TP Minikube

Context

Les fichiers de deployment

Mysql-deployment.yaml

apiVersion: apps/v1

kind: Deployment

metadata:

  name: wordpress-mysql

  labels:

    app: wordpress

spec:

  selector:

    matchLabels:

      app: wordpress

      tier: mysql

  strategy:

    type: Recreate

  template:

    metadata:

      labels:

        app: wordpress

        tier: mysql

    spec:

      containers:

      - image: mysql:8.0

        name: mysql

        env:

        - name: MYSQL\_ROOT\_PASSWORD

          value: password

        - name: MYSQL\_DATABASE

          value: wordpress

        - name: MYSQL\_USER

          value: wordpress

        - name: MYSQL\_PASSWORD

          value: password

        ports:

        - containerPort: 3306

          name: mysql

        volumeMounts:

        - name: mysql-persistent-storage

          mountPath: /var/lib/mysql

      volumes:

      - name: mysql-persistent-storage

        persistentVolumeClaim:

          claimName: mysql-pv-claim

Deployment du service wordpress

Sur windows, renseigner l’@IP du service mysql au lieu du nom du service.

#Activer le ingress controller

minikube addons enable ingress

Activer le ingress controller

Deploy wordpress-ingress.yaml

kubectl apply -f .\wordpress-ingress.yaml

minikube addons enable ingress

kubectl get ingress

NAME CLASS HOSTS ADDRESS PORTS AGE

nginx-ingress nginx site1.local,site2.local 192.168.49.2 80 5d11h

wordpress-ingress nginx site.wordpress 192.168.49.2 80 73s

Modifier le fichier hosts avec l’@IP de la machine et nom du host

Cd C:\Windows\System32\Drivers\etc

Code .

#<IP\_Minikube> <nom\_host>

#192.168.49.2 minikube.local.host

127.0.0.1 site1.local

127.0.0.1 site2.local

# Pour Ingress de wordpress-mysql TP minikube

127.0.0.1 site.wordpress

# End of section

Lancer la commande

Minikube tunnel

Vérifier sur un navigateur

<http://localhost/site.wordpress>

