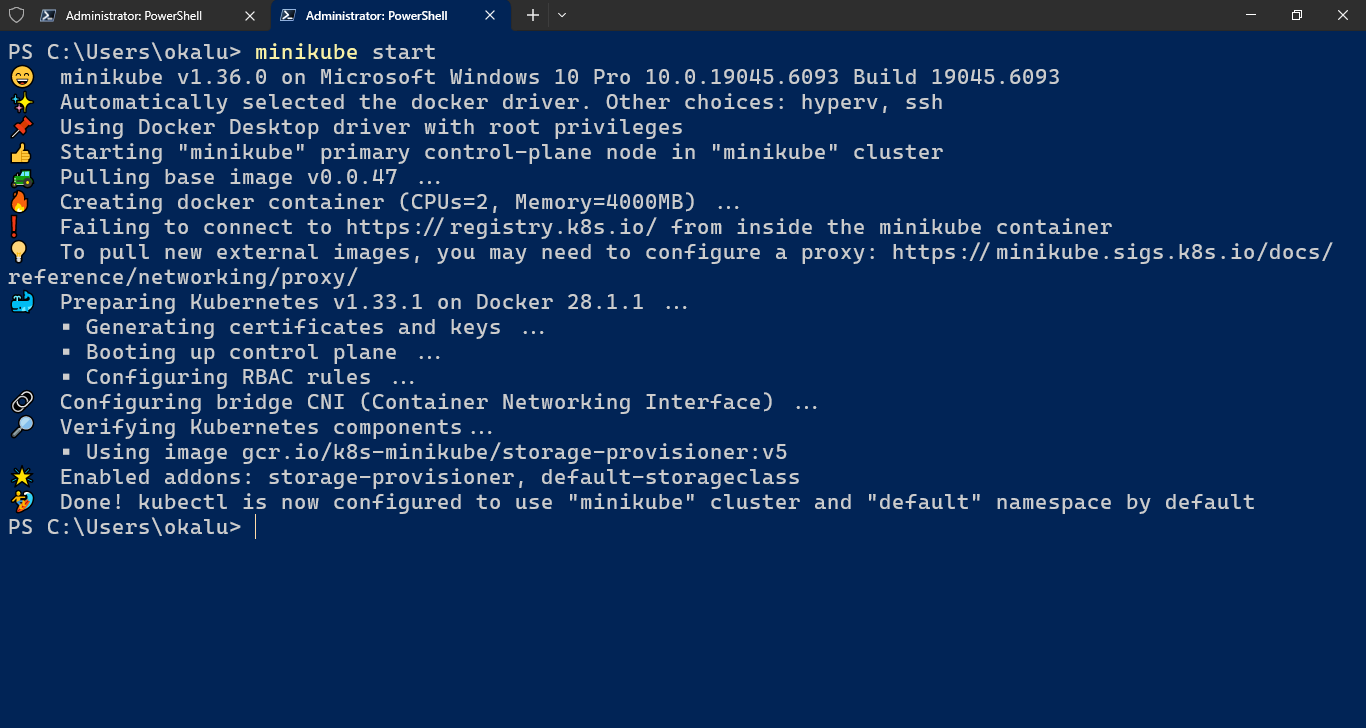
**Deploy a full-stack multi-container Spring Boot WebApp to Local Kubernetes using deployment.yaml file**

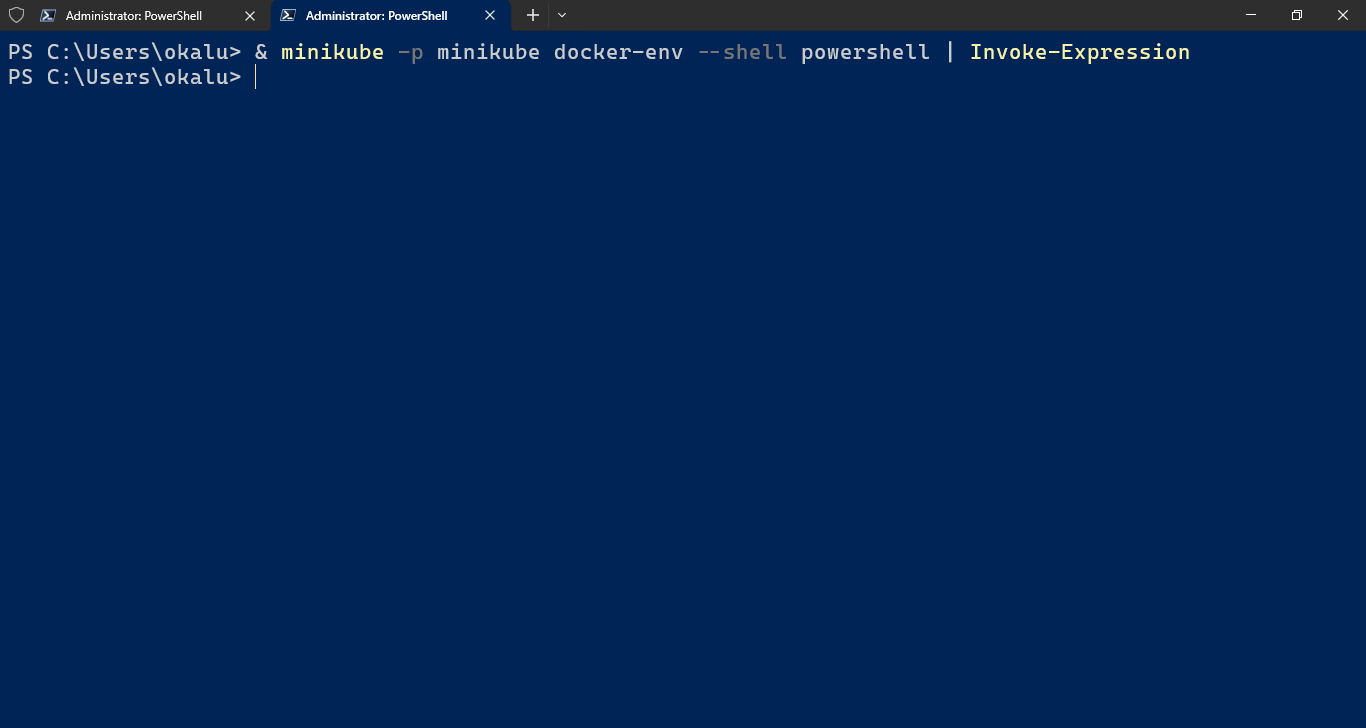
1. First, Start Minikube. Open a command line terminal/shell and execute

> *minikube start*



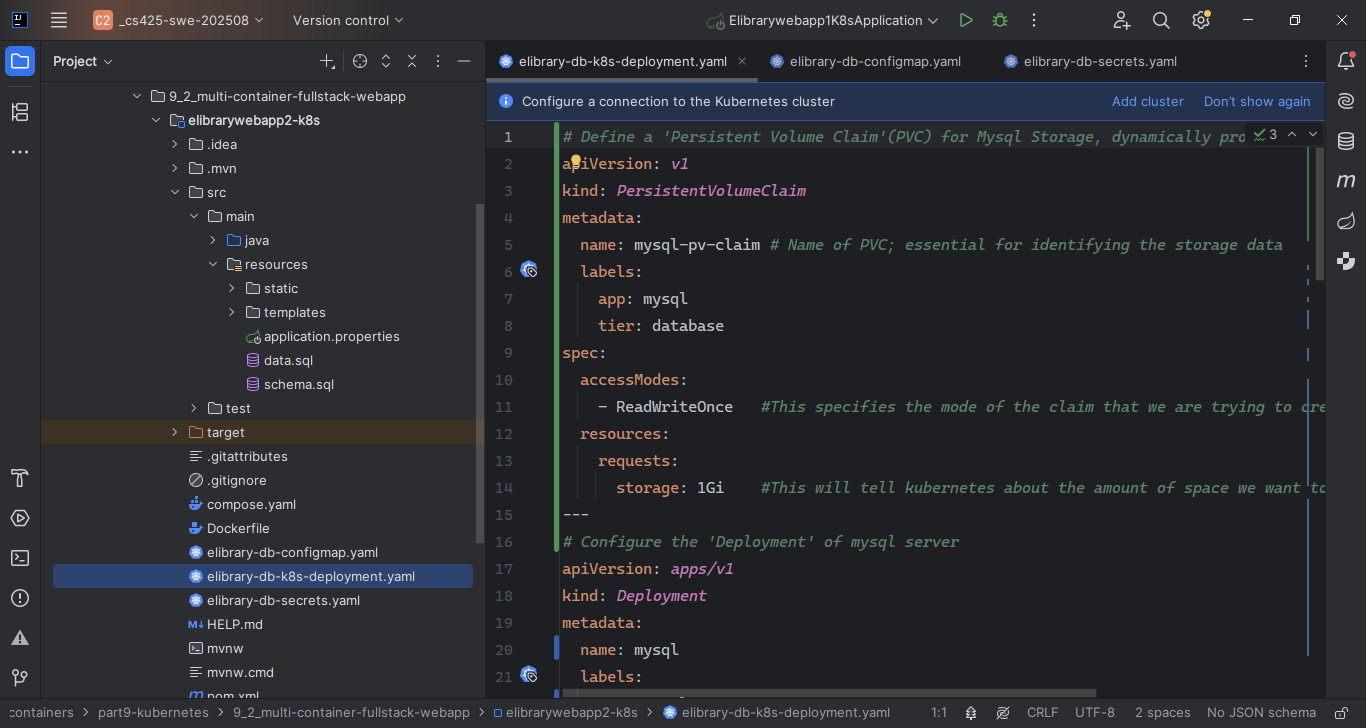
1. Enable Docker operations to execute within the Minikube environment, by configuring the Docker CLI to communicate with the Docker daemon within the Minikube cluster. To do this run the appropriate command for your Shell:
   1. For Windows Powershell:

