**Accuknox QA Trainee Practical Assessment**

**Problem Statement 1:** To complete this project, here’s a breakdown of the steps you would need to follow:

**1. Dockerization:**

* **Create a Dockerfile** for the Wisecow application:
  + Write instructions in the Dockerfile to copy application code from the repository, install dependencies, and expose necessary ports.
  + Ensure that the container is set up to start the application when it runs.
* **Test the Docker Image** locally to ensure it’s running as expected.

**2. Kubernetes Deployment:**

* **Kubernetes Manifest Files**:
  + Create a deployment manifest (deployment.yaml) to define the application pods, replicas, container image, and other configurations.
  + Define a **Kubernetes Service** (service.yaml) to expose the Wisecow application, allowing external access.
* **Namespace and Secrets**:
  + Configure a namespace and Kubernetes secrets for secure configuration management.

**3. Continuous Integration and Deployment (CI/CD):**

* **GitHub Actions Workflow**:
  + Set up a GitHub Actions workflow (.github/workflows/ci-cd.yml) to:
    - Build the Docker image and push it to a container registry (e.g., Docker Hub or GitHub Container Registry).
    - Deploy the Docker image to Kubernetes upon a successful build.
  + Include steps for Kubernetes deployment using kubectl to apply the latest deployment files.

**4. TLS Implementation:**

* **Setup TLS Certificates**:
  + Use a tool like **Cert-Manager** to handle automated TLS certificate provisioning.
  + Configure an **Ingress** resource in Kubernetes to manage HTTPS traffic, enforcing secure communication.

**5. Expected Artifacts:**

* **GitHub Repository Structure**:
  + The repository should contain:
    - **Application Source Code** for Wisecow.
    - The **Dockerfile** for containerization.
    - **Kubernetes Manifest Files** for deployment (deployment.yaml, service.yaml, possibly ingress.yaml).
    - **CI/CD Pipeline Configuration** in the form of a GitHub Actions YAML workflow file.

**End Goal:**

* Successfully containerize, deploy, and secure the Wisecow application in Kubernetes with CI/CD automation and TLS encryption.

This setup will streamline deployment and enhance security, scalability, and maintainability for the Wisecow application.

**Problem Statement 2:**

I don’t have a knowledge of Python. I am proficient in Java. So I make code in java.