PHASE 4: Performance of the Project

TITLE: ROOT CAUSE ANALYSIS FOR EQUIPMENT FAILURES

Objective:

The focus of Phase 4 is to improve the accuracy, speed, and efficiency of the root cause analysis (RCA) system for identifying and addressing equipment failures in industrial environments. This phase involves enhancing data collection methods, refining analytical models, and integrating predictive maintenance features to reduce unplanned downtimes and improve equipment reliability.

1. RCA Model Enhancement

Overview:

The RCA model will be upgraded to analyse failure patterns more accurately using historical and real-time data. The goal is to pinpoint underlying causes more effectively and reduce false diagnoses.

Performance Improvements:

- **Data Enrichment:** The model will include broader failure modes and additional datasets from maintenance logs, sensor inputs, and operational anomalies.
- **Algorithm Refinement:** Improved fault tree analysis (FTA) and machine learning algorithms will be deployed for better diagnostics.

Outcome:

The system will identify the true root causes of equipment failures with greater precision, supporting faster corrective actions and reducing repeat failures.

2. Fault Detection System Optimization

Overview:
The fault detection system will be optimized for quicker detection and classification of equipment anomalies, enabling real-time alerting and faster mitigation.

Key Enhancements:

- **Sensor Integration:** Real-time sensor data (vibration, temperature, pressure, etc.) will be used to detect early signs of equipment degradation.
- **Anomaly Detection:** Advanced anomaly detection algorithms will flag irregular patterns before full failure occurs.

Outcome:
Early identification of failure symptoms will allow preemptive maintenance, minimizing costly breakdowns.

3. Data Visualization and Reporting

Overview:
Enhanced dashboards and visualization tools will present actionable insights from the RCA process in a user-friendly format.

Key Enhancements:

- **Failure Trends Dashboard**: Displays frequency, severity, and duration of failures by equipment type and timeframe.
- **Root Cause Trees**: Visual representation of causal chains for better understanding by maintenance teams.

Outcome:
Improved visibility and transparency into failure causes, helping stakeholders make data-driven decisions to improve equipment reliability.

4. Integration with Maintenance Systems

Overview:
This phase ensures seamless integration with existing Computerized Maintenance Management Systems (CMMS) to automate work orders based on RCA findings.

Key Enhancements:

- **CMMS API Integration**: RCA insights will automatically trigger maintenance actions or recommend preventive measures.
- **Maintenance History Correlation**: Past maintenance records will be linked to current RCA outcomes to detect recurring issues.

Outcome:
Faster response and corrective measures, with reduced manual intervention in generating maintenance tasks.

5. Performance Testing and Metrics Collection

Overview:
The RCA tool's performance will be tested under various failure scenarios and data loads to ensure scalability and robustness.

Implementation:

- **Simulation Testing**: Simulated equipment failure scenarios will test RCA responsiveness and accuracy.
- **Metric Monitoring**: Key metrics such as diagnosis accuracy, false detection rate, time to resolution, and system uptime will be tracked.
- **User Feedback**: Field engineers and plant operators will test the system in real settings and provide usability feedback.

Outcome:
A high-performing RCA system capable of rapid, accurate diagnosis with proven improvements in operational continuity and reduced downtime.

Key Challenges in Phase 4

1. **Handling Incomplete Data**
   - *Challenge*: Inaccurate or missing data can reduce diagnostic effectiveness.
   - *Solution*: Implement data validation and cleansing techniques, and use imputation for missing data.
2. **Integrating with Legacy Equipment**
   - *Challenge*: Older machines may lack compatible sensors or data interfaces.
   - *Solution*: Use IoT adapters or retrofitting kits to enable data collection from legacy assets.
3. **Avoiding False Positives/Negatives**
   - *Challenge*: Misclassification of faults can lead to unnecessary maintenance or missed failures.
   - *Solution*: Continuous model retraining and validation with real-world feedback.

