Type': ['Loamy', 'Clayey', 'Sandy'],
    'pH': [6.5, 5.5, 7.0],
    'Nitrogen': [20, 15, 10],
    'Phosphorus': [15, 10, 5],
    'Potassium': [25, 20, 15]
})

# Function to recommend soil type
def recommend_soil(input_soil, soil_data):
    suitable_soils = soil_data[
        (soil_data['pH'] <= input_soil['pH'] + 0.5) &
        (soil_data['pH'] >= input_soil['pH'] - 0.5) &
        (soil_data['Nitrogen'] <= input_soil['Nitrogen'] + 5) &
        (soil_data['Nitrogen'] >= input_soil['Nitrogen'] - 5) &
        (soil_data['Phosphorus'] <= input_soil['Phosphorus'] + 5) &
        (soil_data['Phosphorus'] >= input_soil['Phosphorus'] - 5) &
        (soil_data['Potassium'] <= input_soil['Potassium'] + 5) &
        (soil_data['Potassium'] >= input_soil['Potassium'] - 5)
    ]
    return suitable_soils

# Crop selection function
def crop_selection(soil_type, water_availability):
    if soil_type == "Loamy" and water_availability == "High":
        return "Rice"
    elif soil_type == "Sandy" and water_availability == "Low":
        return "Millets"
    elif soil_type == "Clayey" and water_availability == "Medium":
        return "Wheat"
    else:
        return "Mixed crops like pulses and legumes"

# Soil management function
def soil_management(soil_type):
    if soil_type == "Loamy":
        return "Regular nutrient addition and balanced irrigation."
    elif soil_type == "Sandy":
        return "Increase organic matter and apply frequent irrigation."
    elif soil_type == "Clayey":
        return "Ensure proper drainage and avoid waterlogging."
    else:
        return "Add organic compost and practice crop rotation."

# Disease identification function
def disease_identification(symptoms):
    if "yellowing leaves" in symptoms:
        return "Nitrogen deficiency, consider applying nitrogen-rich fertilizers."
    elif "wilting" in symptoms:
        return "Possible root rot, ensure proper drainage and reduce overwatering."
    elif "brown spots" in symptoms:
        return "Fungal infection, use appropriate fungicides."
    else:
        return "Consult an agricultural expert for accurate diagnosis."

# Main function to run the system
def main():
    print("Welcome to the Crop and Soil Management System")
    print("Please select an option:")
    print("1. Soil Type Recommendation")
    print("2. Crop Selection")
    print("3. Soil Management")
    print("4. Disease Identification")

    choice = int(input("Enter your choice (1/2/3/4): "))

    if choice == 1:
        input_soil = {}
        input_soil['pH'] = float(input("Enter the pH level of the soil: "))
        input_soil['Nitrogen'] = float(input("Enter the Nitrogen content (in mg/kg): "))
        input_soil['Phosphorus'] = float(input("Enter the Phosphorus content (in mg/kg): "))
        input_soil['Potassium'] = float(input("Enter the Potassium content (in mg/kg): "))
        recommended_soils = recommend_soil(input_soil, soil_data)
```

```
            if not recommended_soils.empty:
                print("Recommended Soil Types:")
                print(recommended_soils)
            else:
                print("No suitable soil types found for the given input.")

        elif choice == 2:
            soil_type = input("Enter soil type (Loamy/Sandy/Clayey): ")
            water_availability = input("Enter water availability (High/Medium/Low): ")
            crop = crop_selection(soil_type, water_availability)
            print(f"Recommended crop: {crop}")

        elif choice == 3:
            soil_type = input("Enter soil type (Loamy/Sandy/Clayey): ")
            management_advice = soil_management(soil_type)
            print(f"Soil management advice: {management_advice}")

        elif choice == 4:
            symptoms = input("Enter observed symptoms (e.g., yellowing leaves, wilting, brown spots): ")
            disease = disease_identification(symptoms)
            print(f"Disease diagnosis and advice: {disease}")

        else:
            print("Invalid choice. Please select a valid option.")


    main()
```

```
Welcome to the Crop and Soil Management System
Please select an option:
1. Soil Type Recommendation
2. Crop Selection
3. Soil Management
4. Disease Identification
Enter your choice (1/2/3/4): 1
Enter the pH level of the soil: 5.5
Enter the Nitrogen content (in mg/kg): 15
Enter the Phosphorus content (in mg/kg): 10
Enter the Potassium content (in mg/kg): 15
Recommended Soil Types:
  Soil_Type   pH  Nitrogen  Phosphorus  Potassium
1    Clayey  5.5        15          10         20
```