*& minikube -p minikube docker-env --shell powershell | Invoke-Expression*

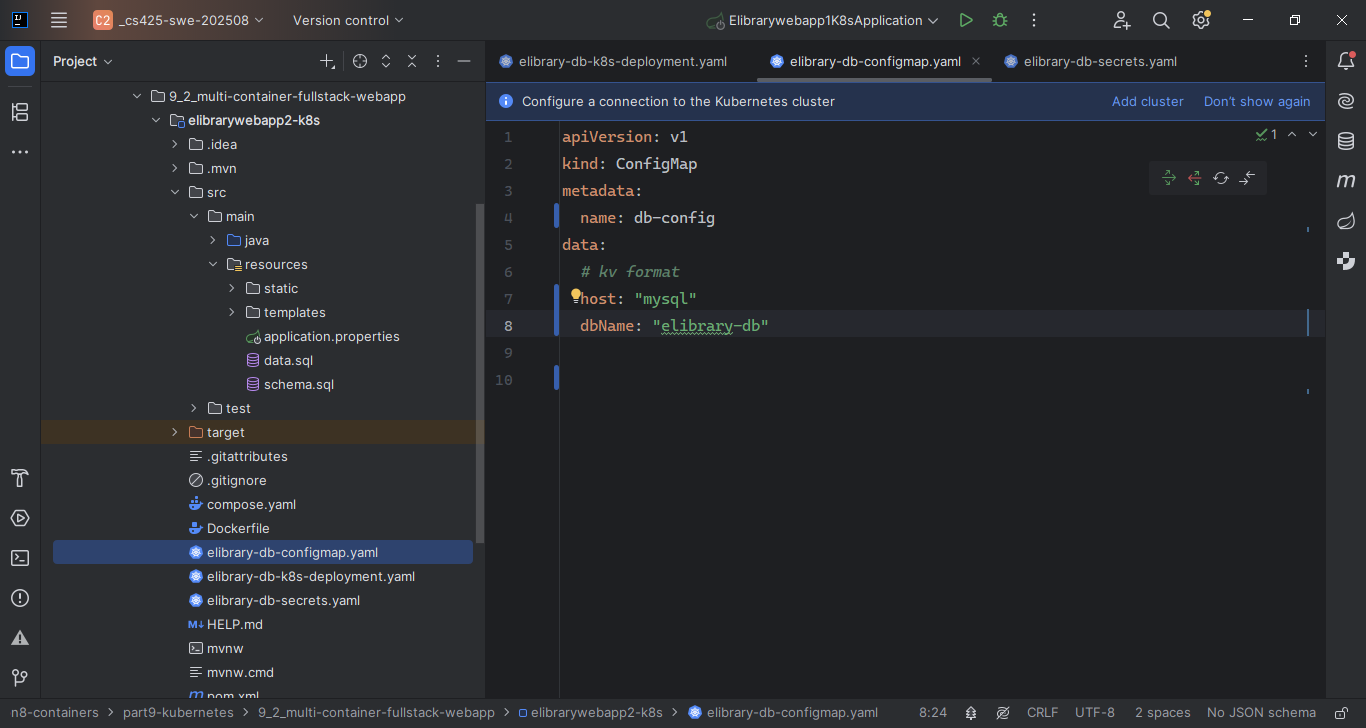


* 1. For Windows CMD shell: ??? (See previous demo)
  2. MacOS bash or zsh: ??? (See previous demo)

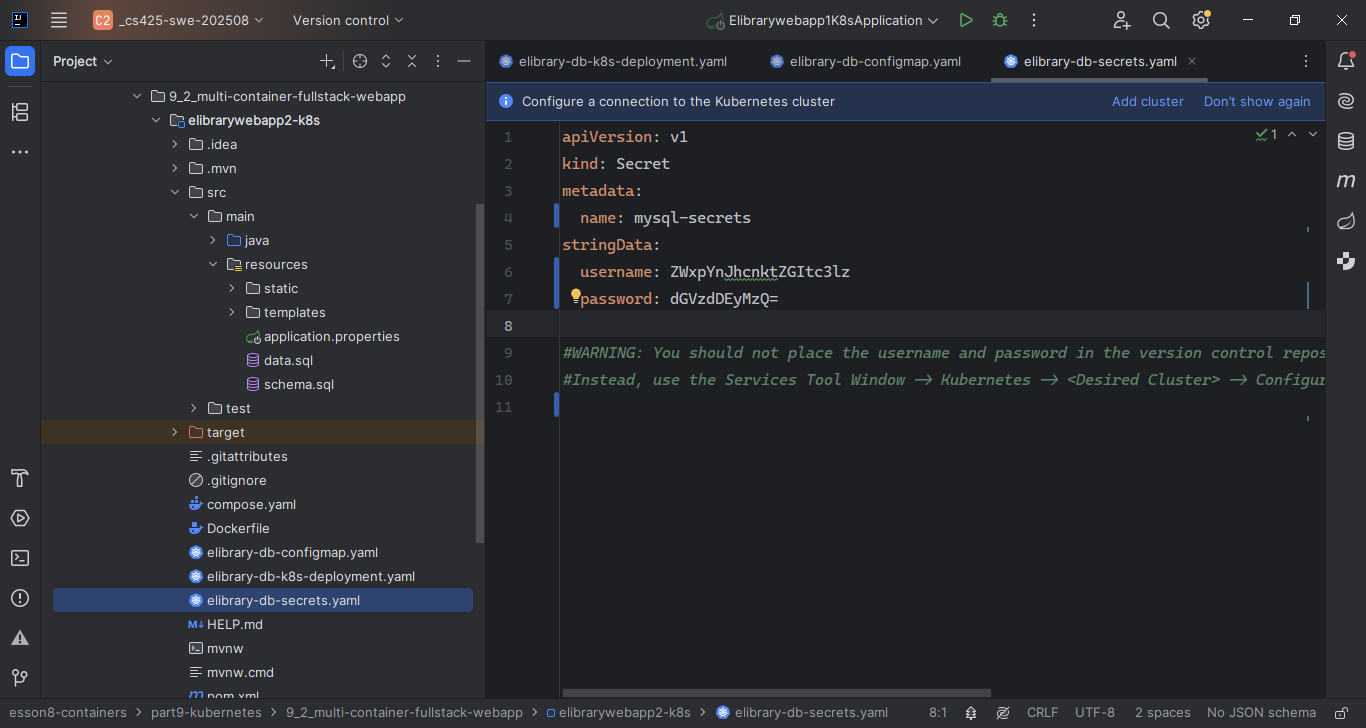
1. Create the database-deployment.yaml file



