# Solutions for Accuknox QA Trainee Practical Assessment

Problem Statement 1: Wisecow Application on Kubernetes

1. Dockerfile:

```
FROM python:3.9-slim

WORKDIR /app

COPY . /app

RUN pip install -r requirements.txt

CMD ["python", "app.py"]
```

2. Kubernetes Manifest Files (Example):

Deployment YAML:

```
apiVersion: apps/v1

kind: Deployment

metadata:

  name: wisecow-deployment

spec:

  replicas: 3

  selector:

    matchLabels:

      app: wisecow

  template:

    metadata:

      labels:

        app: wisecow
    spec:
      containers:
      - name: wisecow
        image: your-docker-image:tag
        ports:
        - containerPort: 80
```

Service YAML:

```
apiVersion: v1
kind: Service
metadata:
  name: wisecow-service
spec:
  selector:
    app: wisecow
  ports:
  - protocol: TCP
    port: 80
    targetPort: 80
  type: LoadBalancer
```

3. CI/CD Pipeline Configuration (GitHub Actions):

```
name: CI/CD Pipeline
on:
  push:
    branches:
      - main
```

```yaml
jobs:
  build:
    runs-on: ubuntu-latest
    steps:
      - name: Checkout code
        uses: actions/checkout@v3
      - name: Build Docker image
        run: |
          docker build -t your-docker-image:tag .
      - name: Push Docker image
        run: |
          docker push your-docker-image:tag
  deploy:
    needs: build
    runs-on: ubuntu-latest
    steps:
      - name: Deploy to Kubernetes
        run: |
          kubectl apply -f deployment.yaml
```

Problem Statement 2: Scripting Tasks

1. System Health Monitoring Script (Python Example):

```
import psutil

def check_system_health():
    cpu = psutil.cpu_percent(interval=1)
    memory = psutil.virtual_memory().percent
    disk = psutil.disk_usage('/').percent
    if cpu > 80 or memory > 80 or disk > 80:
        print("ALERT: System metrics exceeding thresholds!")
    else:
        print(f"CPU: {cpu}%, Memory: {memory}%, Disk: {disk}%")

check_system_health()
```

2. Automated Backup Solution (Bash Example):

```
#!/bin/bash
SOURCE_DIR="/path/to/source"
DEST_DIR="/path/to/destination"
BACKUP_NAME="backup_$(date +%Y%m%d).tar.gz"

tar -czf $DEST_DIR/$BACKUP_NAME $SOURCE_DIR && echo "Backup successful!" || echo "Backup failed."
```

3. Log File Analyzer (Python Example):

```
from collections import Counter


def analyze_logs(log_file):
    with open(log_file, 'r') as file:
        logs = file.readlines()
    errors = [line for line in logs if "404" in line]
    print(f"404 Errors: {len(errors)}")
    print("Top Requested Pages:", Counter([line.split()[6] for line in logs]).most_common(5))


analyze_logs("access.log")
```

4. Application Health Checker (Python Example):

```
import requests


def check_app_health(url):
    try:
        response = requests.get(url)
        if response.status_code == 200:
            print("Application is UP!")
        else:
            print("Application is DOWN!")
    except Exception as e:
        print(f"Error: {e}")


check_app_health("http://example.com")
```