# =============================================

# 🐳 DOCKER COMMANDS - FOOTBALL STATS PLATFORM

# =============================================

# 🚀 TRƯỜNG HỢP 1: CHƯA SETUP PORT TRONG DOCKERFILE

# (Ứng dụng chạy trên port 8080 mặc định trong container)

# Build image

docker build -f infra/docker/FSF.Dockerfile -t football-stats-backend .

# Chạy container - Map host:5000 → container:8080

docker run -d -p 5000:8080 -e ASPNETCORE\_ENVIRONMENT=Development --name football-backend football-stats-backend

# Truy cập: http://localhost:5000/swagger/index.html

# 🔧 TRƯỜNG HỢP 2: ĐÃ SETUP PORT 5000 TRONG DOCKERFILE

# (Sau khi thêm ENV ASPNETCORE\_URLS=http://+:5000 trong Dockerfile dưới EXPOSE)

# Build lại image

docker build -f infra/docker/FSF.Dockerfile -t football-stats-backend .

# Chạy container - Map host:5000 → container:5000

docker run -d -p 5000:5000 -e ASPNETCORE\_ENVIRONMENT=Development --name football-backend football-stats-backend

# Truy cập: http://localhost:5000/swagger/index.html

# 📊 MONITORING & DEBUGGING

# List containers

docker ps

# Xem logs real-time

docker logs -f football-backend

# Xem logs với timestamps

docker logs -ft football-backend

# Xem resource usage

docker stats

# Vào container để debug

docker exec -it football-backend /bin/bash

# Kiểm tra process trong container

docker exec football-backend ps aux

# Kiểm tra port đang lắng nghe trong container

docker exec football-backend netstat -tulpn

# Test API từ bên ngoài

curl http://localhost:5000/swagger/index.html

# ⚡ CONTAINER MANAGEMENT

# Dừng container

docker stop football-backend

# Khởi động lại container

docker restart football-backend

# Xóa container

docker rm football-backend

# Xóa image

docker rmi football-stats-backend

# Xem tất cả containers (kể cả stopped)

docker ps -a

# 🧹 CLEANUP COMMANDS

# Dừng và xóa tất cả containers

docker stop $(docker ps -aq) && docker rm $(docker ps -aq)

# Xóa tất cả images không dùng

docker image prune -a

# Xóa tất cả (containers, images, networks, volumes)

docker system prune -a --volumes

# 🛠️ DOCKER COMPOSE (Khi có đầy đủ services)

# Chạy tất cả services

docker-compose -f infra/docker/docker-compose.yml up --build

# Chỉ chạy backend service

docker-compose -f infra/docker/docker-compose.yml up --build backend

# Chạy ở detached mode

docker-compose -f infra/docker/docker-compose.yml up -d

# =============================================

# 📝 CẤU HÌNH DOCKERFILE HOÀN CHỈNH

# =============================================

# Thêm dòng này vào Dockerfile để fix port vĩnh viễn:

# ENV ASPNETCORE\_URLS=http://+:5000

# Vị trí: Trong phần "base" stage, sau EXPOSE

# FROM mcr.microsoft.com/dotnet/aspnet:8.0 AS base

# WORKDIR /app

# EXPOSE 5000

# EXPOSE 5001

# ENV ASPNETCORE\_URLS=http://+:5000 # ← THÊM DÒNG NÀY

# =============================================

# 🔍 TROUBLESHOOTING

# =============================================

# Nếu gặp lỗi port đã được sử dụng:

# netstat -ano | findstr :5000

# Kiểm tra firewall/antivirus blocking port

# Test với curl từ trong chính máy host:

# curl http://localhost:5000/health

# Kiểm tra Docker daemon đang chạy:

# docker version

# =============================================

# 🎯 QUICK REFERENCE

# =============================================

# 🟢 CHƯA SỬA DOCKERFILE: docker run -p 5000:8080

# 🟢 ĐÃ SỬA DOCKERFILE: docker run -p 5000:5000

# Build → Run → Test → Stop → Cleanup