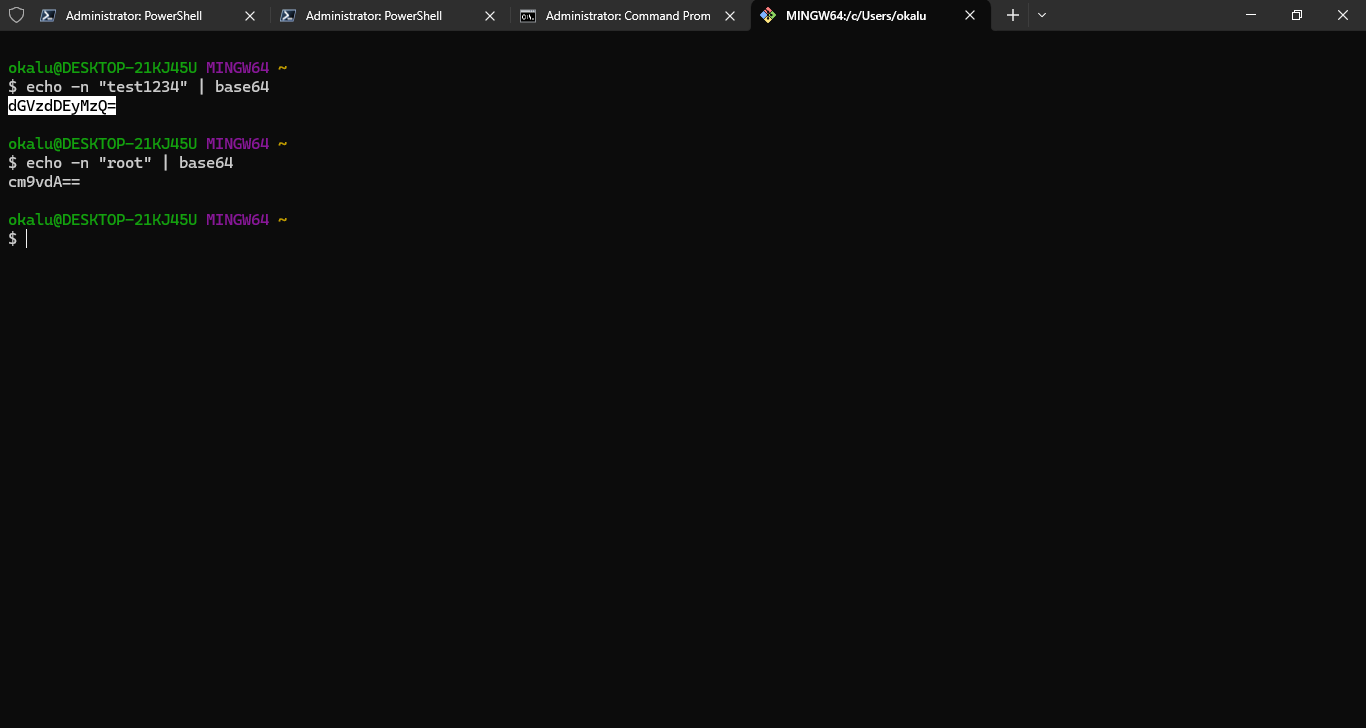
1. Create the configmap.yaml file



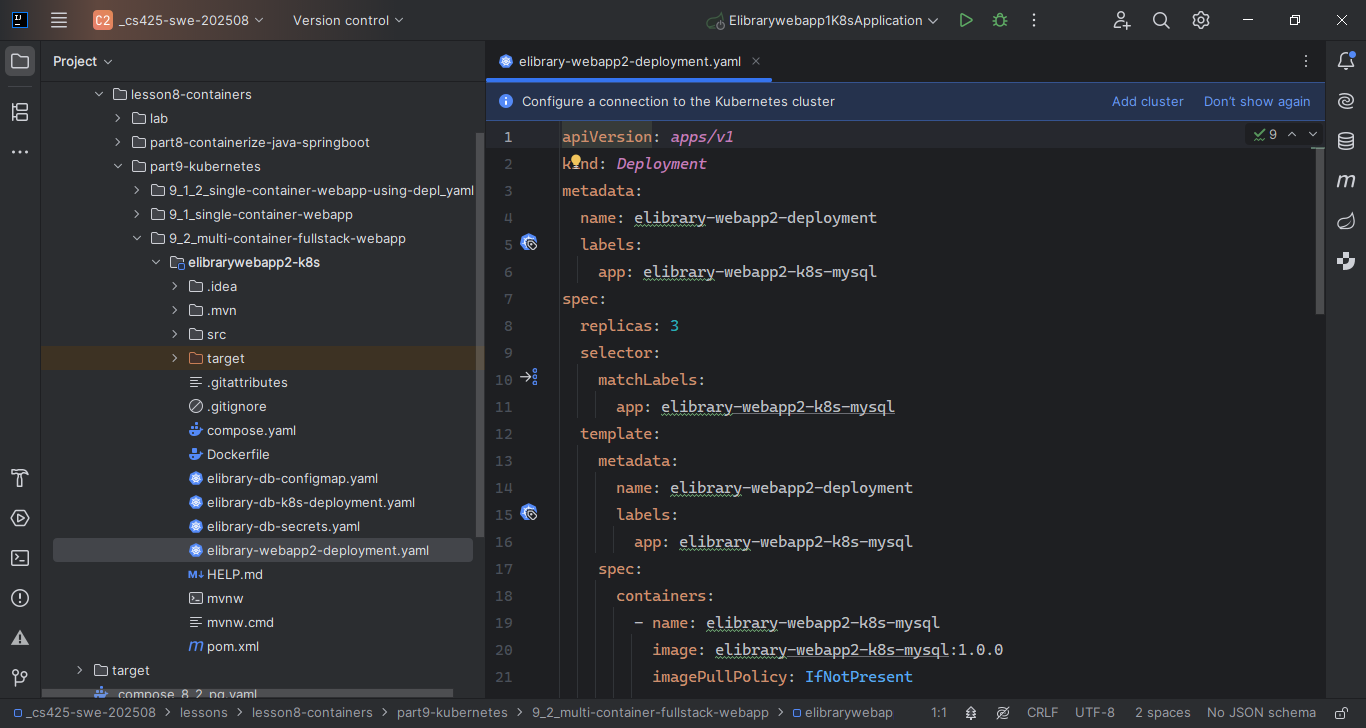
1. Create the secrets.yaml file



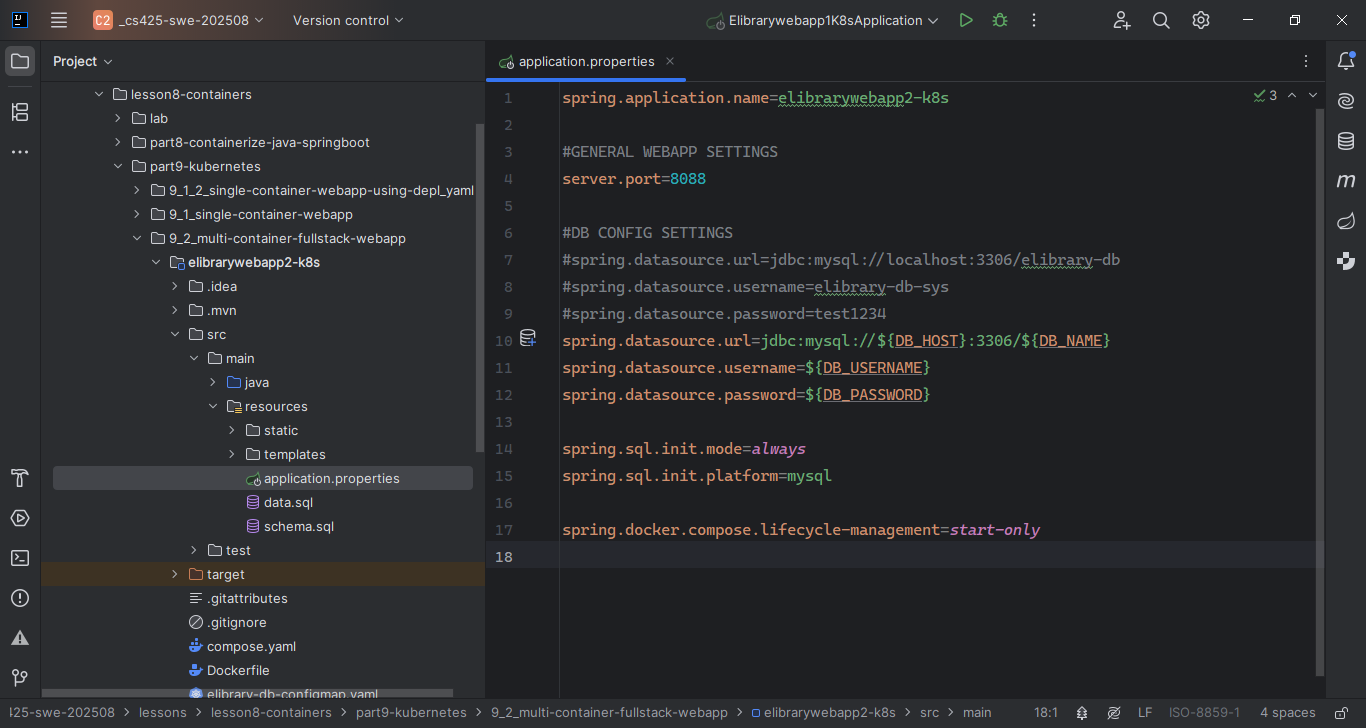
Note: Quick and simple cmd to encrypt plaintext to base64 ciphertext



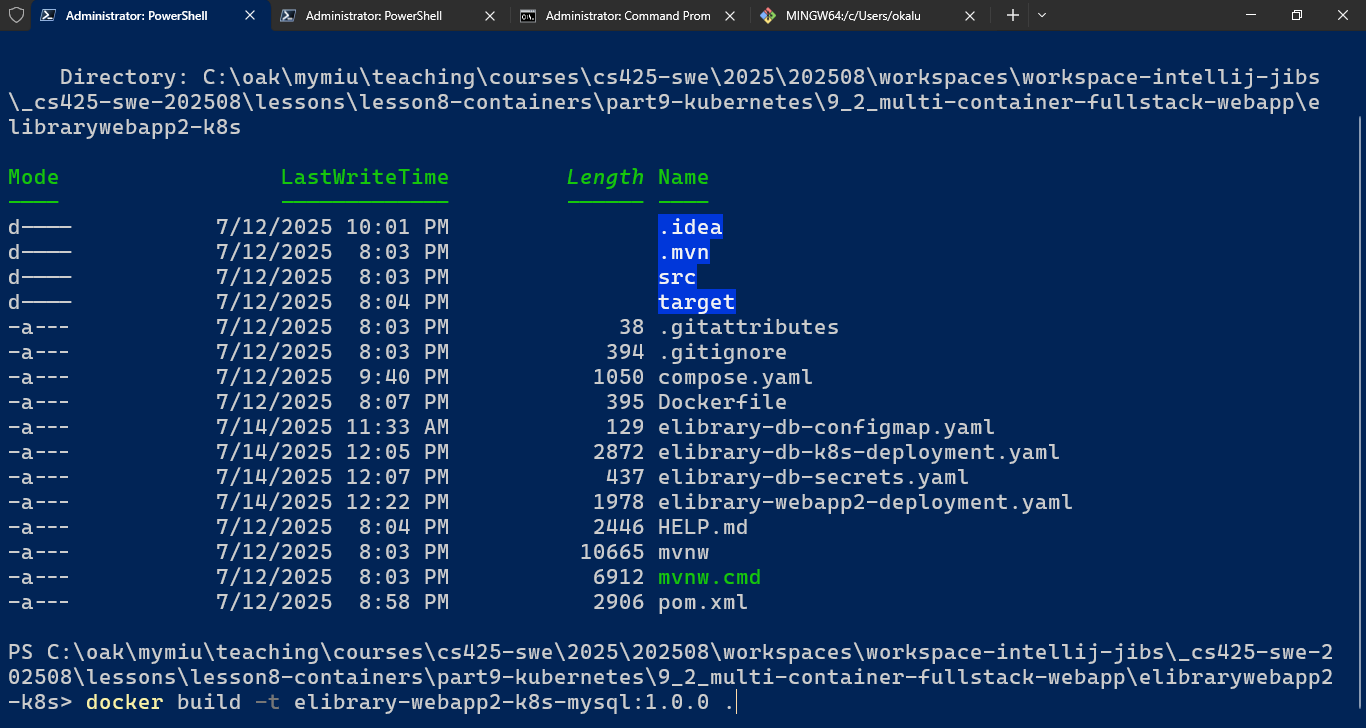
1. Create the webapp-deployment.yaml file

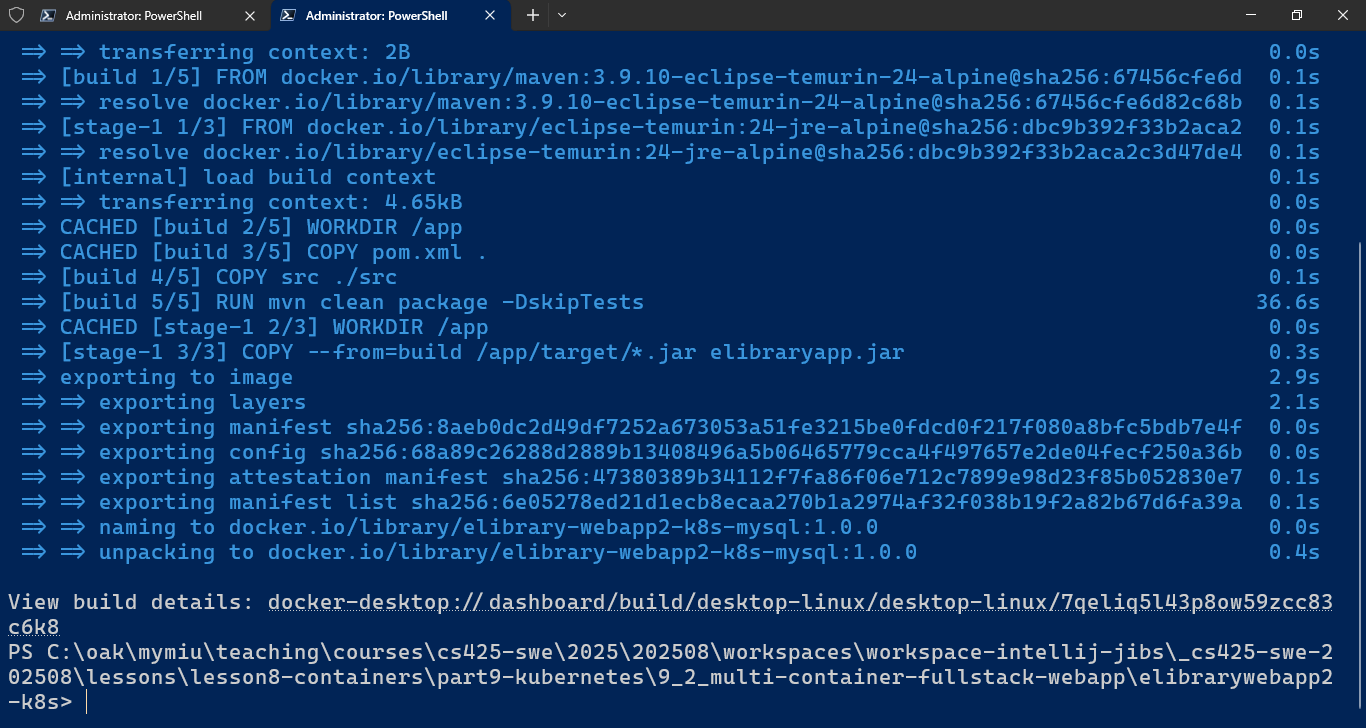


1. Make sure the Webapp’s datasource config settings are using the Environment variables, as specified in the webapp-deployment.yaml

