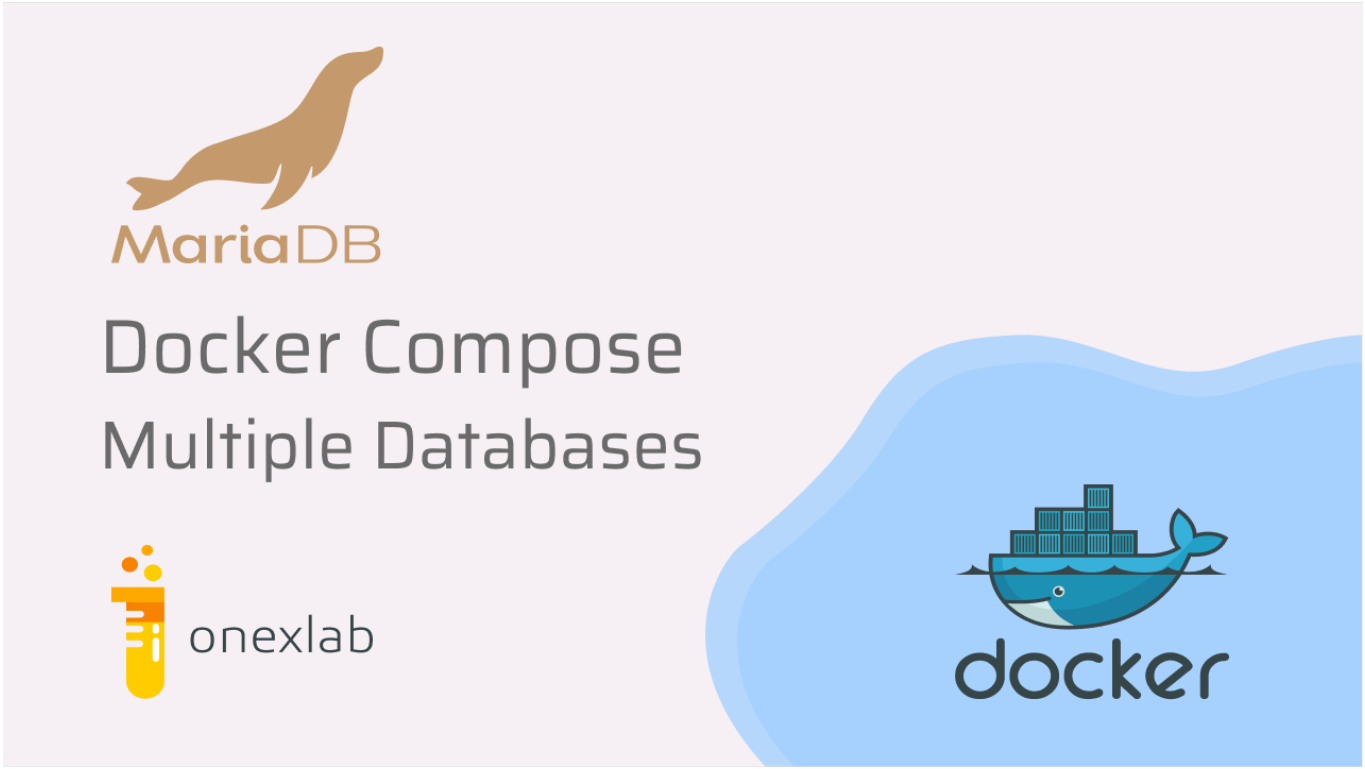
Docker Compose MariaDB

Onexlab Follow
Aug 12 · 3 min read ★

*In this article, We will explain to you how to use **Docker Compose** to create **MariaDB** docker container*



Docker Compose MariaDB Multiple Database

*We will create **MariaDB** following multiple databases using **Docker-Compose***

1. mydb (use can use it for local development)
2. test (use can use it for testing)

> **Production Env:**

The **production database** environment. You can use cloud databases such as **AWS RDS**, **Heroku,** etc.

> **Development Env:**

A dedicated environment for database for local **development**.

> **Test Env:**

A dedicated environment for database **testing**.

Why we need multiple databases?

If you have only a **Production Database** and you are developing a **new feature** if something **breaks** it will affect the **whole Production Database.**