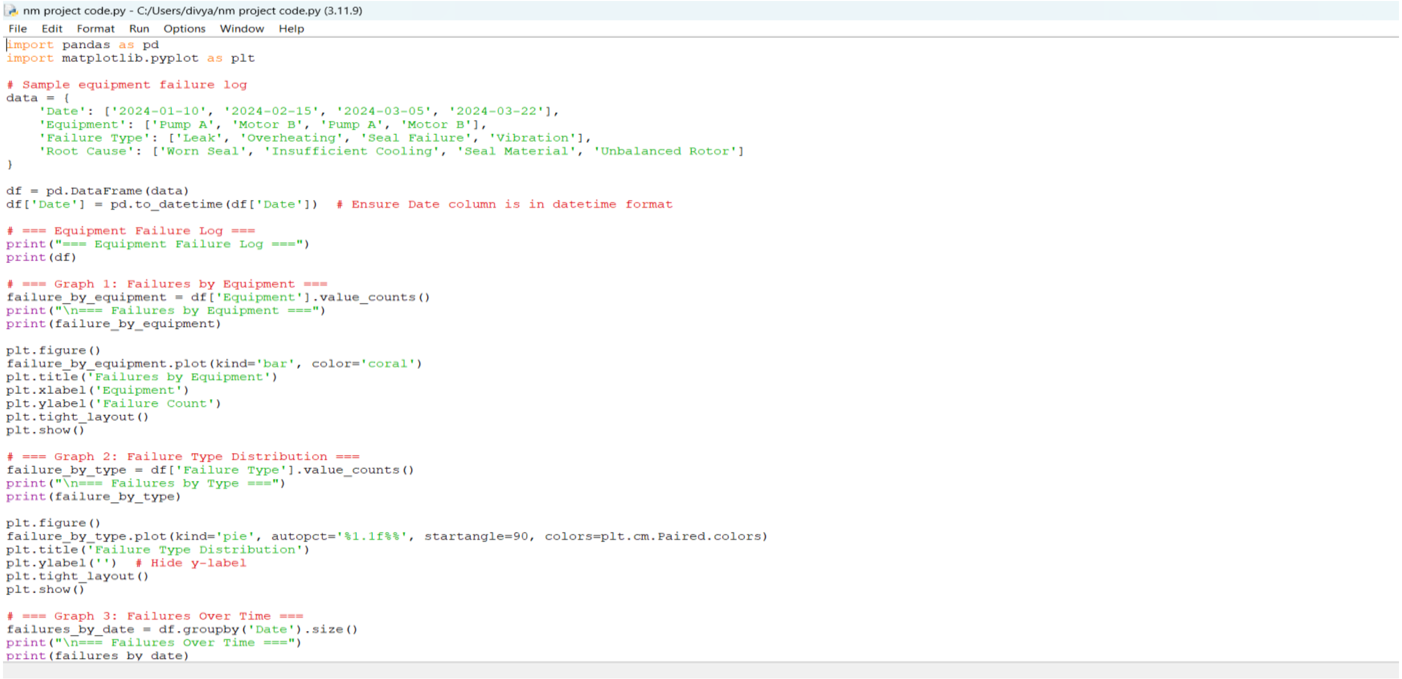
Outcomes of Phase 4

1. **Improved Diagnostic Accuracy**: The system can pinpoint true causes of failure with minimal manual input.
2. **Reduced Downtime**: Quicker detection and intervention reduce unplanned downtime.
3. **Actionable Insights**: Maintenance teams receive clearer, prioritized recommendations.
4. **Higher Operational Efficiency**: Streamlined processes improve asset utilization and reduce costs.