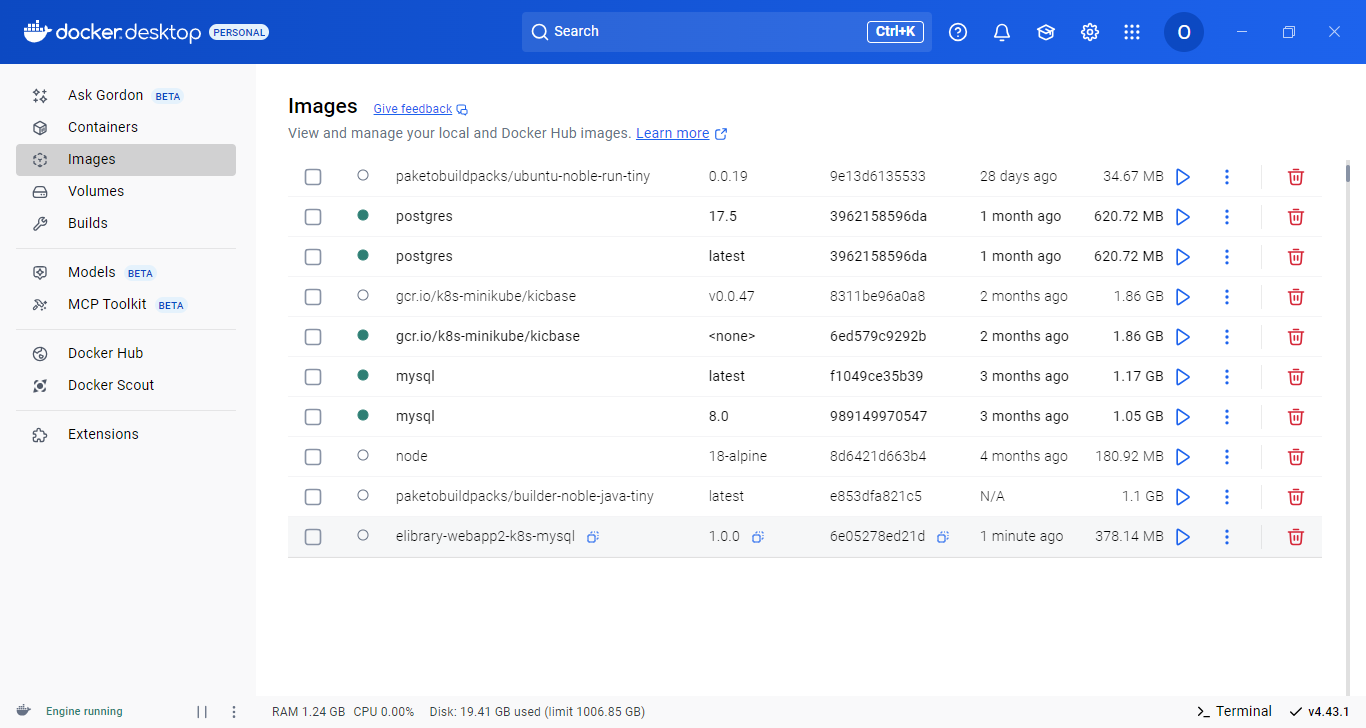


1. Chdir to the webapp project root folder and execute docker build ... cmd to build the image using the provided Dockerfile

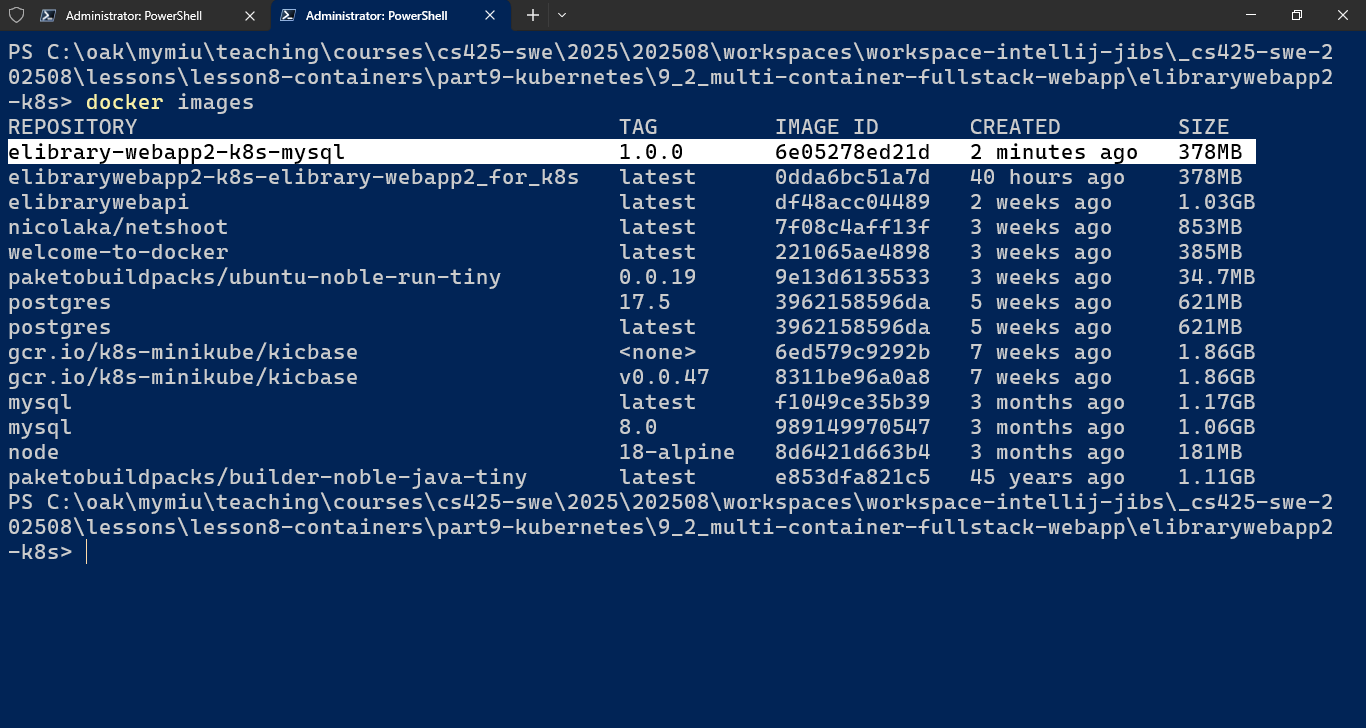




WebApp’s Image built successfully and appears listed in “Images” screen of Docker desktop GUI

