***# Cập nhật danh sách gói***

sudo apt-get update

***# Cài đặt Docker***

sudo apt-get install -y docker.io

***# Khởi động Docker***

sudo systemctl start docker

***# Thiết lập Docker khởi động cùng hệ thống***

sudo systemctl enable docker

***# Thêm người dùng hiện tại vào nhóm Docker***

sudo usermod -aG docker $USER

***# Tải Docker Compose***

sudo curl -L "https://github.com/docker/compose/releases/latest/download/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose

***# Cấp quyền thực thi cho Docker Compose***

sudo chmod +x /usr/local/bin/docker-compose

***# Kiểm tra phiên bản Docker***

docker --version

***# Kiểm tra phiên bản Docker Compose***

docker-compose --version

***# Tạo thư mục cho dự án Docker Compose***

mkdir mongo-docker-compose

cd mongo-docker-compose

***# Tạo file docker-compose.yml***

nano docker-compose.yml

***# Tạo key file để xác thực***

openssl rand -base64 756 > mongo-keyfile

chmod 400 mongo-keyfile

***# KHỞI ĐỘNG LẠI WSL2 sau đó chạy câu lệnh sau để khởi động các container***

docker-compose up -d

***# Setup cấu hình MongoDB bao gồm các container: config, shard1, shard2, shard3, mongos***

***# Tạo file docker-compose.yml với các cấu hình cho config (cổng 27019), shard (cổng 27018), mongos (cổng 27017)***

***# Cấu hình Config Server***

docker exec -it configsvr mongosh --port 27019

rs.initiate({

\_id: "configReplSet",

configsvr: true,

members: [{ \_id: 0, host: "configsvr:27019" }]

})

***# Cấu hình Shard1 (tương tự cho shard2 và shard3)***

docker exec -it shard1 mongosh --port 27018

rs.initiate({

\_id: "shard1ReplSet",

members: [{ \_id: 0, host: "shard1:27018" }]

})

***# Cấu hình Mongos***

docker exec -it mongos mongosh

sh.addShard("shard1ReplSet/shard1:27018")

sh.addShard("shard2ReplSet/shard2:27018")

sh.addShard("shard3ReplSet/shard3:27018")

use admin

db.createUser({

user: "admin",

pwd: "admin",

roles: [ { role: "root", db: "admin" } ]

})

***# Kiểm tra trạng thái router***

docker exec -it mongos mongosh

sh.status()

**Extend**

***# Container***

***# Liệt kê tất cả các container, bao gồm cả những container đã dừng***

docker ps -a

***# Khởi động một container cụ thể***

docker start <container\_name>

***# Dừng tất cả các container đang chạy***

docker stop $(docker ps -q)

***# Xóa tất cả các container***

docker rm $(docker ps -a -q)

***# Xóa tất cả các container đã dừng để giải phóng không gian***

docker container prune

***# Kiểm tra các tiến trình đang sử dụng cổng 27017***

sudo lsof -i :27017

***# Xem log của một container cụ thể***

docker logs <container\_name>

***# Volume***

***# Liệt kê tất cả các volume***

docker volume ls

***# Xóa tất cả các volume không sử dụng***

docker volume prune

***# Xóa một volume cụ thể***

docker volume rm <volume\_name>

***# Image***

***# Liệt kê tất cả các image***

docker images

***# Xóa một image cụ thể***

docker rmi <image\_id>

***# Xóa tất cả các image không sử dụng***

docker image prune -a

***# Docker Compose***

***# Liệt kê tất cả các dịch vụ Docker Compose***

docker-compose ls

***# Dừng và xóa tất cả các container, network và volume được tạo bởi Docker Compose***

docker-compose down

***# Dừng và xóa tất cả các container, network, giữ lại volumes***

docker-compose down --volumes

***# Backup***

***# Tạo bản sao lưu của MongoDB và lưu vào file backup.gz trong container***

docker exec mongo-container mongodump --archive=/data/backup.gz

***# Sao chép file backup.gz từ container ra máy chủ***

docker cp mongo-container:/data/backup.gz ./backup.gz

***# Restore***

***# Sao chép file backup.gz từ máy chủ vào container***

docker cp ./backup.gz mongo-container:/data/backup.gz

***# Khôi phục dữ liệu MongoDB từ file backup.gz trong container***

docker exec mongo-container mongorestore --archive=/data/backup.gz

***# Truy cập từ máy khách***

from pymongo import MongoClient

# Kết nối tới mongos router trên Máy A (192.168.1.100)

client = MongoClient("mongodb://192.168.1.100:27017")

# Truy cập vào database "myDatabase"

db = client.myDatabase

# Tạo bộ sưu tập (collection) và thêm dữ liệu

collection = db.myCollection

collection.insert\_one({"name": "test", "value": 123})