```
import pandas as pd

# Sample soil data
soil_data = pd.DataFrame({
    'Soil_Type': ['Loamy', 'Clayey', 'Sandy'],
    'pH': [6.5, 5.5, 7.0],
    'Nitrogen': [20, 15, 10],
    'Phosphorus': [15, 10, 5],
    'Potassium': [25, 20, 15]
})

# Function to recommend soil type
def recommend_soil(input_soil, soil_data):
    suitable_soils = soil_data[
        (soil_data['pH'] <= input_soil['pH'] + 0.5) &
        (soil_data['pH'] >= input_soil['pH'] - 0.5) &
        (soil_data['Nitrogen'] <= input_soil['Nitrogen'] + 5) &
        (soil_data['Nitrogen'] >= input_soil['Nitrogen'] - 5) &
        (soil_data['Phosphorus'] <= input_soil['Phosphorus'] + 5) &
        (soil_data['Phosphorus'] >= input_soil['Phosphorus'] - 5) &
        (soil_data['Potassium'] <= input_soil['Potassium'] + 5) &
        (soil_data['Potassium'] >= input_soil['Potassium'] - 5)
    ]
    return suitable_soils

# Crop selection function
def crop_selection(soil_type, water_availability):
    if soil_type == "Loamy" and water_availability == "High":
        return "Rice"
    elif soil_type == "Sandy" and water_availability == "Low":
        return "Millets"
    elif soil_type == "Clayey" and water_availability == "Medium":
        return "Wheat"
    else:
        return "Mixed crops like pulses and legumes"

# Soil management function
def soil_management(soil_type):
    if soil_type == "Loamy":
        return "Regular nutrient addition and balanced irrigation."
    elif soil_type == "Sandy":
        return "Increase organic matter and apply frequent irrigation."
    elif soil_type == "Clayey":
        return "Ensure proper drainage and avoid waterlogging."
    else:
        return "Add organic compost and practice crop rotation."

# Disease identification function
def disease_identification(symptoms):
    if "yellowing leaves" in symptoms:
        return "Nitrogen deficiency, consider applying nitrogen-rich fertilizers."
    elif "wilting" in symptoms:
        return "Possible root rot, ensure proper drainage and reduce overwatering."
    elif "brown spots" in symptoms:
        return "Fungal infection, use appropriate fungicides."
    else:
        return "Consult an agricultural expert for accurate diagnosis."

# Main function to run the system
def main():
    print("Welcome to the Crop and Soil Management System")
    print("Please select an option:")
    print("1. Soil Type Recommendation")
    print("2. Crop Selection")
    print("3. Soil Management")
    print("4. Disease Identification")

    choice = int(input("Enter your choice (1/2/3/4): "))

    if choice == 1:
        input_soil = {}
        input_soil['pH'] = float(input("Enter the pH level of the soil: "))
        input_soil['Nitrogen'] = float(input("Enter the Nitrogen content (in mg/kg): "))
        input_soil['Phosphorus'] = float(input("Enter the Phosphorus content (in mg/kg): "))
        input_soil['Potassium'] = float(input("Enter the Potassium content (in mg/kg): "))
        recommended_soils = recommend_soil(input_soil, soil_data)
```

```
        if not recommended_soils.empty:
            print("Recommended Soil Types:")
            print(recommended_soils)
        else:
            print("No suitable soil types found for the given input.")

    elif choice == 2:
        soil_type = input("Enter soil type (Loamy/Sandy/Clayey): ")
        water_availability = input("Enter water availability (High/Medium/Low): ")
        crop = crop_selection(soil_type, water_availability)
        print(f"Recommended crop: {crop}")

    elif choice == 3:
        soil_type = input("Enter soil type (Loamy/Sandy/Clayey): ")
        management_advice = soil_management(soil_type)
        print(f"Soil management advice: {management_advice}")

    elif choice == 4:
        symptoms = input("Enter observed symptoms (e.g., yellowing leaves, wilting, brown spots): ")
        disease = disease_identification(symptoms)
        print(f"Disease diagnosis and advice: {disease}")

    else:
        print("Invalid choice. Please select a valid option.")


main()
```

```
Welcome to the Crop and Soil Management System
Please select an option:
1. Soil Type Recommendation
2. Crop Selection
3. Soil Management
4. Disease Identification
Enter your choice (1/2/3/4): 2
Enter soil type (Loamy/Sandy/Clayey): Loamy
Enter water availability (High/Medium/Low): Medium
Recommended crop: Mixed crops like pulses and legumes
```

