

Docker Compose

Contents [hide]

1 Copyright Notice

2 Author

3 Docker Compose Installation

4 Wordpress Example

4.1 Start Wordpress without Docker Compose

4.2 Start Wordpress with Docker Compose

5 Mongo Counter Example

Copyright Notice

Copyright © 2004-2021 by NobleProg Limited All rights reserved.

This publication is protected by copyright, and permission must be obtained from the publisher prior to any prohibited reproduction, storage in a retrieval system, or transmission in any form or by any means, electronic, mechanical, photocopying, recording, or likewise.

training@nobleprog.com tel: +44 20 7558 8274

Author

■ Kamil Baran

Docker Compose Installation

```
$ sudo -i
$ curl -L https://github.com/docker/compose/releases/download/1.26.2/docker-compose-`uname -s`-`uname -m` > /usr/local/bin/docker-compose
$ chmod +x /usr/local/bin/docker-compose
$ curl -L https://raw.githubusercontent.com/docker/compose/$(docker-compose version --short)/contrib/completion/bash/docker-compose > /etc/bash_completion.d/docker-compose
$ exit
$ docker-compose --version
```

■ more info: docs.docker.com/compose/install

Wordpress Example

Start Wordpress without Docker Compose

To start a Wordpress site in containers, you need to complete following steps:

1. create custom network for the application
2. create volume for database container
3. create volume for web server container
4. create database container
5. create web server container

The commands below represents an example of all required steps:

```
$ docker network create wordpress_net
$ docker volume create wordpress_mysql_data
$ docker volume create wordpress_html
$ docker run -d --name wordpress_mysql --network wordpress_net -v wordpress_mysql_data:/var/lib/mysql -e MYSQL_ROOT_PASSWORD=NobleProg -e MYSQL_DATABASE=wordpress -e MYSQL_USER=wordpress -e MYSQL_PASSWORD=wordpress mysql:5.7
$ docker run -d --name wordpress_apache --network wordpress_net -v wordpress_html:/var/www/html -p 80:80 -e WORDPRESS_DB_HOST=wordpress_mysql:3306 -e WORDPRESS_DB_USER=wordpress -e WORDPRESS_DB_PASSWORD=wordpress wordpress:php7.2-apache
```

When you want to delete your Wordpress site completely you need to remove containers, volumes and network by executing following commands:

```
$ docker container rm -f wordpress_mysql wordpress_apache
$ docker volume rm wordpress_mysql_data wordpress_html
$ docker network rm wordpress_net
```

Start Wordpress with Docker Compose

Docker Compose allows you to manage your multi-container applications by running a straightforward command. For example, to start, stop and remove Wordpress site, you need to execute following commands accordingly:

```
$ docker-compose up -d
$ docker-compose stop
$ docker-compose down --volumes
```

There are only a few things you need to do before you can start Wordpress site by using Docker Compose:

1. Install Docker Compose (usually it's not installed with Docker)
2. create a docker-compose.yml (or .yaml) in an empty directory containing following content
3. run above commands in the same directory where the docker-compose.yml file is located (or use -f option to specify a path to the file).

```
version: '3'
services:
  mysql:
    image: mysql:latest
    volumes:
      - mysql_data:/var/lib/mysql
    environment:
      - MYSQL_ROOT_PASSWORD=NobleProg
      - MYSQL_DATABASE=wordpress
      - MYSQL_USER=wordpress
      - MYSQL_PASSWORD=wordpress
    networks:
      - net
  apache:
    depends_on:
      - mysql
    image: wordpress:latest
    volumes:
      - html:/var/www/html
    ports:
      - 80:80
    environment:
      - WORDPRESS_DB_HOST=mysql:3306
      - WORDPRESS_DB_USER=wordpress
      - WORDPRESS_DB_PASSWORD=wordpress
    networks:
      - net
volumes:
  mysql_data:
  html:
networks:
  net:
```

Mongo Counter Example

- create an empty directory called mc-app
- copy content from httpd (Dockerfile and html directory) and mongodb (Dockerfile) directories created previously
- create a docker-compose.yml file containing below configuration

```
version: '3'
services:
  httpd:
    build: ./httpd
    image: kamilbaran/training:httpd
    network_mode: host
  mongo:
    build: ./mongo
    image: kamilbaran/training:mongo
    network_mode: host
```

- in order to build all images at once and start entire application run below command

```
$ docker-compose up --build -d
```

- more information about options available in docker-compose files: docs.docker.com/compose/compose-file

Categories: Docker | Course code dockcm