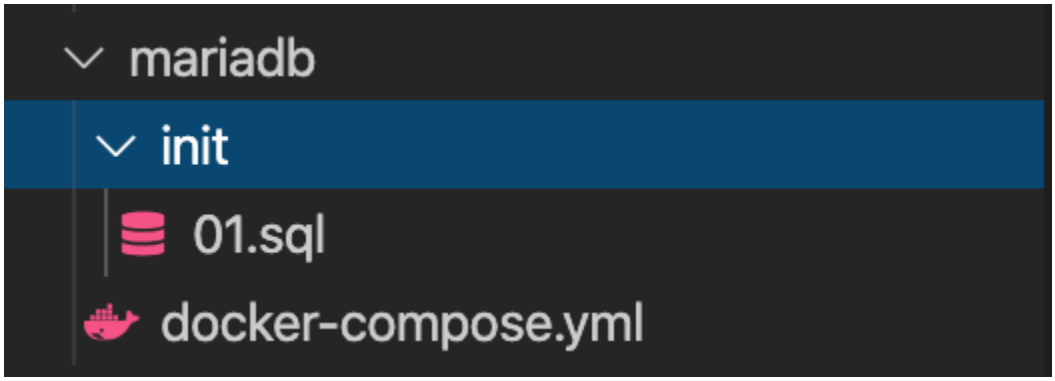
For best practice, You must have a different **Database Development Environment** because if you are developing a new feature in a **Development Environment** if something breaks it will not impact on **Production Database Environment.**

Let's Start Creating Multiple Environment Database

Hope you are familiar with "**Docker-Compose**"

Folder structure

```
Project
├── docker-compose.yml (File)
├── init (Directory)
│   ├── 01.sql (File)
```



Create a new file **docker-compose.yml**

Create a new file **docker-compose.yml**

```yaml
version: '3.1'

services:

  db:
    image: mariadb
    restart: always
    environment:
      MYSQL_ROOT_PASSWORD: example
      MYSQL_DATABASE: mydb
      MYSQL_USER: user
      MYSQL_PASSWORD: user

    volumes:
        - ./init:/docker-entrypoint-initdb.d

  adminer:
    image: adminer
    restart: always
    ports:
      - 8080:8080
```

docker-compose.yml hosted with ❤️ by **GitHub** view raw

Above file, we have created a **MariaDB** Docker container with default **Port No:** 3306

As well as set environment variables such as default **username, password** of **MariaDB** container.

if you check the entire file we have also created volumes that point to the `./init:/docker-entrypoint-initdb.d` file.

```
volumes:
    - ./init:/docker-entrypoint-initdb.d
```

The script inside **“init/01.sql”** will create multiple databases upon container startup. You can see the following file we have created 1 new database `test`

1. **mydb** (Local development)
2. **test** (For Testing)

```
CREATE DATABASE IF NOT EXISTS `test`;
GRANT ALL ON `test`.* TO 'user'@'%';
```

01.sql hosted with ❤ by GitHub view raw

When you run the following command in the root directory for the project.

```
docker-compose up
```

then you will see in the console **“init/01.sql”** script will be executed and create a new database `test` upon docker startup.