```
import pandas as pd

# Sample soil data
soil_data = pd.DataFrame({
    'Soil_Type': ['Loamy', 'Clayey', 'Sandy'],
    'pH': [6.5, 5.5, 7.0],
    'Nitrogen': [20, 15, 10],
    'Phosphorus': [15, 10, 5],
    'Potassium': [25, 20, 15]
})

# Function to recommend soil type
def recommend_soil(input_soil, soil_data):
    suitable_soils = soil_data[
        (soil_data['pH'] <= input_soil['pH'] + 0.5) &
        (soil_data['pH'] >= input_soil['pH'] - 0.5) &
        (soil_data['Nitrogen'] <= input_soil['Nitrogen'] + 5) &
        (soil_data['Nitrogen'] >= input_soil['Nitrogen'] - 5) &
        (soil_data['Phosphorus'] <= input_soil['Phosphorus'] + 5) &
        (soil_data['Phosphorus'] >= input_soil['Phosphorus'] - 5) &
        (soil_data['Potassium'] <= input_soil['Potassium'] + 5) &
        (soil_data['Potassium'] >= input_soil['Potassium'] - 5)
    ]
    return suitable_soils

# Crop selection function
def crop_selection(soil_type, water_availability):
    if soil_type == "Loamy" and water_availability == "High":
        return "Rice"
    elif soil_type == "Sandy" and water_availability == "Low":
        return "Millets"
    elif soil_type == "Clayey" and water_availability == "Medium":
        return "Wheat"
    else:
        return "Mixed crops like pulses and legumes"

# Soil management function
def soil_management(soil_type):
    if soil_type == "Loamy":
        return "Regular nutrient addition and balanced irrigation."
    elif soil_type == "Sandy":
        return "Increase organic matter and apply frequent irrigation."
    elif soil_type == "Clayey":
        return "Ensure proper drainage and avoid waterlogging."
    else:
        return "Add organic compost and practice crop rotation."

# Disease identification function
def disease_identification(symptoms):
    if "yellowing leaves" in symptoms:
        return "Nitrogen deficiency, consider applying nitrogen-rich fertilizers."
    elif "wilting" in symptoms:
        return "Possible root rot, ensure proper drainage and reduce overwatering."
    elif "brown spots" in symptoms:
        return "Fungal infection, use appropriate fungicides."
    else:
        return "Consult an agricultural expert for accurate diagnosis."

# Main function to run the system
def main():
    print("Welcome to the Crop and Soil Management System")
    print("Please select an option:")
    print("1. Soil Type Recommendation")
    print("2. Crop Selection")
    print("3. Soil Management")
    print("4. Disease Identification")

    choice = int(input("Enter your choice (1/2/3/4): "))

    if choice == 1:
        input_soil = {}
        input_soil['pH'] = float(input("Enter the pH level of the soil: "))
        input_soil['Nitrogen'] = float(input("Enter the Nitrogen content (in mg/kg): "))
        input_soil['Phosphorus'] = float(input("Enter the Phosphorus content (in mg/kg): "))
        input_soil['Potassium'] = float(input("Enter the Potassium content (in mg/kg): "))
        recommended_soils = recommend_soil(input_soil, soil_data)
```

```
            if not recommended_soils.empty:
                print("Recommended Soil Types:")
                print(recommended_soils)
            else:
                print("No suitable soil types found for the given input.")

        elif choice == 2:
            soil_type = input("Enter soil type (Loamy/Sandy/Clayey): ")
            water_availability = input("Enter water availability (High/Medium/Low): ")
            crop = crop_selection(soil_type, water_availability)
            print(f"Recommended crop: {crop}")

        elif choice == 3:
            soil_type = input("Enter soil type (Loamy/Sandy/Clayey): ")
            management_advice = soil_management(soil_type)
            print(f"Soil management advice: {management_advice}")

        elif choice == 4:
            symptoms = input("Enter observed symptoms (e.g., yellowing leaves, wilting, brown spots): ")
            disease = disease_identification(symptoms)
            print(f"Disease diagnosis and advice: {disease}")

        else:
            print("Invalid choice. Please select a valid option.")


main()
```

```
Welcome to the Crop and Soil Management System
Please select an option:
1. Soil Type Recommendation
2. Crop Selection
3. Soil Management
4. Disease Identification
Enter your choice (1/2/3/4): 3
Enter soil type (Loamy/Sandy/Clayey): Sandy
Soil management advice: Increase organic matter and apply frequent irrigation.
```

