#!/bin/bash

# Update the system

sudo apt-get update

# Uninstall old versions of Docker

sudo apt-get remove docker docker-engine docker.io

# Install dependencies

sudo apt-get install \

apt-transport-https \

ca-certificates \

curl \

gnupg-agent \

software-properties-common

# Add Docker's GPG key

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

# Set up the Docker stable repository

sudo add-apt-repository \

"deb [arch=amd64] https://download.docker.com/linux/ubuntu \

$(lsb\_release -cs) \

stable"

# Update the system again to pick up the new repository

sudo apt-get update

# Install the latest version of Docker CE

sudo apt-get install docker-ce docker-ce-cli containerd.io

# Start Docker and enable it to start at boot time

sudo systemctl start docker

sudo systemctl enable docker

# Verify the installation by running the Docker hello-world image

sudo docker run hello-world

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#!/bin/bash

# Define the Docker image name and tag

image="myimage:latest"

# Build the Docker image

docker build -t "$image" .

# Stop and remove the running container if it exists

if [ "$(docker ps -aq --filter name=mycontainer)" ]; then

docker stop mycontainer

docker rm mycontainer

fi

# Run the Docker container

docker run --name mycontainer -d "$image"

# Check if the container is running

if [ "$(docker ps -q --filter name=mycontainer)" ]; then

echo "The container is running successfully."

else

echo "The container failed to start."

fi

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*