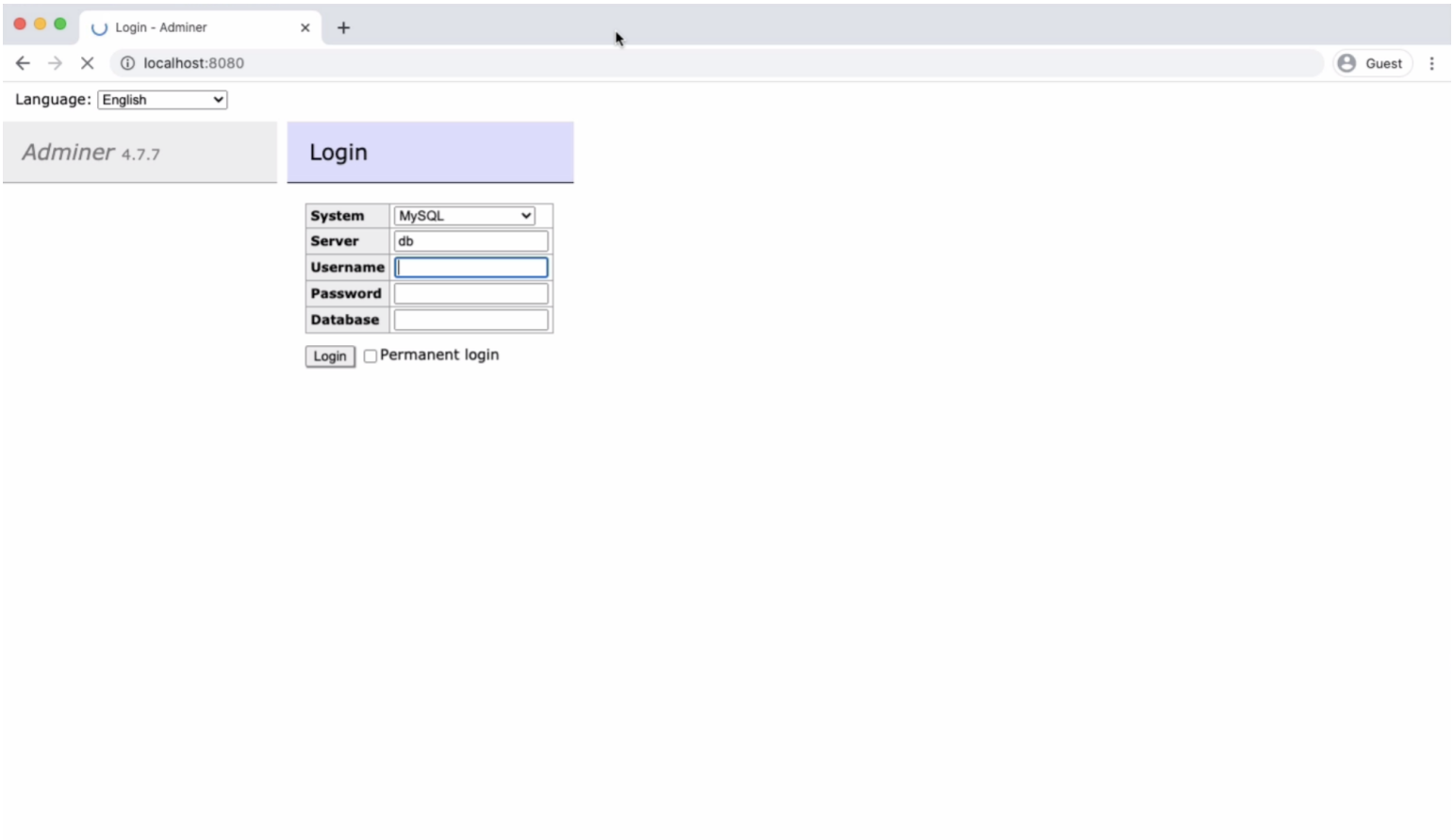
How to test connection

For testing, we used **Adminer** configured already in the **docker-compose** file.

Try to run following URL in the browser

```
http://localhost:8080
```

You will see the following screen



Then enter **username**

```
user
```

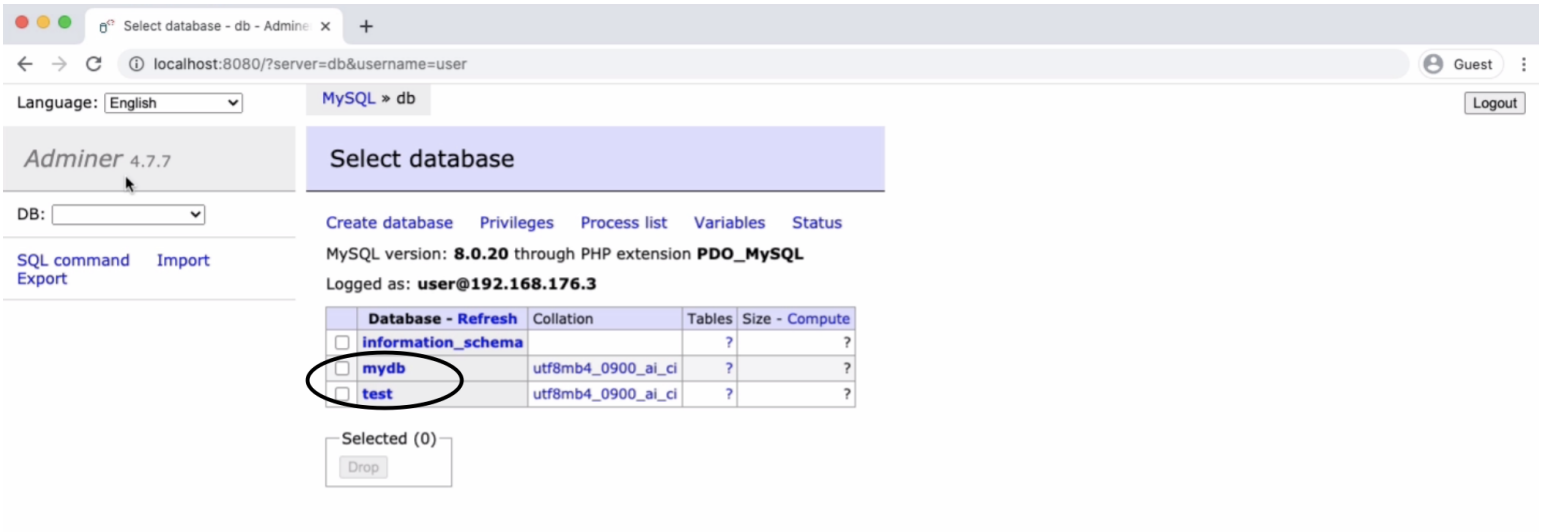
and **password**

```
user
```

then hit the **Login** button then you will see there are 2 database

1. mydb
2. test

as shown in the following screenshot.



GitHub

oxlb/docker-compose-mariadb-multiple-database

Docker Compose MariaDb Multiple Database. Contribute to oxlb/docker-compose-mariadb-multiple-database development by...

github.com

Docker | Docker Compose | Mariadb | Onexlab | Containers