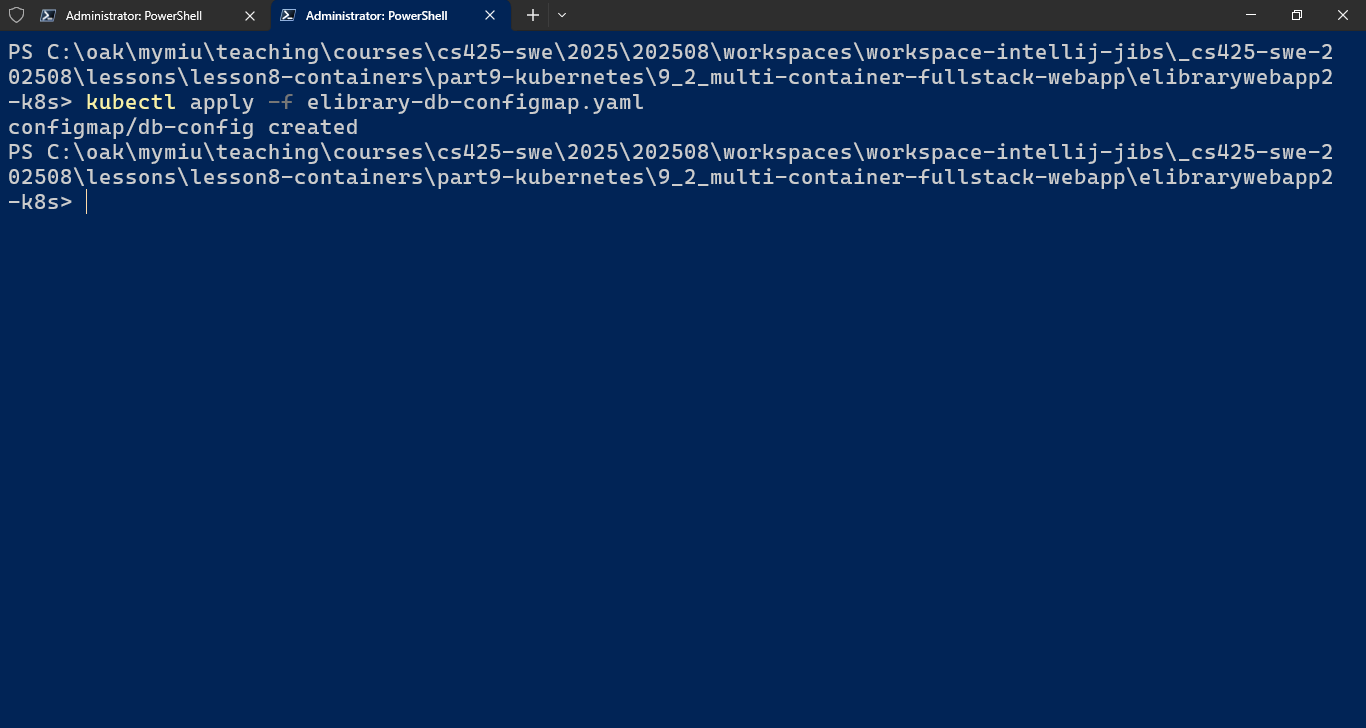


Also, execute cmd > docker images to see the new image listed

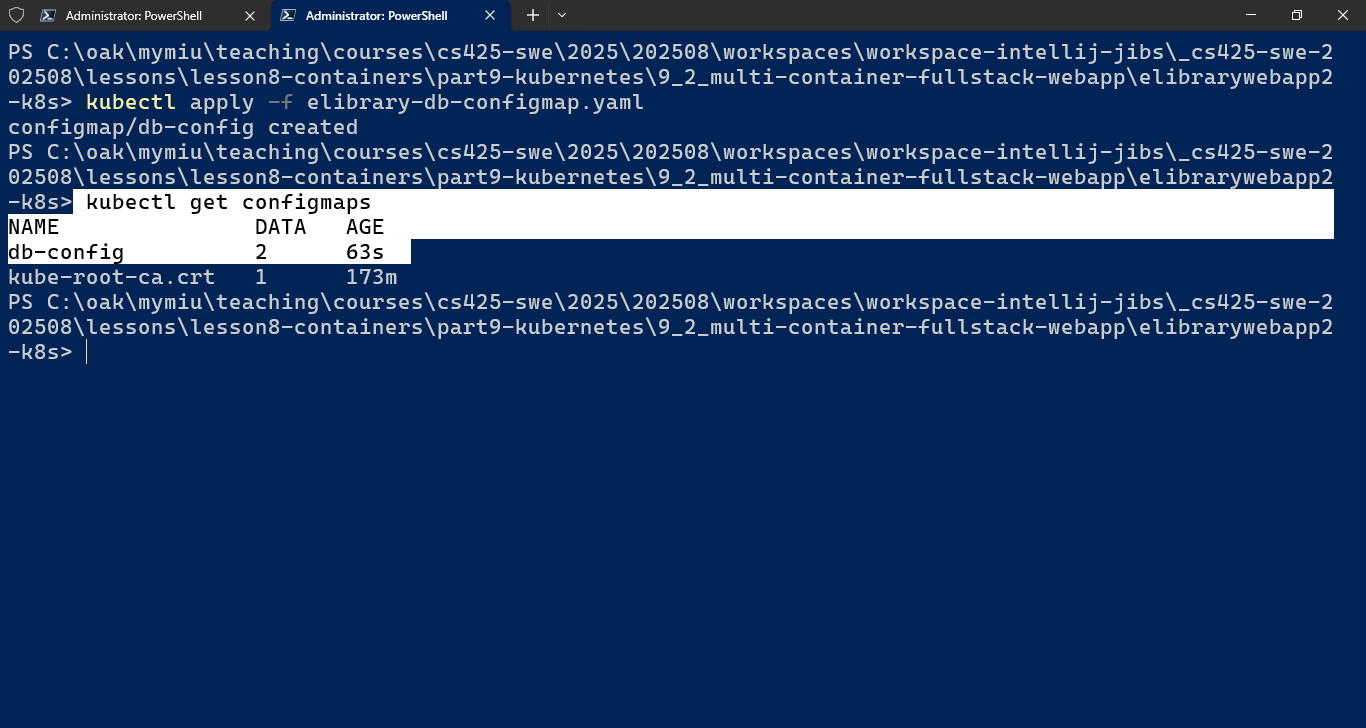


1. Deploy the Config Map object/script

> kubectl apply –f elibrary-db-configmap.yaml

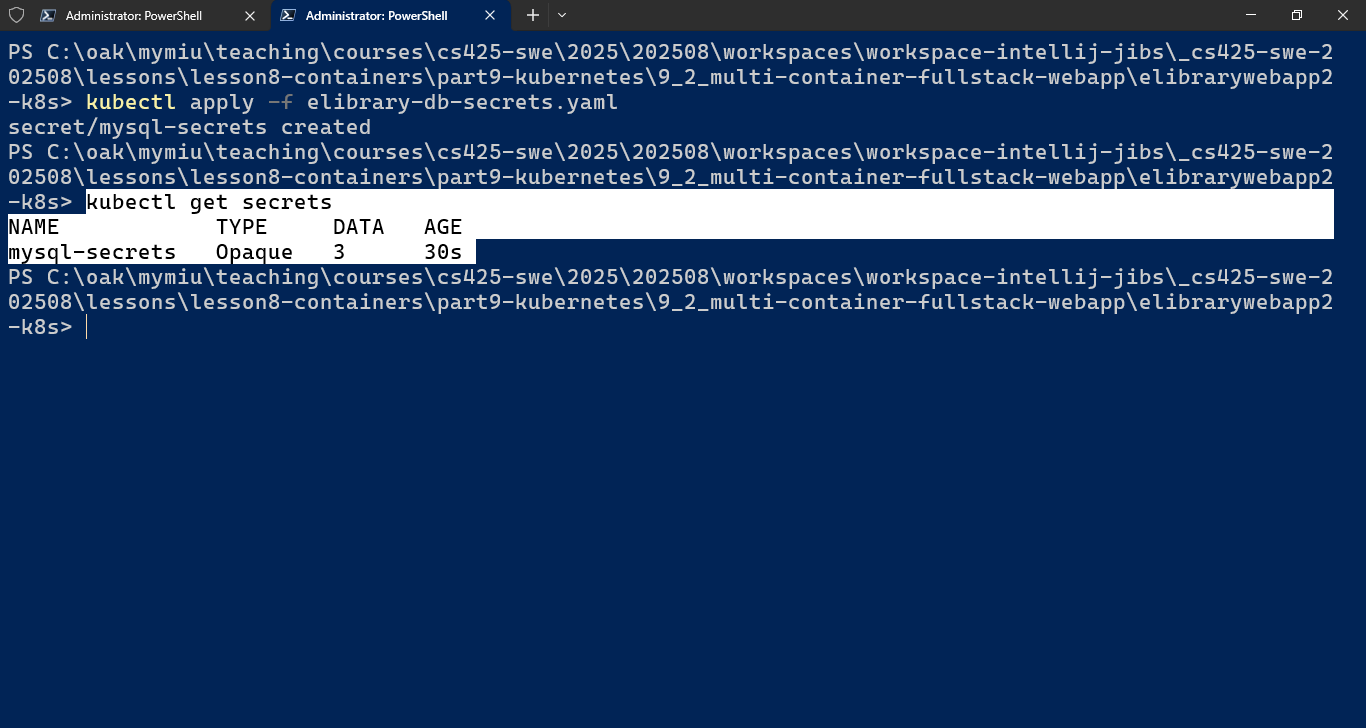
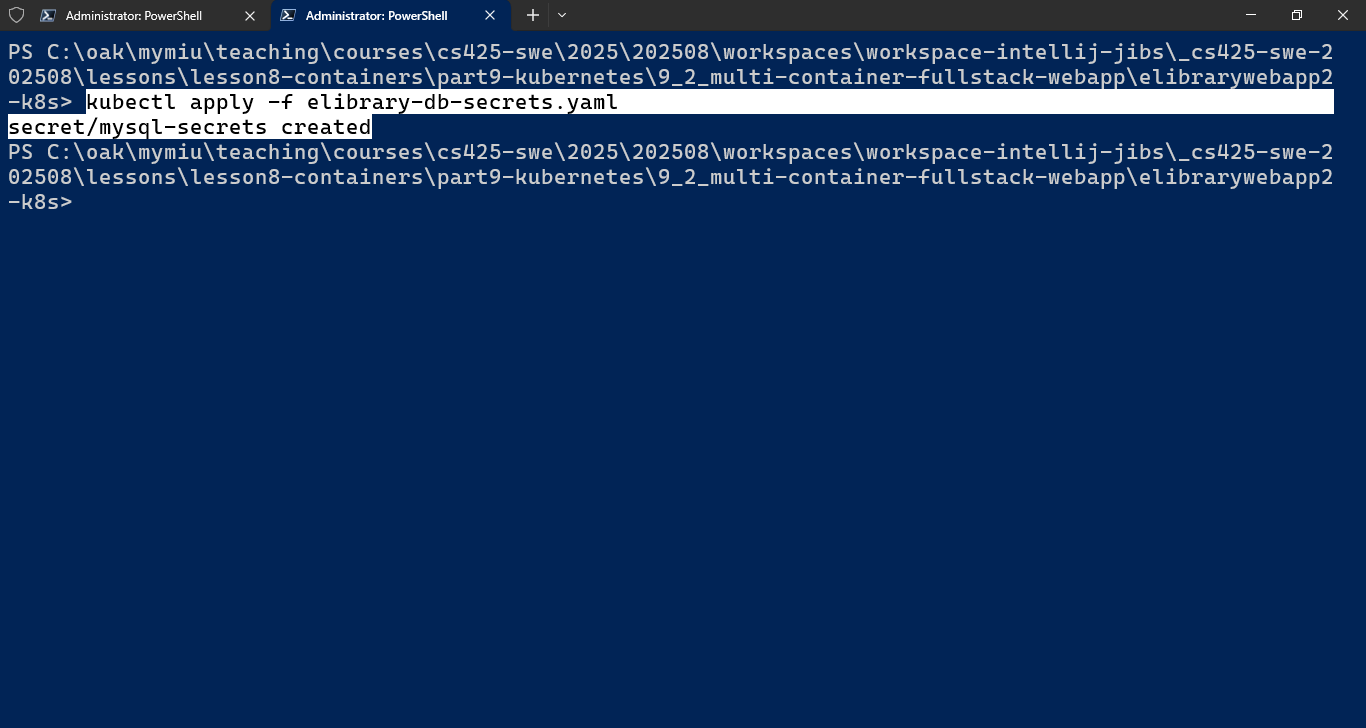


1. Verify the config map has been created by executing cmd > kubectl get configmaps



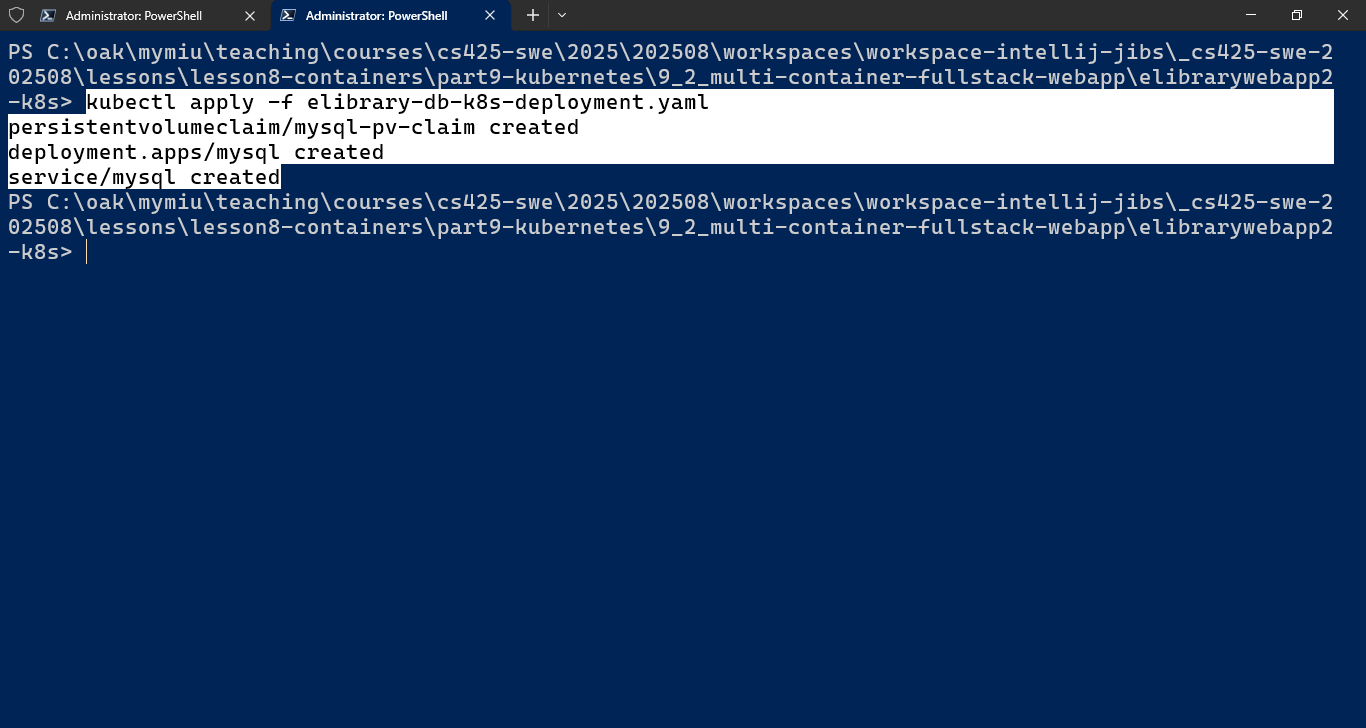
1. Create the secrets object by executing

