

Lab 5: Create first docker compose file with nginx and mysql

In this lab we are going to bringup a nginx and mysql containers using docker-compose.

Pre-requisite

- Create an account with [DockerHub](#)
- Open [PWD](#) Platform on your browser
- Click on **Add New Instance** on the left side of the screen to bring up Alpine OS instance on the right side

Setup environment:

```
$ mkdir Myapp  
$ cd Myapp
```

Now lets create passowrd file for our DB:

```
$ openssl rand -base64 32 > db_password.txt  
$ openssl rand -base64 32 > db_root_password.txt
```

Create a docker-compose.yml file:

```
version: '3.1'  
services:  
  #Nginx Service  
  webserver:  
    image: nginx:alpine  
    container_name: webserver  
    restart: unless-stopped  
    ports:  
      - "80:80"  
      - "443:443"  
  #Mysql DB  
  db:  
    image: mysql:5.7  
    container_name: Mysqldb  
    restart: unless-stopped  
    volumes:  
      - db_data:/var/lib/mysql  
    ports:  
      - "3306:3306"  
    environment:  
      MYSQL_ROOT_PASSWORD_FILE: /run/secrets/db_root_password  
      MYSQL_DATABASE: wordpress  
      MYSQL_USER: wordpress  
      MYSQL_PASSWORD_FILE: /run/secrets/db_password  
    secrets:  
      - db_root_password  
      - db_password  
secrets:  
  db_password:
```

```
file: db_password.txt
db_root_password:
file: db_root_password.txt

volumes:
db_data:
```

Create the compose container:

```
$ sudo docker-compose up
```

SSH into the instance and check the app

List out the compose services:

```
$ docker-compose ps
```

Name	Command	State	Ports
Mysqldb	docker-entrypoint.sh mysqld	Up	0.0.0.0:3306->3306/tcp, 33060/tcp
webserver	nginx -g daemon off;	Up	0.0.0.0:443->443/tcp, 0.0.0.0:80->80/tcp

Verify the nginx is running:

```
$ curl http://localhost
```

Verify the Mysql db:

```
$ docker exec -it Mysqldb mysql -u root -p

Enter the root password which is in db_root_password.txt
```