import pandas as pd

import matplotlib.pyplot as plt

import numpy as np

# Sample Data (You can replace this with actual data from CSV, database, or API)

# For agricultural analysis, we'll have columns: crop type, yield, rainfall, temperature, soil condition.

data\_agriculture = {

"Crop": ["Wheat", "Rice", "Corn", "Barley", "Oats"],

"Yield (kg/ha)": [3000, 4500, 3500, 2500, 2000],

"Rainfall (mm)": [600, 1200, 800, 500, 700],

"Temperature (°C)": [25, 28, 30, 22, 24],

"Soil Quality (1-10)": [8, 7, 6, 8, 6]

}

# Raw Material Analysis Data

# For raw material analysis, we have stock levels, cost, and quality ratings.

data\_raw\_material = {

"Material": ["Steel", "Cotton", "Plastic", "Wood", "Copper"],

"Stock (tons)": [500, 1000, 700, 800, 600],

"Cost per Ton ($)": [1500, 1200, 800, 450, 600],

"Quality Rating (1-10)": [9, 7, 6, 8, 9]

}

# Create DataFrames

df\_agriculture = pd.DataFrame(data\_agriculture)

df\_raw\_material = pd.DataFrame(data\_raw\_material)

# Agriculture Analysis

def plot\_agriculture\_analysis(df):

# Crop Yield vs Rainfall

plt.figure(figsize=(10, 6))

plt.scatter(df['Rainfall (mm)'], df['Yield (kg/ha)'], c='blue', label='Yield vs Rainfall')

plt.title('Crop Yield vs Rainfall')

plt.xlabel('Rainfall (mm)')

plt.ylabel('Yield (kg/ha)')

plt.grid(True)

plt.legend()

plt.show()

# Yield vs Soil Quality

plt.figure(figsize=(10, 6))

plt.scatter(df['Soil Quality (1-10)'], df['Yield (kg/ha)'], c='green', label='Yield vs Soil Quality')

plt.title('Crop Yield vs Soil Quality')

plt.xlabel('Soil Quality (1-10)')

plt.ylabel('Yield (kg/ha)')

plt.grid(True)

plt.legend()

plt.show()

# Raw Material Analysis

def plot\_raw\_material\_analysis(df):

# Stock vs Cost

plt.figure(figsize=(10, 6))

plt.bar(df['Material'], df['Stock (tons)'], color='orange', label='Stock Levels')

plt.title('Raw Material Stock Levels')

plt.xlabel('Material')

plt.ylabel('Stock (tons)')

plt.grid(True)

plt.legend()

plt.show()

# Quality Rating vs Cost

plt.figure(figsize=(10, 6))

plt.scatter(df['Cost per Ton ($)'], df['Quality Rating (1-10)'], c='red', label='Cost vs Quality')

plt.title('Raw Material Cost vs Quality Rating')

plt.xlabel('Cost per Ton ($)')

plt.ylabel('Quality Rating (1-10)')

plt.grid(True)

plt.legend()

plt.show()

# Analyzing Agriculture Data

plot\_agriculture\_analysis(df\_agriculture)

# Analyzing Raw Material Data

plot\_raw\_material\_analysis(df\_raw\_material)

# Additional Statistical Analysis

def calculate\_agriculture\_statistics(df):

print("\nAgriculture Data Statistics:")

print("Mean Yield (kg/ha): ", np.mean(df['Yield (kg/ha)']))

print("Mean Rainfall (mm): ", np.mean(df['Rainfall (mm)']))

print("Mean Temperature (°C): ", np.mean(df['Temperature (°C)']))

print("Mean Soil Quality (1-10): ", np.mean(df['Soil Quality (1-10)']))

def calculate\_raw\_material\_statistics(df):

print("\nRaw Material Data Statistics:")

print("Total Stock (tons): ", np.sum(df['Stock (tons)']))

print("Average Cost per Ton ($): ", np.mean(df['Cost per Ton ($)']))

print("Average Quality Rating (1-10): ", np.mean(df['Quality Rating (1-10)']))

# Displaying Statistics

calculate\_agriculture\_statistics(df\_agriculture)

calculate\_raw\_material\_statistics(df\_raw\_material)