1. Tải docker về

// update

**sudo apt-get update**

<// cai cac goi can thiet

 **apt-transport-https:** Cung cấp khả năng truyền tải các gói phần mềm qua HTTPS, giúp bảo mật khi tải các gói từ các kho lưu trữ bên ngoài.

 **ca-certificates:** Cài đặt các chứng chỉ CA để hệ thống có thể xác thực các kết nối SSL/TLS an toàn.

 **curl:** Công cụ dòng lệnh dùng để chuyển dữ liệu từ hoặc đến một máy chủ, sử dụng nhiều giao thức khác nhau.

 **software-properties-common:** Cung cấp các lệnh tiện ích như add-apt-repository, giúp dễ dàng thêm các kho lưu trữ phần mềm mới.

/>

**sudo apt-get install -y apt-transport-https ca-certificates curl software-properties-common**

<// Thêm Docker GPG key

Docker ký các gói phần mềm của họ bằng GPG key để đảm bảo tính toàn vẹn và bảo mật. Lệnh này tải xuống và thêm GPG key của Docker vào hệ thống, cho phép hệ thống xác thực các gói Docker khi cài đặt.

/>

**curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add –**

</ thêm docker repository

Lệnh này thêm kho lưu trữ chính thức của Docker vào danh sách các nguồn phần mềm của hệ thống. $(lsb\_release -cs) sẽ tự động thay thế bằng tên mã phát hành của Ubuntu (ví dụ: bionic, focal) để đảm bảo rằng bạn đang nhận được phiên bản phù hợp với hệ thống của mình.

/>

**sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu $(lsb\_release -cs) stable"**

<// update

Sau khi thêm Docker repository, cần phải cập nhật lại danh sách gói để hệ thống nhận diện các gói mới có sẵn từ kho lưu trữ của Docker.

/>

**sudo apt-get update**

// cài docker

**sudo apt-get install -y docker-ce**

<// Khởi động Docker và thiết lập Docker chạy tự động khi khởi động hệ thống

 sudo systemctl enable docker: Lệnh này cấu hình Docker để tự động khởi động khi hệ thống khởi động. Điều này giúp Docker luôn sẵn sàng chạy container khi hệ thống bật lên.

 sudo systemctl start docker: Khởi động dịch vụ Docker ngay lập tức, giúp bạn có thể bắt đầu sử dụng Docker ngay sau khi cài đặt.

/>

**sudo systemctl enable docker**

**sudo systemctl start docker**

</ thêm user vào nhóm docker

Nếu bạn dự định sử dụng Docker làm driver cho Minikube, bạn cần thêm user của bạn vào nhóm Docker và khởi động lại phiên làm việc:

Sau khi thực hiện lệnh trên, khởi động lại phiên terminal của bạn hoặc đăng nhập lại.

/>

**sudo usermod -aG docker $USER**

**newgrp docker**

// kiểm tra trạng thái và start

**sudo systemctl status docker**

**sudo systemctl start docker**

1. Cài minikube

// cài công cụ cần thiết

**sudo apt-get update**

**sudo apt-get install -y curl apt-transport-https**

// cài minikube

**curl -LO https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64**

**sudo install minikube-linux-amd64 /usr/local/bin/minikube**

// khởi động minikube

**minikube start --driver=docker**

// cài kubectl

**curl -LO "https://dl.k8s.io/release/v1.26.0/bin/linux/amd64/kubectl"**

**chmod +x ./kubectl**

**sudo mv ./kubectl /usr/local/bin/kubectl**

1. Thiết lập PV và Pod mysql

Tạo 1 file có tên **mysql\_pv\_pvc.yaml** có nội dung sau:

**apiVersion: v1**

**kind: PersistentVolume**

**metadata:**

**name: mysql-pv**

**spec:**

**capacity:**

**storage: 5Gi**

**accessModes:**

**- ReadWriteOnce**

**hostPath:**

**path: /mnt/data**

**storageClassName: manual**

**---**

**apiVersion: v1**

**kind: PersistentVolumeClaim**

**metadata:**

**name: mysql-pvc**

**spec:**

**accessModes:**

**- ReadWriteOnce**

**resources:**

**requests:**

**storage: 5Gi**

**storageClassName: manual**

apply file đó

**kubectl apply –f mysql\_pv\_pvc.yaml**

tạo 1 file **mysql\_pod.yaml** để chạy mysql

**apiVersion: v1**

**kind: Pod**

**metadata:**

**name: mysql**

**spec:**

**containers:**

**- name: mysql**

**image: mysql:5.7**

**env:**

**- name: MYSQL\_ROOT\_PASSWORD**

**value: ""**

**ports:**

**- containerPort: 3306**

**volumeMounts:**

**- mountPath: /var/lib/mysql**

**name: mysql-storage**

**volumes:**

**- name: mysql-storage**

**persistentVolumeClaim:**

**claimName: mysql-pvc**

cập nhật :

**kubectl apply –f mysql\_pod.yaml**

1. Truy cập và điền dữ liệu trong pod đã chạy

Dùng lệnh sau để kết nối đến mysql

**kubectl exec –it mysql bash**

**mysql –u root –p**

sau bước này là ta có thể truy cập vào ứng dụng mysql của chúng ta. Giờ hãy thực hiện sử dụng cơ sở dữ liệu này để thực hiện các công việc mong muốn

ví dụ ta có file sample\_data.sql

**-- Tạo cơ sở dữ liệu mới**

**CREATE DATABASE IF NOT EXISTS SampleDB;**

**USE SampleDB;**

**-- Tạo bảng `Users`**

**CREATE TABLE IF NOT EXISTS Users (**

**id INT AUTO\_INCREMENT PRIMARY KEY,**

**username VARCHAR(50) NOT NULL,**

**email VARCHAR(100) NOT NULL,**

**age INT NOT NULL,**

**registered\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP**

**);**

**-- Chèn 10 dòng dữ liệu vào bảng `Users`**

**INSERT INTO Users (username, email, age) VALUES ('alice', 'alice@example.com', 25);**

**INSERT INTO Users (username, email, age) VALUES ('bob', 'bob@example.com', 30);**

**INSERT INTO Users (username, email, age) VALUES ('carol', 'carol@example.com', 28);**

**INSERT INTO Users (username, email, age) VALUES ('dave', 'dave@example.com', 35);**

**INSERT INTO Users (username, email, age) VALUES ('eve', 'eve@example.com', 22);**

**INSERT INTO Users (username, email, age) VALUES ('frank', 'frank@example.com', 40);**

**INSERT INTO Users (username, email, age) VALUES ('grace', 'grace@example.com', 29);**

**INSERT INTO Users (username, email, age) VALUES ('heidi', 'heidi@example.com', 31);**

**INSERT INTO Users (username, email, age) VALUES ('ivan', 'ivan@example.com', 33);**

**INSERT INTO Users (username, email, age) VALUES ('judy', 'judy@example.com', 27);**

điền dữ liệu:

**kubectl cp sample\_data.sql mysql:/tmp/sample\_data.sql**

**kubeclt exec –it mysql – bash**

**mysql –u root –p < tmp/sample\_data.sql**