**SOURCE CODE FOR AI-STRUCTURAL HEALTH MONITORING**

import random

import time

# Set thresholds

VIBRATION\_THRESHOLD = 5.0 # mm/s

STRAIN\_THRESHOLD = 250 # microstrain

def read\_vibration\_sensor():

"""Simulate vibration sensor reading (0–10 mm/s)."""

return round(random.uniform(0, 10), 2)

def read\_strain\_sensor():

"""Simulate strain sensor reading (100–300 microstrain)."""

return random.randint(100, 300)

def check\_anomalies(vibration, strain):

"""Check if readings exceed thresholds."""

anomalies = []

if vibration > VIBRATION\_THRESHOLD:

anomalies.append(f"High vibration: {vibration} mm/s")

if strain > STRAIN\_THRESHOLD:

anomalies.append(f"High strain: {strain} µε")

return anomalies

def monitor\_structure(cycles=10, interval=1):

"""Main monitoring loop."""

print("=== Structural Health Monitoring ===")

for i in range(1, cycles + 1):

vibration = read\_vibration\_sensor()

strain = read\_strain\_sensor()

print(f"\nCycle {i}:")

print(f" Vibration: {vibration} mm/s")

print(f" Strain: {strain} µε")

anomalies = check\_anomalies(vibration, strain)

if anomalies:

print(" ⚠️ Anomalies Detected:")

for anomaly in anomalies:

print(f" - {anomaly}")

else:

print(" ✅ Structure is stable.")

time.sleep(interval)

print("\nMonitoring complete.")

if \_\_name\_\_ == "\_\_main\_\_":

monitor\_structure()

def check\_anomalies(vibration, strain):

"""Check if readings exceed thresholds."""

anomalies = []

if vibration > VIBRATION\_THRESHOLD:

anomalies.append(f"High vibration: {vibration} mm/s")

if strain > STRAIN\_THRESHOLD:

anomalies.append(f"High strain: {strain} µε")

return anomalies

def monitor\_structure(cycles=10, interval=1):

"""Main monitoring loop."""

print("=== Structural Health Monitoring ===")

for i in range(1, cycles + 1):

vibration = read\_vibration\_sensor()

strain = read\_strain\_sensor()

print(f"\nCycle {i}:")

print(f" Vibration: {vibration} mm/s")

print(f" Strain: {strain} µε")

anomalies = check\_anomalies(vibration, strain)

if anomalies:

print(" ⚠️ Anomalies Detected:")

for anomaly in anomalies:

print(f" - {anomaly}")

else:

print(" ✅ Structure is stable.")

time.sleep(interval)

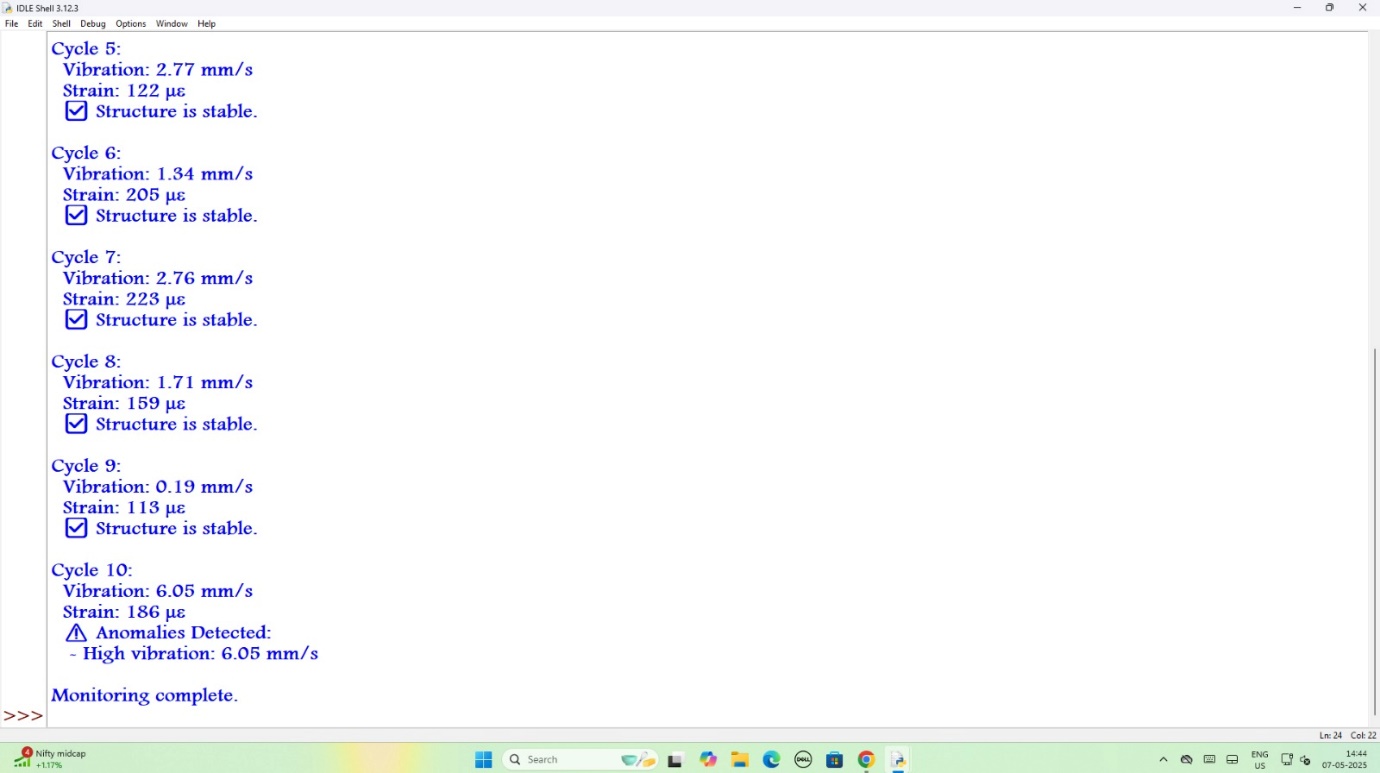
print("\nMonitoring complete.")

if \_\_name\_\_ == "\_\_main\_\_":

monitor\_structure()

**OUTPUT:**

****

****