



**APPLICATION CONTAINERIZATION AND ORCHESTRATION
LAB**

STUDENT

SACHIN AGGARWAL
500097500
B4

SUBMITTED TO

DR.HITESH KUMAR SHARMA

AIM: Working with Docker Compose File to Control Multiple Containers

Steps to Complete:

Creating compose files

- ❖ Create a directory named `nginx` in your root.

```
mkdir nginx
```

- ❖ Switch to that directory and create a file named `docker-compose.yaml`

```
cd nginx
```

vi docker-compose.yml

```
Sachins-Air:/ sachinaggarwal$ pwd
/
Sachins-Air:/ sachinaggarwal$ mkdir nginx
mkdir: nginx: Read-only file system
Sachins-Air:/ sachinaggarwal$ mkdir ~/nginx
mkdir: /Users/sachinaggarwal/nginx: File exists
Sachins-Air:/ sachinaggarwal$ cd ~/nginx
Sachins-Air:nginx sachinaggarwal$ vi docker-compose.yml
```

```
version: '2'
services:
  databases:
    image: mysql
    ports:
      - "3307:3306"
    env_file:
      - envs.env
  web:
    image: nginx
    ports:
      - "80:80"
    depends_on:
      - databases
```

- ❖ Use docker-compose version 2 to create docker-compose.yaml file.

Create a service named "databases". Use image named "mysql"

Map container 3306 port to host machine 3306 port.

Add environment variables named

"MYSQL_ROOT_PASSWORD", "MYSQL_DATABASE",
"MYSQL_USER" and "MYSQL_PASSWORD" along with
corresponding values for all.

```
cat evs.env
```

```
MYSQL_ROOT_PASSWORD=redhat08
```

```
MYSQL_DATABASE=nginxdb
```

```
MYSQL_USER=root
```

```
Sachins-Air:nginx sachinaggarwal$ cat > evs.env <<EOF
> MYSQL_ROOT_PASSWORD=redhat08
> MYSQL_DATABASE=nginxdb
> MYSQL_USER=root
> MYSQL_PASSWORD=yourpassword
> EOF
```

Add another service named "web"

Use image "nginx"

```
cat docker-compose.yml
```

```
Sachins-Air:nginx sachinaggarwal$ cat docker-compose.yml
version: '2'
services:
  databases:
    image: mysql
    ports:
      - "3307:3306"
    env_file:
      - evs.env
  web:
    image: nginx
    ports:
      - "80:80"
    depends_on:
      - databases
```

Running images using docker-compose

- ❖ Save docker-compose.yaml file and do docker-compose up.

docker-compose up -d

```
Sachins-Air:nginx sachinagarwal$ docker-compose up -d
[+] Running 11/11
✓ databases 10 layers [#####] 0B/0B Pulled 56.3s
✓ e39ec8f010eb Pull complete 20.3s
✓ e2b7fadc33ec Pull complete 20.4s
✓ 9d193449aafd Pull complete 20.4s
✓ 6ea497c74b15 Pull complete 20.6s
✓ 7778acbf55f3 Pull complete 20.6s
✓ b65a5d7a2435 Pull complete 20.6s
✓ cef9fb0078a5 Pull complete 36.9s
✓ 5b6dc73ec724 Pull complete 36.9s
✓ d992bb39e209 Pull complete 50.7s
✓ 4431432b89a3 Pull complete 50.7s
[+] Running 3/3
✓ Network nginx_default Created 0.1s
✓ Container nginx-databases-1 Started 1.2s
✓ Container nginx-web-1 Started 0.9s
```

Verify nginx service is up and is accessible on the machine.

curl localhost:80

```
Sachins-Air:nginx sachinagarwal$ curl http://localhost:80
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
```