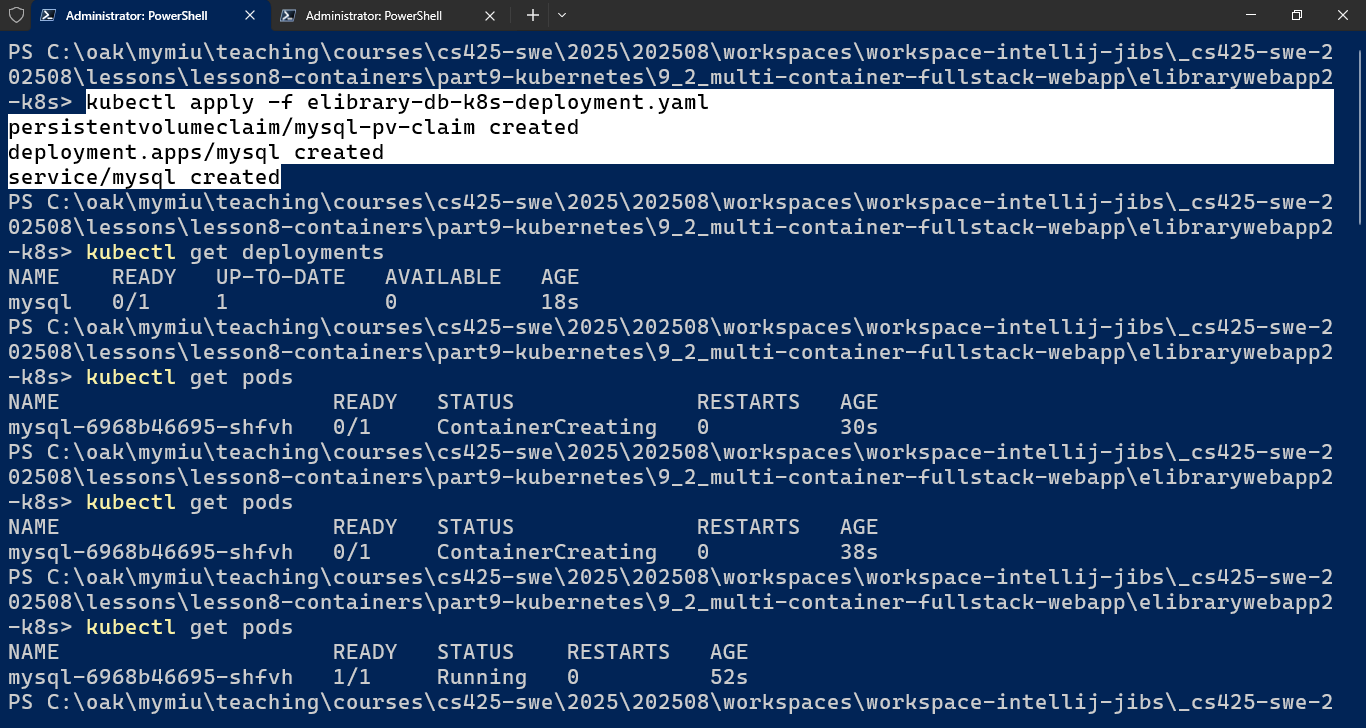
> kubectl apply –f [app-db-secrets.yaml]

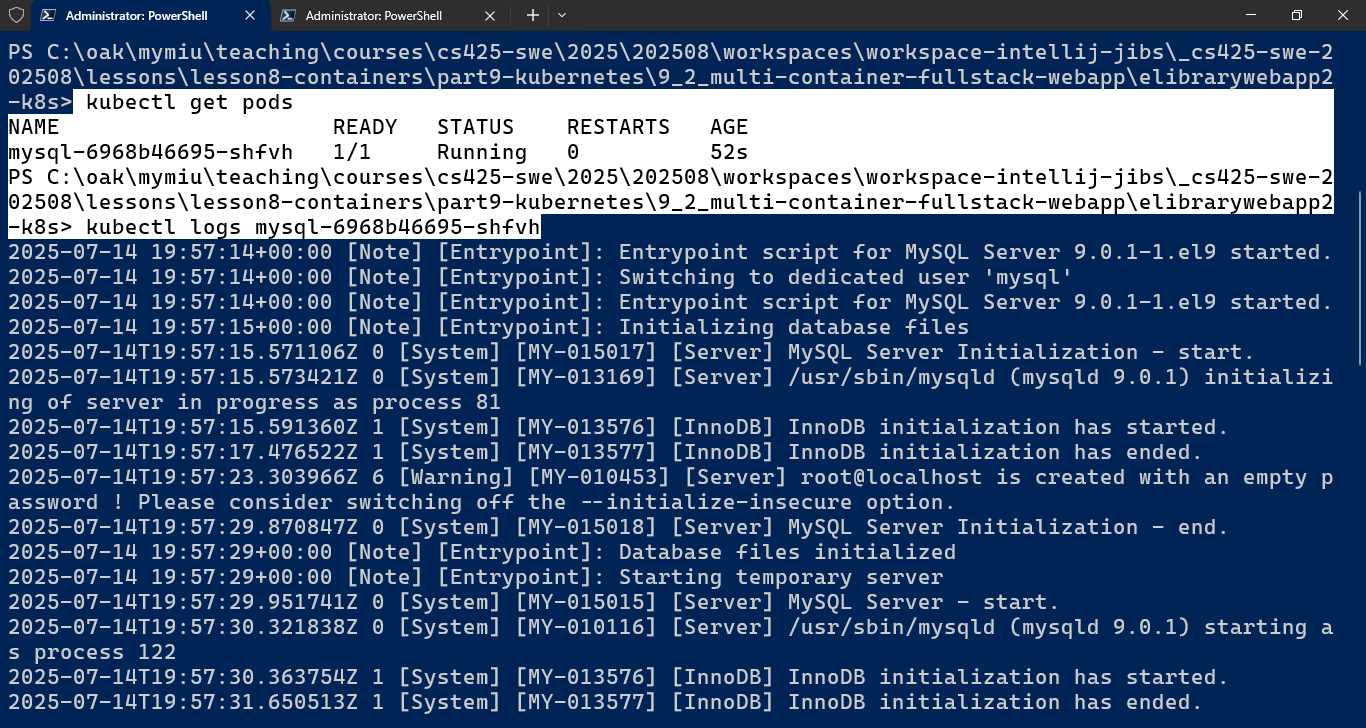


1. Create the Database deployment by executing

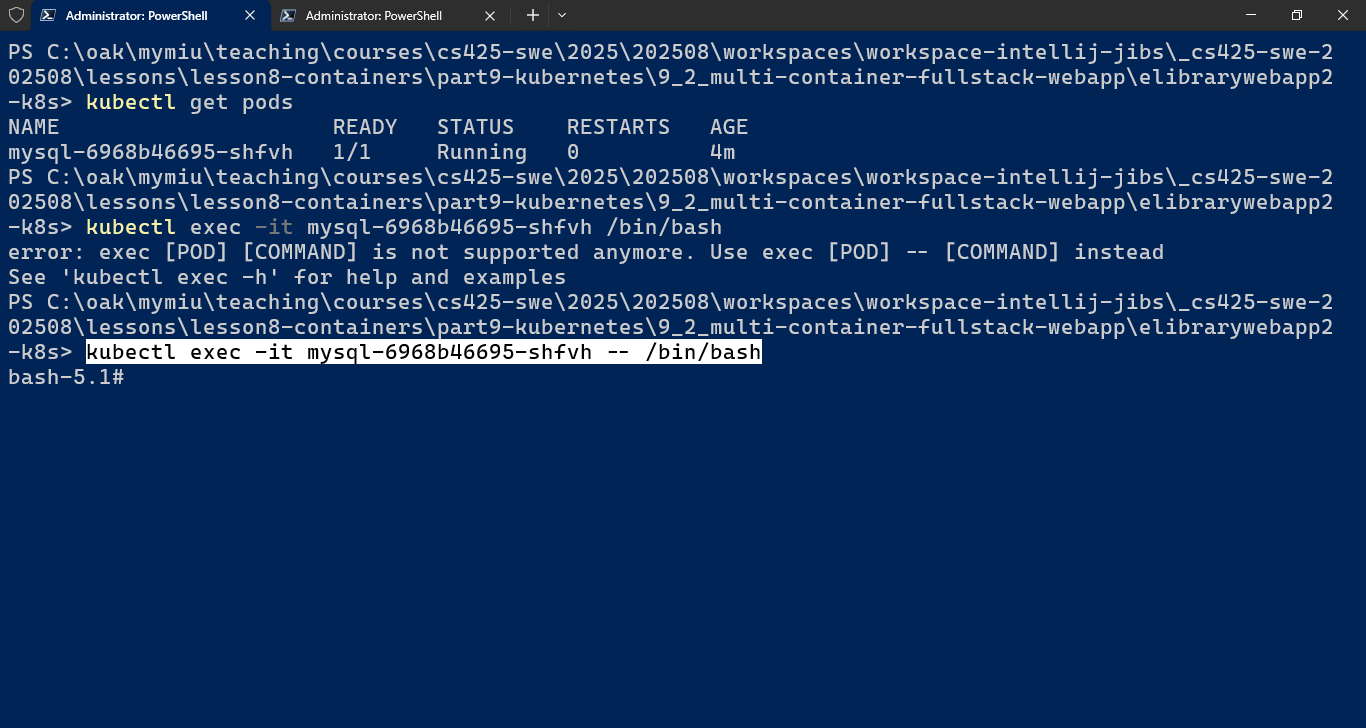
> kubectl apply –f [db-deployment.yaml]



