SQUARESHIFT

DevOps Assignment

1. Deploy a simple web application using Kubernetes. The web application should consist of a single container that listens on port 8080 and displays a "Hello, World!" message when accessed via a web browser.

   Your solution should include the following components:

   - A Kubernetes Deployment that creates a single replica of the container running the web application.
   - A Kubernetes Service that exposes the web application to the internet.
   - A Kubernetes Ingress that routes traffic from the internet to the Service.

   You should use the nginx ingress controller and provide the necessary configuration files to deploy the application. When completed, you should be able to access the "Hello, World!" message by navigating to the public IP address of your Kubernetes cluster in a web browser.
2. You have been tasked with deploying a multi-tier web application in Kubernetes. The application consists of a web front-end, an application server, and a database server. The web front-end should listen on port 80, the application server should listen on port 8080, and the database server should listen on port 3306.

   Your solution should include the following components:

   - A Kubernetes Deployment for each component of the application, with appropriate labels and selectors.
   - A Kubernetes Service for each component of the application, with appropriate selectors and port mappings.
   - A Kubernetes ConfigMap and Secret for storing application configuration and sensitive information, respectively.
   - A Kubernetes Volume and VolumeClaim for persisting data in the database server.
   - A Kubernetes Ingress that routes traffic from the internet to the web front-end Service.

   You should provide the necessary configuration files to deploy the application

3. Create a ClusterRole with the name api-clusterrole, and create a ClusterRo- leBinding named api-clusterrolebinding. Map the ServiceAccount from the previous step to theAPI resources pods with the operations watch, list, and get.
4. Create a new Secret named basic-auth of type kubernetes.io/basic-auth. Assign the key-value pairs username=super and password=my-s8cr3t. Mount the Secret as a volume with the path /etc/secret and read-only permissions to the Pods controlled by theDeployment.
5. Create a PersistentVolume named logs-pv that maps to the hostPath /tmp/logs. The access mode should be ReadWriteOnce and ReadOnlyMany. Provision a stor- age capacity of 2Gi. Assign the reclaim policy Delete and an empty string as the storage class. Ensure that the status of the PersistentVolume shows Available.