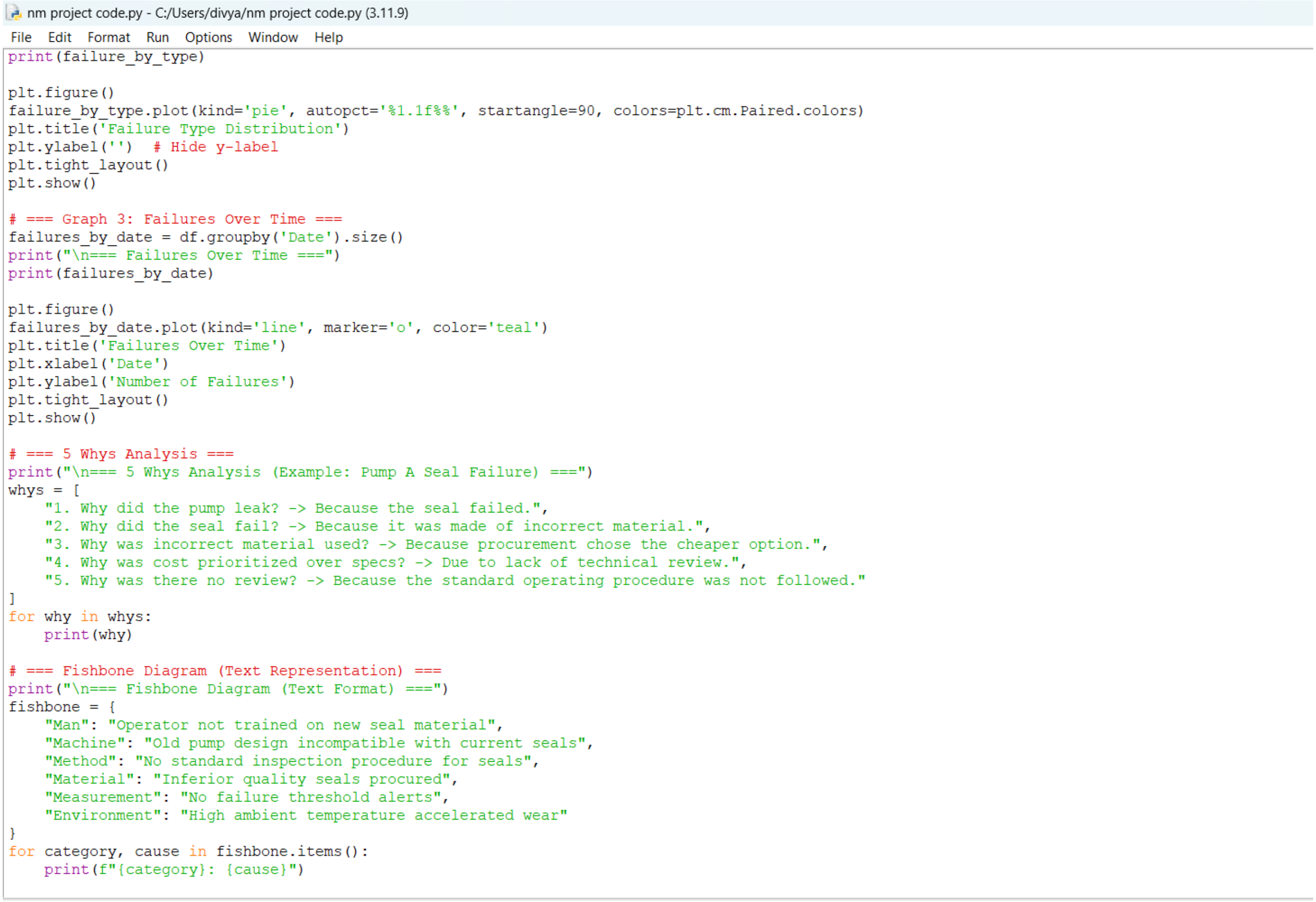
Next Steps for Finalization

In the final phase, the RCA system will be deployed fully across the target facility. Post-deployment feedback will be collected to further refine model accuracy and enhance predictive capabilities.