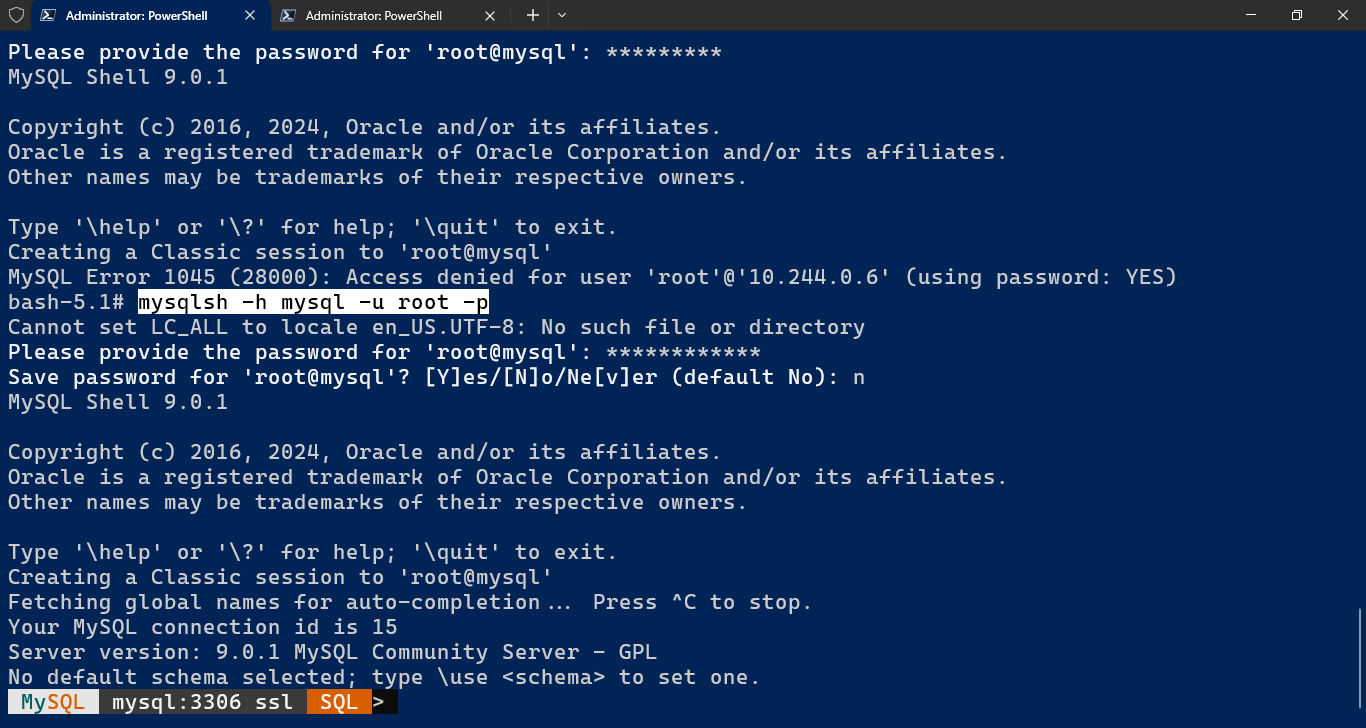


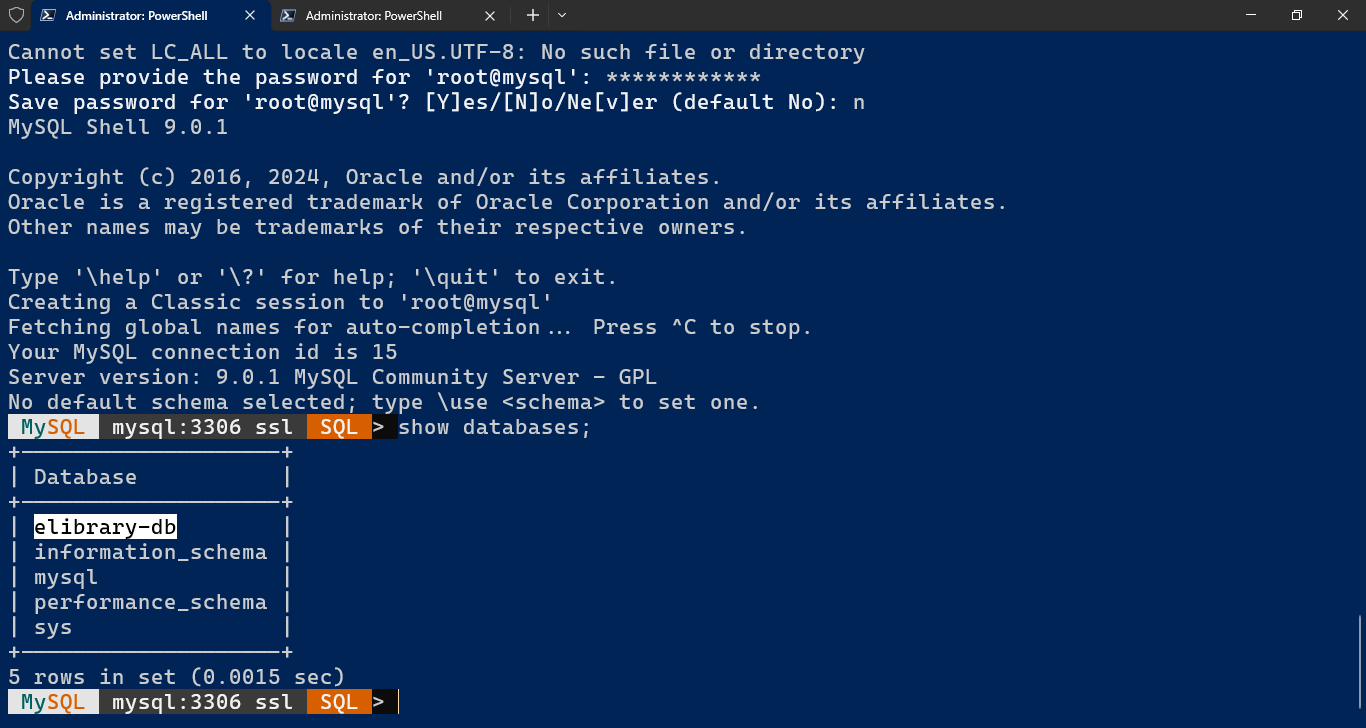


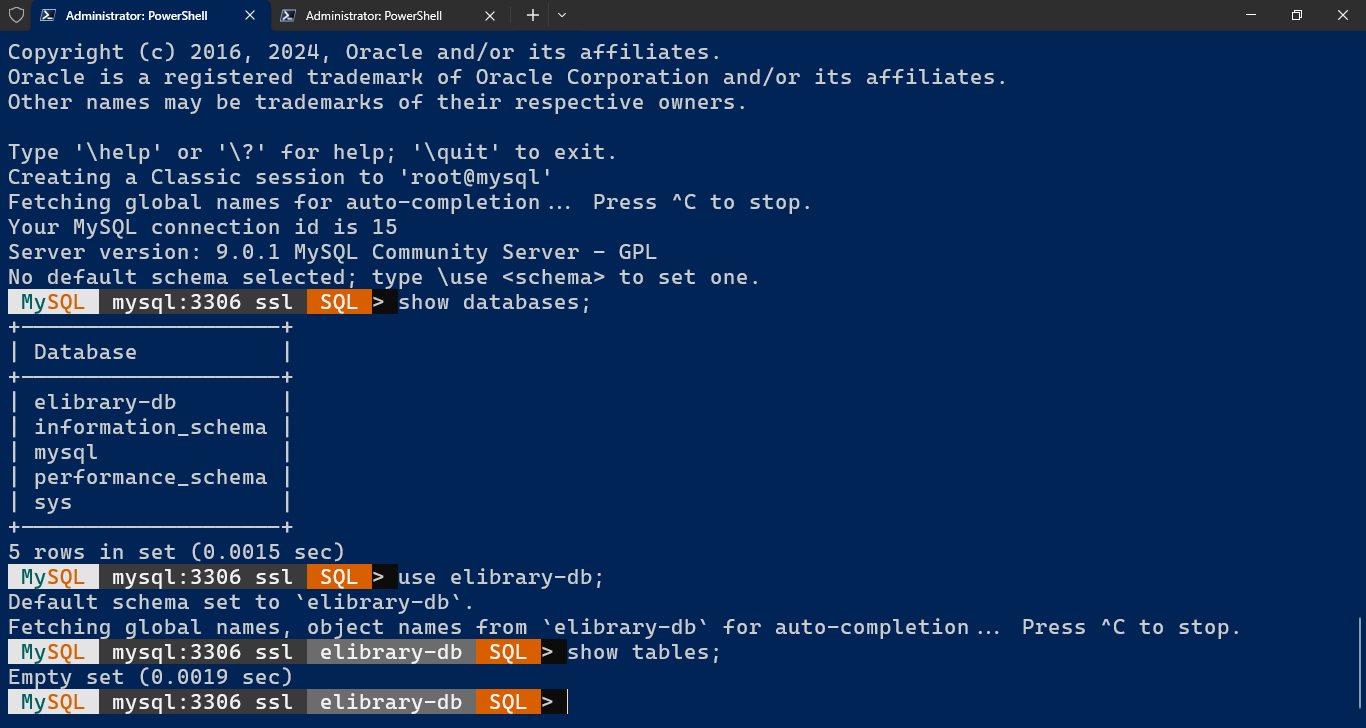
1. Execute a bash shell inside the pod running mysql-9 db



