

Docker is a powerful tool for running containers, but many applications consist of a collection of multiple containers running different software and interacting with one another. These complex applications can become difficult to maintain when all of their components are managed separately. Docker Compose offers a solution to this problem by allowing you to declaratively describe a set of multiple resources and manage them as a unit. In this lesson, we will discuss Docker Compose and demonstrate creating and managing a simple, multi-container application.

## Relevant Documentation

- <https://docs.docker.com/compose/>

## Lesson Reference

1. Install Docker Compose.

```
sudo curl -L "https://github.com/docker/compose/releases/download/1.24.0/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
sudo chmod +x /usr/local/bin/docker-compose
docker-compose version
```

2. Set up a Docker Compose project.

```
mkdir nginx-compose
cd nginx-compose
vi docker-compose.yml
```

3. Create your `docker-compose.yml`.

```
version: '3'
services:
  web:
    image: nginx
    ports:
      - "8080:80"
  redis:
    image: redis:alpine
```

4. Start your Compose app.

```
docker-compose up -d
```

5. List the Docker Compose containers, then stop the app.

```
docker-compose ps
docker-compose down
```