SAMPLE CODE:


nm project code.py - C:/Users/divya/nm project code.py (3.11.9)
File Edit Format Run Options Window Help

```python
import pandas as pd
import matplotlib.pyplot as plt

# Sample equipment failure log
data = {
    'Date': ['2024-01-10', '2024-02-15', '2024-03-05', '2024-03-22'],
    'Equipment': ['Pump A', 'Motor B', 'Pump A', 'Motor B'],
    'Failure Type': ['Leak', 'Overheating', 'Seal Failure', 'Vibration'],
    'Root Cause': ['Worn Seal', 'Insufficient Cooling', 'Seal Material', 'Unbalanced Rotor']
}

df = pd.DataFrame(data)
df['Date'] = pd.to_datetime(df['Date'])  # Ensure Date column is in datetime format

# === Equipment Failure Log ===
print("=== Equipment Failure Log ===")
print(df)

# === Graph 1: Failures by Equipment ===
failure_by_equipment = df['Equipment'].value_counts()
print("\n=== Failures by Equipment ===")
print(failure_by_equipment)

plt.figure()
failure_by_equipment.plot(kind='bar', color='coral')
plt.title('Failures by Equipment')
plt.xlabel('Equipment')
plt.ylabel('Failure Count')
plt.tight_layout()
plt.show()

# === Graph 2: Failure Type Distribution ===
failure_by_type = df['Failure Type'].value_counts()
print("\n=== Failures by Type ===")
print(failure_by_type)

plt.figure()
failure_by_type.plot(kind='pie', autopct='%1.1f%%', startangle=90, colors=plt.cm.Paired.colors)
plt.title('Failure Type Distribution')
plt.ylabel('')  # Hide y-label
plt.tight_layout()
plt.show()

# === Graph 3: Failures Over Time ===
failures_by_date = df.groupby('Date').size()
print("\n=== Failures Over Time ===")
print(failures_by_date)
```


nm project code.py - C:/Users/divya/nm project code.py (3.11.9)
File Edit Format Run Options Window Help

```python
print(failure_by_type)

plt.figure()
failure_by_type.plot(kind='pie', autopct='%1.1f%%', startangle=90, colors=plt.cm.Paired.colors)
plt.title('Failure Type Distribution')
plt.ylabel('')  # Hide y-label
plt.tight_layout()
plt.show()

# === Graph 3: Failures Over Time ===
failures_by_date = df.groupby('Date').size()
print("\n=== Failures Over Time ===")
print(failures_by_date)

plt.figure()
failures_by_date.plot(kind='line', marker='o', color='teal')
plt.title('Failures Over Time')
plt.xlabel('Date')
plt.ylabel('Number of Failures')
plt.tight_layout()
plt.show()

# === 5 Whys Analysis ===
print("\n=== 5 Whys Analysis (Example: Pump A Seal Failure) ===")
whys = [
    "1. Why did the pump leak? -> Because the seal failed.",
    "2. Why did the seal fail? -> Because it was made of incorrect material.",
    "3. Why was incorrect material used? -> Because procurement chose the cheaper option.",
    "4. Why was cost prioritized over specs? -> Due to lack of technical review.",
    "5. Why was there no review? -> Because the standard operating procedure was not followed."
]
for why in whys:
    print(why)

# === Fishbone Diagram (Text Representation) ===
print("\n=== Fishbone Diagram (Text Format) ===")
fishbone = {
    "Man": "Operator not trained on new seal material",
    "Machine": "Old pump design incompatible with current seals",
    "Method": "No standard inspection procedure for seals",
    "Material": "Inferior quality seals procured",
    "Measurement": "No failure threshold alerts",
    "Environment": "High ambient temperature accelerated wear"
}
for category, cause in fishbone.items():
    print(f"{category}: {cause}")
```

OUTCOMES:

