

## EXPERIMENT 6

### 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
```

```
PS C:\Users\manya\OneDrive\Desktop\ACO> mkdir nginx

Directory: C:\Users\manya\OneDrive\Desktop\ACO

Mode                LastWriteTime         Length Name
----                -
d-----          02-12-2023    22:44             nginx

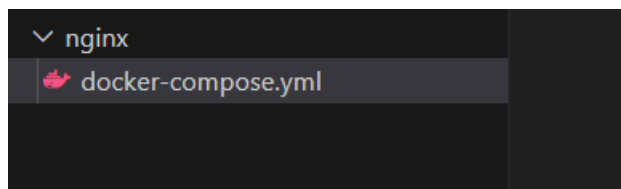
PS C:\Users\manya\OneDrive\Desktop\ACO> |
```

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

```
cd nginx
```

```
PS C:\Users\manya\OneDrive\Desktop\ACO> cd nginx
PS C:\Users\manya\OneDrive\Desktop\ACO\nginx> |
```

```
vi docker-compose.yml
```



- ❖ 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
```

Add another service named "web"  
Use image "nginx"

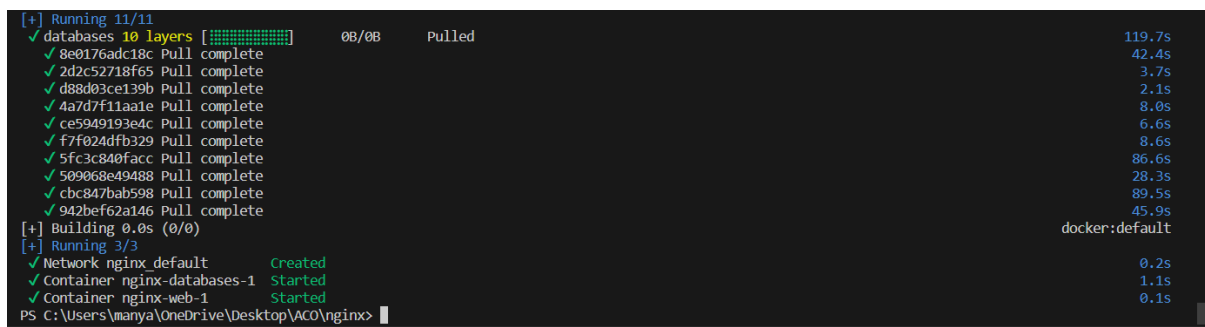
```
cat docker-compose.yml
```

```
version: '3'
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
```



```
[+] Running 11/11
✓ databases 10 layers [#####] 0B/0B Pulled 119.7s
✓ 8e0176adc18c Pull complete 42.4s
✓ 2d2c52718f65 Pull complete 3.7s
✓ d88d03ce139b Pull complete 2.1s
✓ 4a7d7f11aa1e Pull complete 8.0s
✓ ce5949193e4c Pull complete 6.6s
✓ f7f024dfb329 Pull complete 8.6s
✓ 5fc3c840facc Pull complete 86.6s
✓ 509068e49488 Pull complete 28.3s
✓ cbc847bab598 Pull complete 89.5s
✓ 942bef62a146 Pull complete 45.9s
[+] Building 0.0s (0/0) docker:default
[+] Running 3/3
✓ Network nginx_default Created 0.2s
✓ Container nginx-databases-1 Started 1.1s
✓ Container nginx-web-1 Started 0.1s
PS C:\Users\manya\OneDrive\Desktop\ACO\nginx>
```

- ❖ Verify nginx service is up and is accessible on machine.

```
curl localhost:80
```

```
PS C:\Users\manya\OneDrive\Desktop\ACO\nginx> curl 127.0.0.1:80
```

```
StatusCode      : 200
StatusDescription : OK
Content         : <!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>...
RawContent      : HTTP/1.1 200 OK
                  Connection: keep-alive
                  Accept-Ranges: bytes
                  Content-Length: 615
                  Content-Type: text/html
                  Date: Sat, 02 Dec 2023 17:52:10 GMT
```

localhost

## Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to [nginx.org](https://nginx.org).  
Commercial support is available at [nginx.com](https://nginx.com).

*Thank you for using nginx.*

Stop and remove your docker container using docker-compose.

```
docker-compose down
```

```
PS C:\Users\manya\OneDrive\Desktop\ACO\nginx> docker-compose down
[+] Running 3/3
 ✓ Container nginx-web-1      Removed
 ✓ Container nginx-databases-1 Removed
 ✓ Network nginx_default     Removed
PS C:\Users\manya\OneDrive\Desktop\ACO\nginx> █
```