### Pre-requisites

* 3 x Ubuntu VM's running version 18.04 LTS (Spun up with your choice of cloud provider).
* Docker installed on all VM's
* Port 80 open on your Manager VM

### Create Cluster

Put together a cluster with 3 nodes, one manager and 2 workers.

1. On the machine you have chosen to be the Swarm Manager run the following command:

docker swarm init

This will initialise the Swarm Manager and Docker will output a Join Token. Note this down as you will need this command to add workers to your Swarm.

1. Log on to the Virtual Machine's that will be worker node's in the Swarm, run the join command that was generated earlier, the command should be similiar to the command below:

docker swarm join --token SWMTKN-1-14601cadnn15hf45gjqql23jta4vk4kxh8uy6zyn8vo0fxzu5z-31kwu4qd6up8ktez4kj62ppvr 192.168.61.1:2357

If you can’t obtain the join command anymore, run the join-token command below on the Swarm Manager VM:

docker swarm join-token worker

1. Log on to your second VM that will also be a worker node, run the same join-token command to join the Swarm as a worker.

Now you have 3 nodes in the Swarm! To check if the Swarm has been configured properly run the following command:

docker node ls

### Create a Service

Create a service that uses the bobcrutchley/python-http-server:latest image and is named python-http-server. Run the following command on your manager node to do this:

docker service create --name python-http-server bobcrutchley/python-http-server:latest

### Update the Service

Update the service so that there are 10 replicas and the port 9000 (inside the container) has been published to 80 (outside the container). Run the following Command to update the service:

docker service update --replicas 10 --publish-add 80:9000 python-http-server

The port numbers in the command above 80:9000 represent the Published Port which is 80 and the Container Port 9000.

### Access the Service

Use the curl CLI tool to view the info.json file served by the application, this file shows the name of the host that it is running on.

Run the following command:

curl http://[YOUR\_PRIVATE\_IP]/info.json

You can substitute the private IP address here with the private IP address of your manager node. Curl the file multiple times, what do you notice about the output?

### Remove the Worker Nodes

Update the amount of replicas to 2. Run the command Below on the manager VM to do this:

docker service update --replicas 2 python-http-server

Drain both of the worker nodes and remove them. Run the command below and replace the [NODE\_NAME] with your own.

docker node update --availability drain [NODE\_NAME]

You can get the node names by running docker node ls

## Clean up

1. Remove Service (Run on Swarm Manager):

docker service rm python-http-server

1. On each worker node run the command below to shut it down:

docker swarm leave

1. Now remove the nodes using the command below on the Swarm Manager:

docker node rm [YOUR NODE NAME]

1. Shutdown and remove all VM resources on your cloud provider platform to avoid unexpected charges and fees.