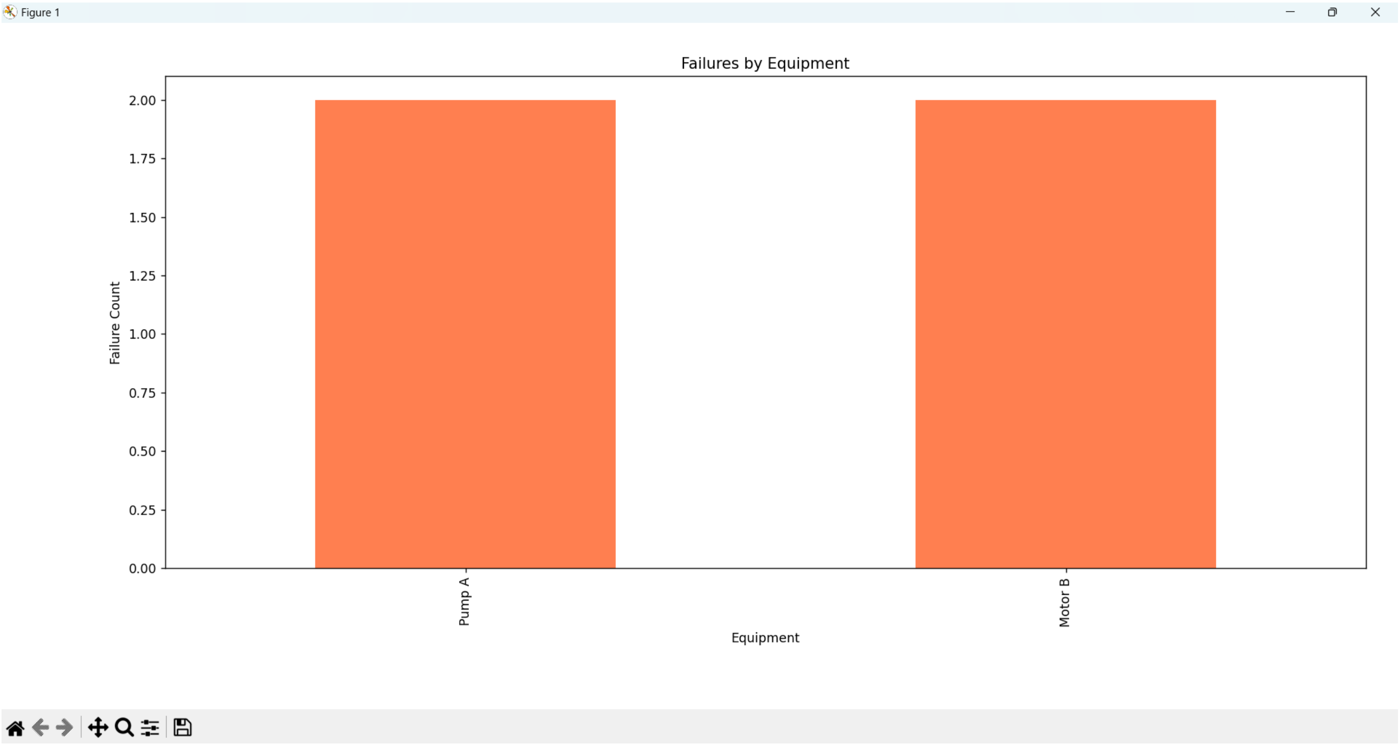


Figure 1
Failures by Equipment
Failure Count
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
Pump A
Motor B
Equipment