1. Using mysql shell or mysql client, Connect to the mysql db server instance running inside the kubernetes cluster

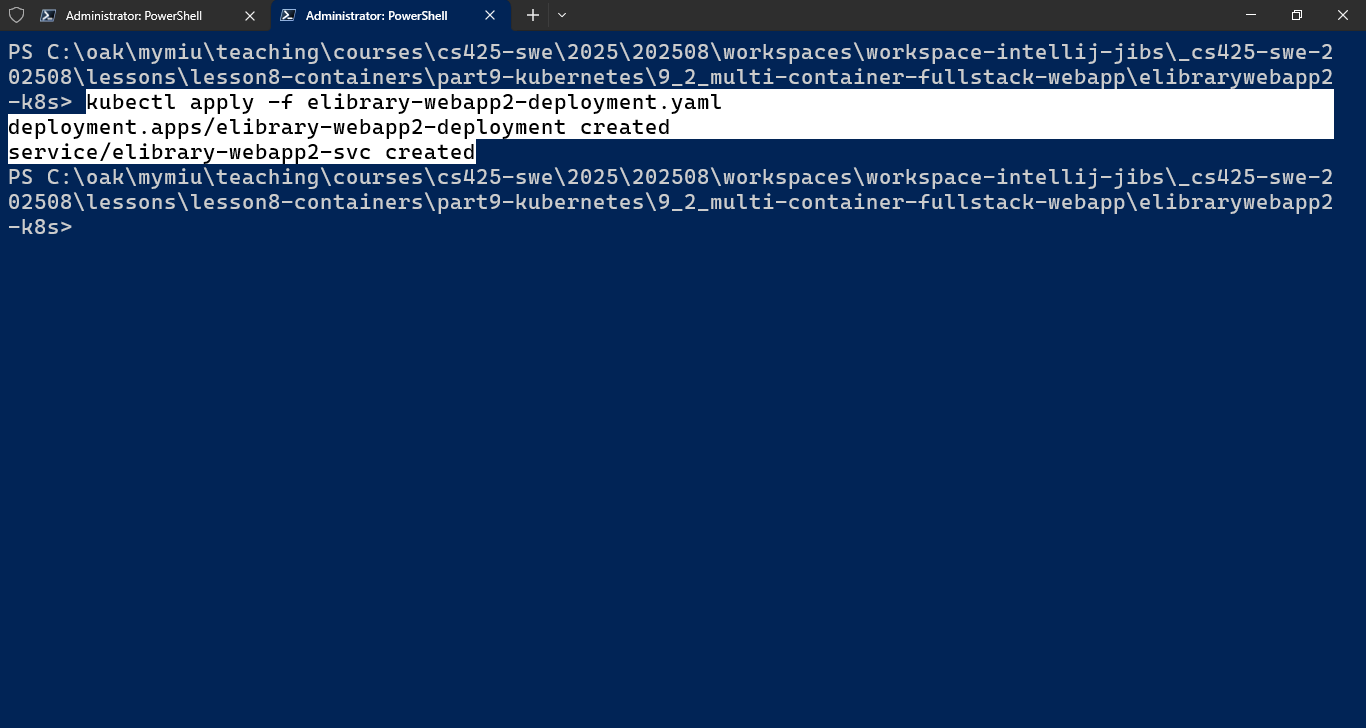


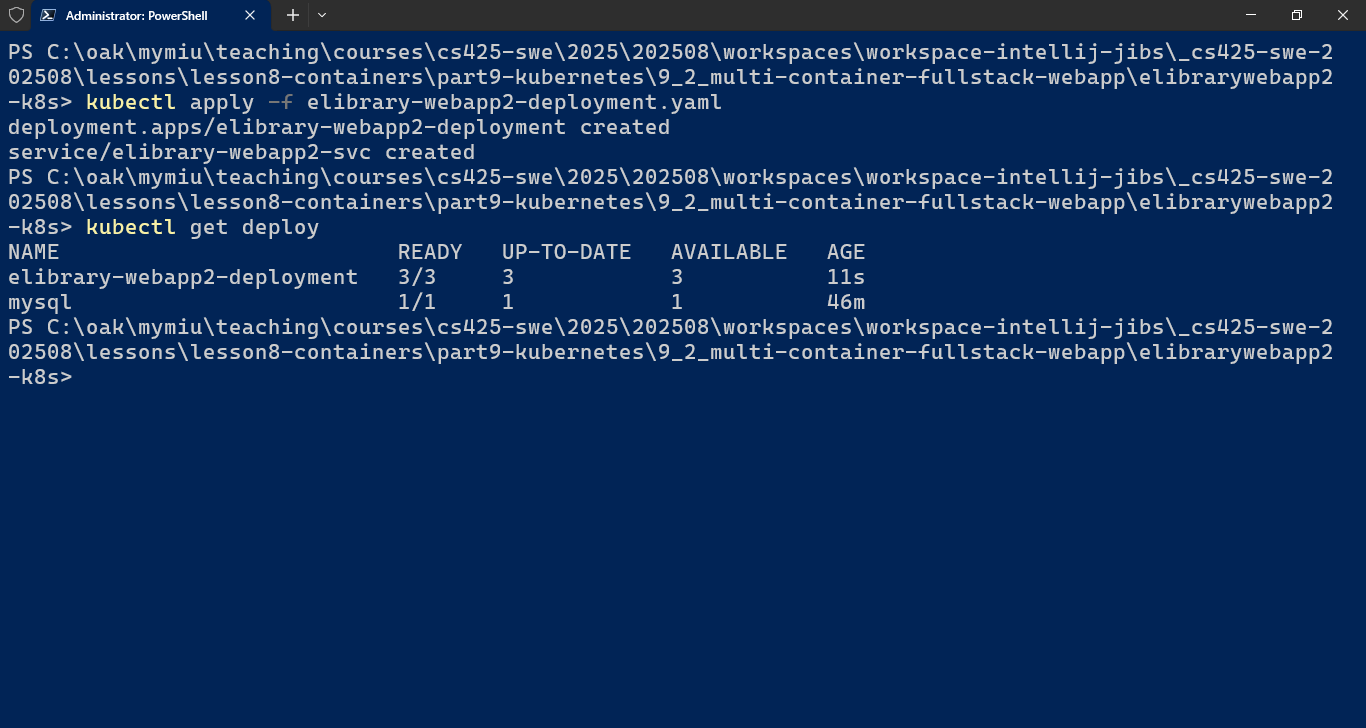




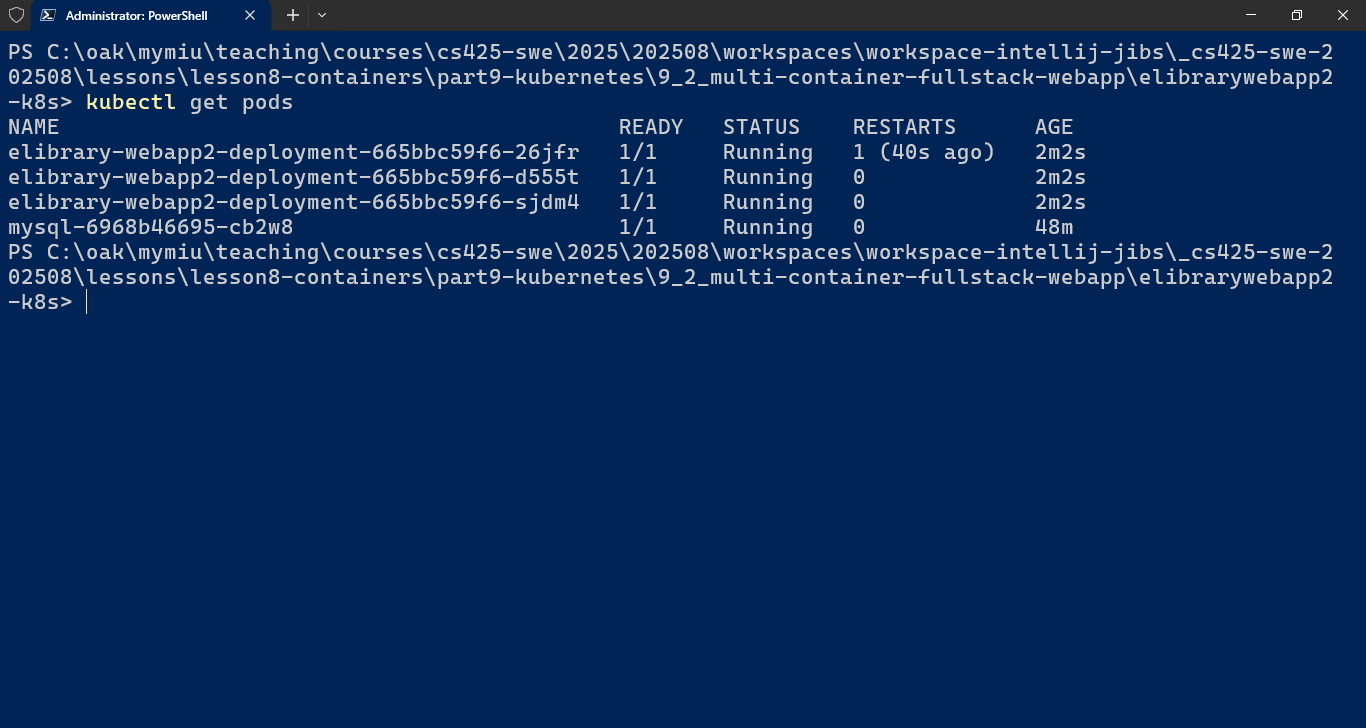
1. Next, create the deployment for the webapp

> kubectl apply –f [webapp-deployment.yaml]

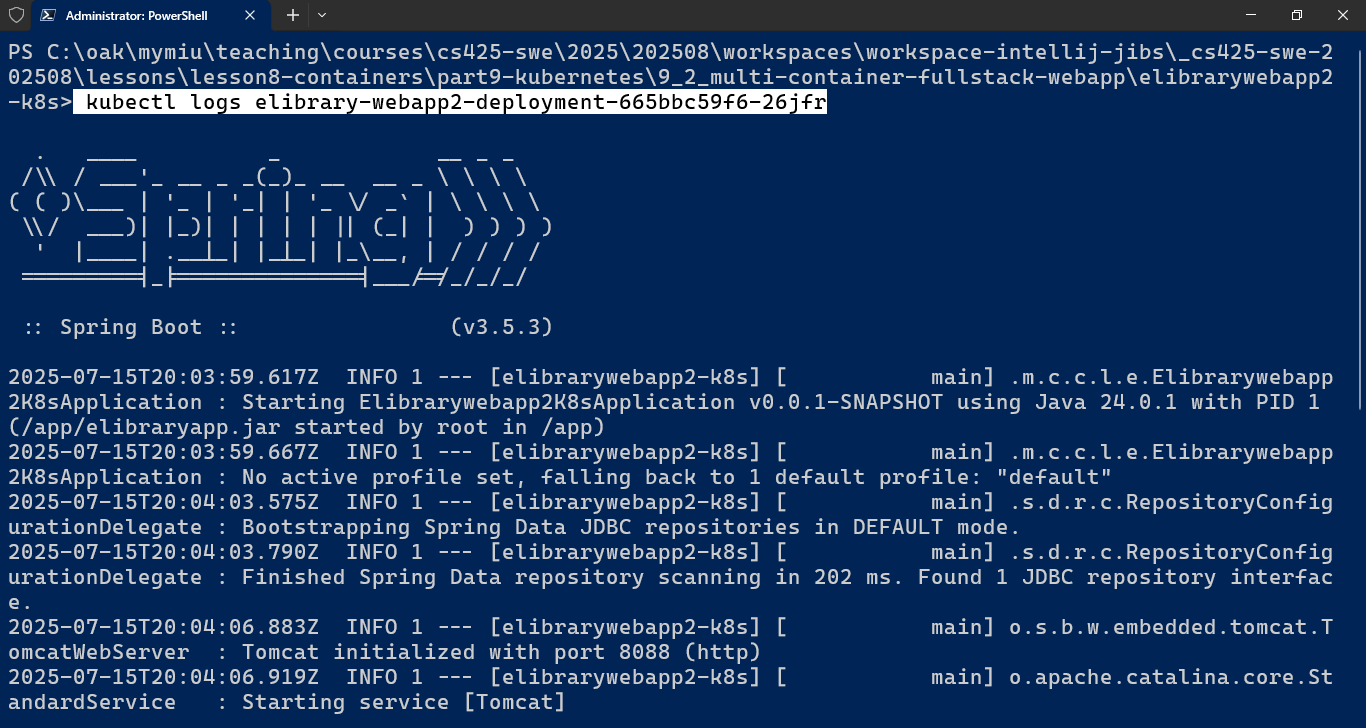




1. See the pods, by executing > kubectl get pods

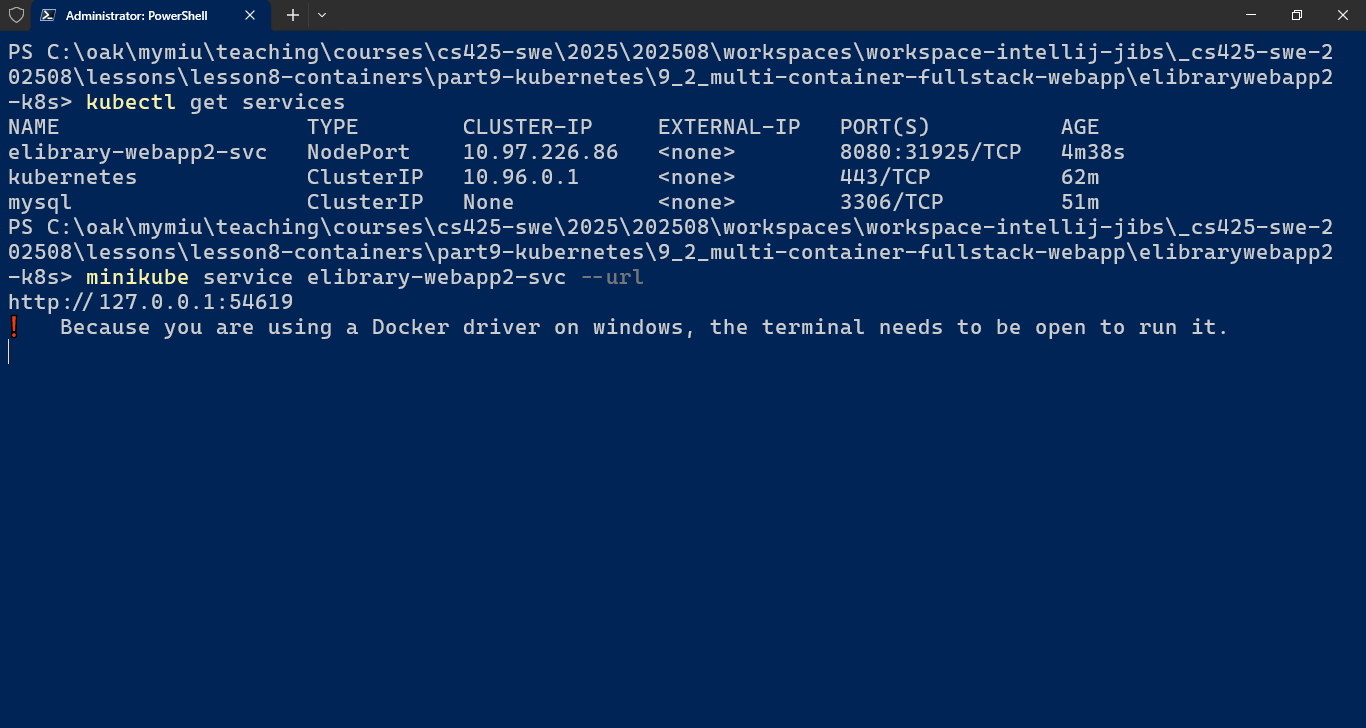


1. See the logs from the pods: > kubectl logs [pod-name]



1. Obtain the URL for accessing the deployed webapp through the service

> minikube service [svc-name] --url



1. ...
2. ...