import pandas as pd

from sklearn.ensemble import RandomForestClassifier

from sklearn.model\_selection import train\_test\_split

from sklearn.metrics import accuracy\_score

import smtplib

import os

# 1. Prepare dataset

data = {

'rainfall': [120, 80, 150, 30, 90, 200, 250, 50],

'water\_level': [2.5, 1.8, 3.2, 0.9, 2.1, 3.6, 4.0, 1.2],

'soil\_moisture': [45, 30, 60, 15, 35, 70, 80, 20],

'flood': [1, 0, 1, 0, 0, 1, 1, 0]

}

df = pd.DataFrame(data)

X = df[['rainfall', 'water\_level', 'soil\_moisture']]

y = df['flood']

# 2. Train/test split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(

X, y, test\_size=0.25, random\_state=42

)

# 3. Train the model

model = RandomForestClassifier(n\_estimators=100, random\_state=42)

model.fit(X\_train, y\_train)

# 4. Evaluate accuracy

y\_pred = model.predict(X\_test)

print("Model Accuracy:", accuracy\_score(y\_test, y\_pred))

# 5. New sensor data (replace with live sensor input in real system)

new\_data = [[180, 3.1, 65]]

prediction = model.predict(new\_data)

# 6. Send alert if needed

def send\_alert():

try:

sender\_email = os.getenv("ALERT\_EMAIL")

sender\_password = os.getenv("ALERT\_PASSWORD")

receiver\_email = os.getenv("RECEIVER\_EMAIL")

if not all([sender\_email, sender\_password, receiver\_email]):

raise EnvironmentError("Missing email credentials.")

server = smtplib.SMTP('smtp.example.com', 587)

server.starttls()

server.login(sender\_email, sender\_password)

message = "Subject: Flood Alert\n\nFlood risk detected in your area. Take immediate precautions!"

server.sendmail(sender\_email, receiver\_email, message)

server.quit()

print("✅ Email alert sent successfully!")

except Exception as e:

print(f"⚠ Mock alert: Email not sent. Reason: {e}")

if prediction[0] == 1:

print("⚠ ALERT: Flood risk detected! Notifying authorities...")

send\_alert()

else:

print("✅ No flood risk detected.")