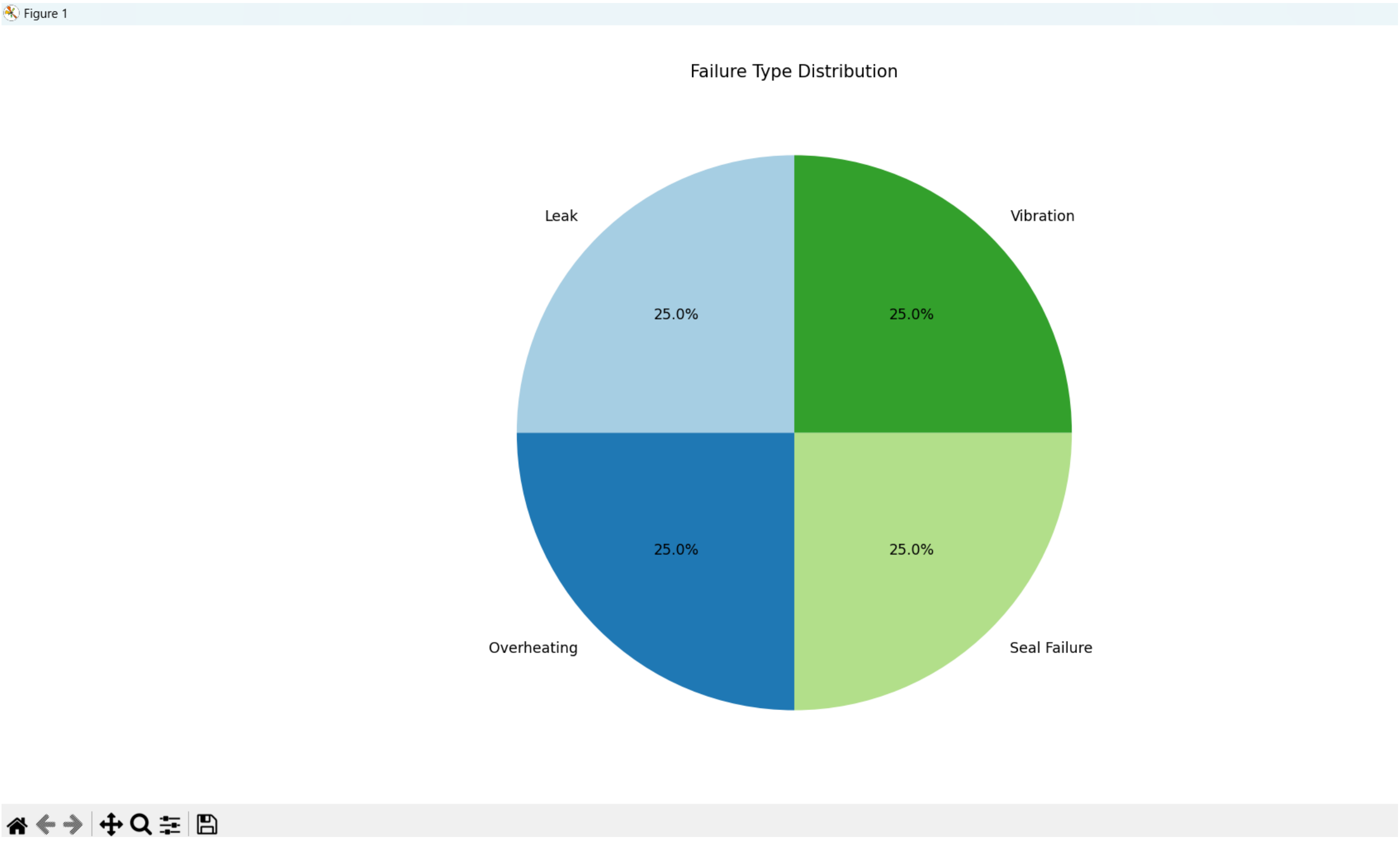
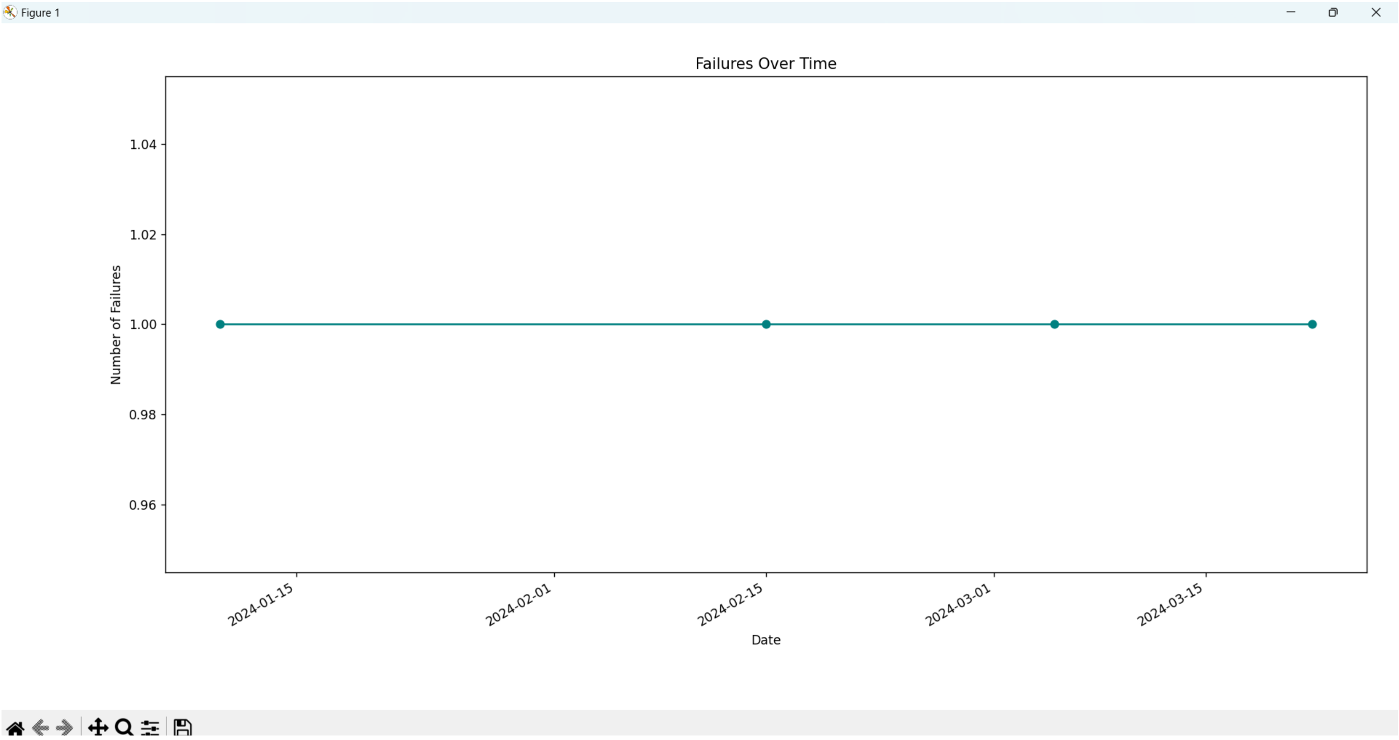
