n this tutorial, we will create a simple Docker Compose configuration which creates an NGINX container and scales the application by making multiple replicas of it.

First, ensure Docker (and Docker Compose if using Linux) is installed.

**Docker Compose Configuration File**

Docker Compose is configured using YAML files, here we will configure one to create an NGINX container.  
The configuration shown below will create an NGINX container and publish port 80 to a random high port such as 35000.

Create a folder for this tutorial called docker-compose-nginx-tutorial and change to that directory:

mkdir docker-compose-nginx-tutorial && cd $\_

Now create a file called docker-compose.yaml and enter the following:

version: "3.8"

services:

nginx:

image: nginx:alpine

ports:

- target: 80

protocol: tcp

**Run Your Configuration**

You should now be able to run your first configuration using a docker-compose command:

docker-compose up -d

**View the Running Containers Using Compose**

The containers that are running can now be viewed:

docker-compose ps

The output should be something like this:

Name Command State Ports

-------------------------------------------------------------------

test\_nginx\_1 nginx -g daemon off; Up 0.0.0.0:32768->80/tcp

**Access the Application**

We can see that under the **Ports** column the high port that the container has been published on is listed.  
Try connect to the container in a browser or by using a curl command using that high port.

For this example we could use:

curl localhost:32768

Creating a response like this:

<!DOCTYPE html>

<html>

<head>

<title>Welcome to nginx!</title>

<style>

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>

**Scale Your Application**

Compose will allow you to scale your application to an amount that you specify.  
You will be able to scale to different amounts depending on the machine that you are using and the resources that are being used by each of the containers.

In this instance we should be fine scaling to 3 NGINX containers:

docker-compose up -d --scale nginx=3

Now when you view the running containers there should be 3 instances of NGINX running:

docker-compose ps

Name Command State Ports

-------------------------------------------------------------------

test\_nginx\_1 nginx -g daemon off; Up 0.0.0.0:32768->80/tcp

test\_nginx\_2 nginx -g daemon off; Up 0.0.0.0:32769->80/tcp

test\_nginx\_3 nginx -g daemon off; Up 0.0.0.0:32770->80/tc

**Clean Up**

We can now stop and remove all the containers and images used by running this command:

docker-compose down --rmi all