```
import pandas as pd

# Sample soil data
soil_data = pd.DataFrame({
    'Soil_Type': ['Loamy', 'Clayey', 'Sandy'],
    'pH': [6.5, 5.5, 7.0],
    'Nitrogen': [20, 15, 10],
    'Phosphorus': [15, 10, 5],
    'Potassium': [25, 20, 15]
})

# Function to recommend soil type
def recommend_soil(input_soil, soil_data):
    suitable_soils = soil_data[
        (soil_data['pH'] <= input_soil['pH'] + 0.5) &
        (soil_data['pH'] >= input_soil['pH'] - 0.5) &
        (soil_data['Nitrogen'] <= input_soil['Nitrogen'] + 5) &
        (soil_data['Nitrogen'] >= input_soil['Nitrogen'] - 5) &
        (soil_data['Phosphorus'] <= input_soil['Phosphorus'] + 5) &
        (soil_data['Phosphorus'] >= input_soil['Phosphorus'] - 5) &
        (soil_data['Potassium'] <= input_soil['Potassium'] + 5) &
        (soil_data['Potassium'] >= input_soil['Potassium'] - 5)
    ]
    return suitable_soils

# Crop selection function
def crop_selection(soil_type, water_availability):
    if soil_type == "Loamy" and water_availability == "High":
        return "Rice"
    elif soil_type == "Sandy" and water_availability == "Low":
        return "Millets"
    elif soil_type == "Clayey" and water_availability == "Medium":
        return "Wheat"
    else:
        return "Mixed crops like pulses and legumes"

# Soil management function
def soil_management(soil_type):
    if soil_type == "Loamy":
        return "Regular nutrient addition and balanced irrigation."
    elif soil_type == "Sandy":
        return "Increase organic matter and apply frequent irrigation."
    elif soil_type == "Clayey":
        return "Ensure proper drainage and avoid waterlogging."
    else:
        return "Add organic compost and practice crop rotation."

# Disease identification function
def disease_identification(symptoms):
    if "yellowing leaves" in symptoms:
        return "Nitrogen deficiency, consider applying nitrogen-rich fertilizers."
    elif "wilting" in symptoms:
        return "Possible root rot, ensure proper drainage and reduce overwatering."
    elif "brown spots" in symptoms:
        return "Fungal infection, use appropriate fungicides."
    else:
        return "Consult an agricultural expert for accurate diagnosis."

# Main function to run the system
def main():
    print("Welcome to the Crop and Soil Management System")
    print("Please select an option:")
    print("1. Soil Type Recommendation")
    print("2. Crop Selection")
    print("3. Soil Management")
    print("4. Disease Identification")

    choice = int(input("Enter your choice (1/2/3/4): "))

    if choice == 1:
        input_soil = {}
        input_soil['pH'] = float(input("Enter the pH level of the soil: "))
        input_soil['Nitrogen'] = float(input("Enter the Nitrogen content (in mg/kg): "))
        input_soil['Phosphorus'] = float(input("Enter the Phosphorus content (in mg/kg): "))
        input_soil['Potassium'] = float(input("Enter the Potassium content (in mg/kg): "))
        recommended_soils = recommend_soil(input_soil, soil_data)
```

```
            if not recommended_soils.empty:
                print("Recommended Soil Types:")
                print(recommended_soils)
            else:
                print("No suitable soil types found for the given input.")

        elif choice == 2:
            soil_type = input("Enter soil type (Loamy/Sandy/Clayey): ")
            water_availability = input("Enter water availability (High/Medium/Low): ")
            crop = crop_selection(soil_type, water_availability)
            print(f"Recommended crop: {crop}")

        elif choice == 3:
            soil_type = input("Enter soil type (Loamy/Sandy/Clayey): ")
            management_advice = soil_management(soil_type)
            print(f"Soil management advice: {management_advice}")

        elif choice == 4:
            symptoms = input("Enter observed symptoms (e.g., yellowing leaves, wilting, brown spots): ")
            disease = disease_identification(symptoms)
            print(f"Disease diagnosis and advice: {disease}")

        else:
            print("Invalid choice. Please select a valid option.")


main()
```

```
Welcome to the Crop and Soil Management System
Please select an option:
1. Soil Type Recommendation
2. Crop Selection
3. Soil Management
4. Disease Identification
Enter your choice (1/2/3/4): 4
Enter observed symptoms (e.g., yellowing leaves, wilting, brown spots): brown spots
Disease diagnosis and advice: Fungal infection, use appropriate fungicides.
```