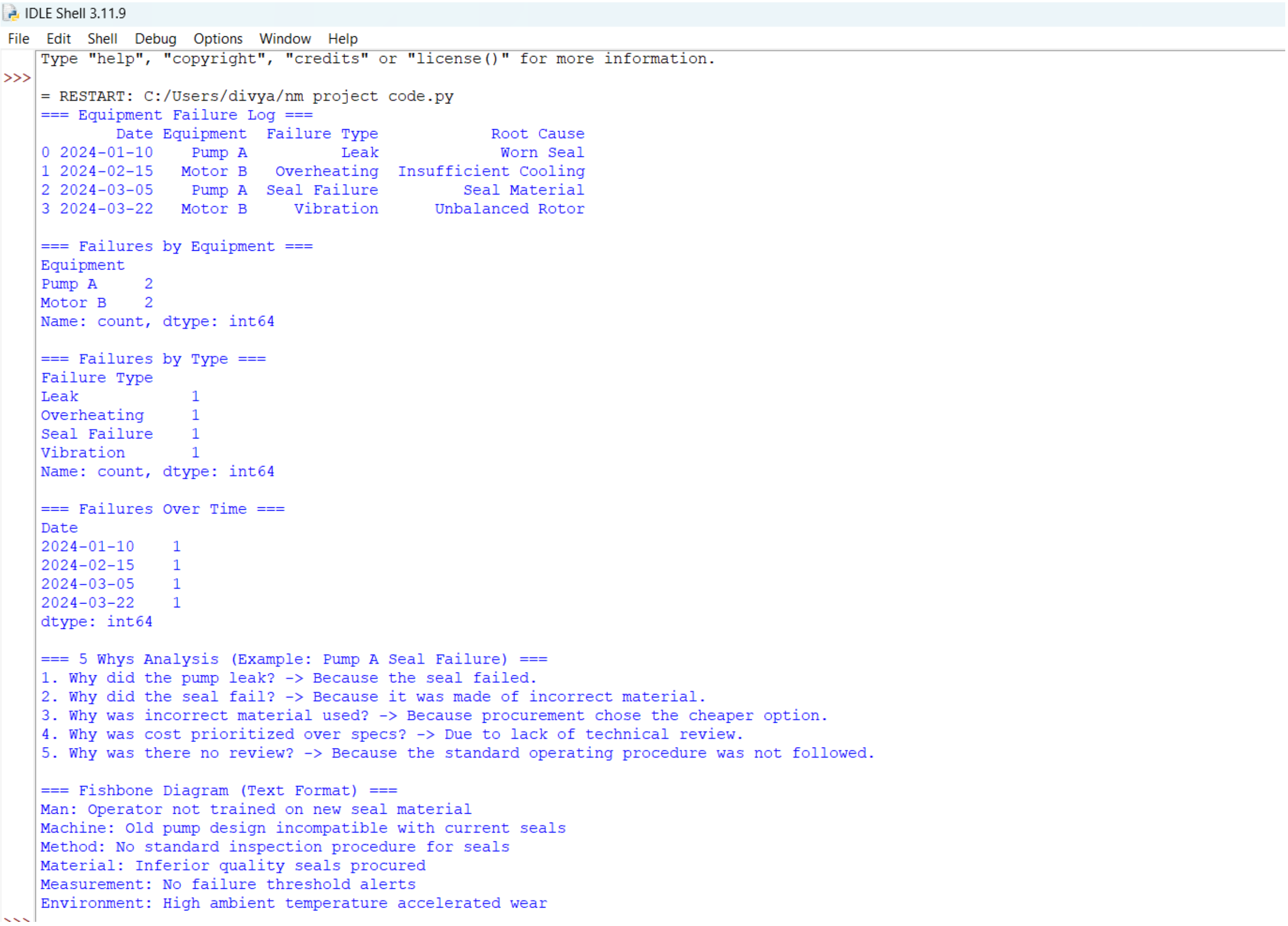


Figure 1
Failure Type Distribution
Leak
25.0%
Vibration
25.0%
Overheating
25.0%
Seal Failure
25.0%





```
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:/Users/divya/nm project code.py
=== Equipment Failure Log ===
        Date Equipment  Failure Type            Root Cause
0 2024-01-10    Pump A          Leak             Worn Seal
1 2024-02-15   Motor B   Overheating  Insufficient Cooling
2 2024-03-05    Pump A  Seal Failure         Seal Material
3 2024-03-22   Motor B     Vibration      Unbalanced Rotor

=== Failures by Equipment ===
Equipment
Pump A     2
Motor B    2
Name: count, dtype: int64

=== Failures by Type ===
Failure Type
Leak            1
Overheating     1
Seal Failure    1
Vibration       1
Name: count, dtype: int64

=== Failures Over Time ===
Date
2024-01-10    1
2024-02-15    1
2024-03-05    1
2024-03-22    1
dtype: int64

=== 5 Whys Analysis (Example: Pump A Seal Failure) ===
1. Why did the pump leak? -> Because the seal failed.
2. Why did the seal fail? -> Because it was made of incorrect material.
3. Why was incorrect material used? -> Because procurement chose the cheaper option.
4. Why was cost prioritized over specs? -> Due to lack of technical review.
5. Why was there no review? -> Because the standard operating procedure was not followed.

=== Fishbone Diagram (Text Format) ===
Man: Operator not trained on new seal material
Machine: Old pump design incompatible with current seals
Method: No standard inspection procedure for seals
Material: Inferior quality seals procured
Measurement: No failure threshold alerts
Environment: High ambient temperature accelerated wear
>>>
```