For the swarm to work, each virtual machine needs to have Docker installed.

To check if there is Docker on the VM, run the below command.

```
$docker version
```

An IP address needs to be addressed to advertise the swarm service.

Use the public IP address from the cloud provider, and to figure that out, run the below command on the same VM.

```
$docker swarm init
```

The output from the above command will have IP addresses within the parenthesis. Use the appropriate public IP address to host the swarm services.

Note: Swarm communicates through port 2377.

Start the Swarm: Run the below command on the same virtual machine.

```
$docker swarm init --advertise-addr <the_public_Ip_address>
```

This now makes the current virtual machine, that had the above command ran on, the leader-manager node in the swarm.

Side note: There can only be one leader at a time.

From the output of the above command, copy and past the "docker swarm join...." string all the way till the end of the port number.

Note: (What you are copying is the swarm join command, so if wanting to add another virtual machine to the swarm, that is the command to do it.)

In another virtual machine paste that string you just copied and run it.

After running the swarm join command on the virtual machine it will now be joined into the swarm. But it joins only as a worker node, not a manager node.

To join another virtual machine in the swarm run the same swarm join command that was copied earlier and run it on that virtual machine.

Note: Virtual machines that join the swarm with the swarm join command cannot run swarm commands in their virtual machine and they do not have access to swarm commands because they are worker nodes.

If you want to change a virtual machine to be a manager node instead of a worker node, run the below commands in the leader node's virtual machine. (The leader node has access to swarm commands.)

To find the node's host names run the below command.

$ **`docker node ls`** Choose a VM host to be a manager then run the next command.

$ **`docker node update --role manager <VM_host_name>`**

A second way to do the above commands is to run the below command within the leader's virtual machine.

$ **`docker swarm join-token manager`**

Copy the string output from the above command starting at, "docker swarm-join....." all the way to the ending of the port number.

Now, pick a virtual machine that you would like to be a manager in the swarm, paste what you copied from the above command and run it on your virtual machine of pick.

On the leader virtual machine run the below command to keep track of all instances/nodes.

$ **`docker node ls`**

To list the containers running on a particular virtual machine that is in the swarm. Run the command below.

```
$ docker node ps <VM_host_name>
```

If the leader node, deployed a stack of services, the below command will output the service names that were created or deployed within the swarm.

```
$ docker service ls
```

From the output of the above command, pick a service and run the below command with that service name.

```
$ docker service ps <the_Service_name>
```

The above command will list all the container activities in the swarm.

If the leader node deployed a stack (a YAML file of instructions) run the below command to see the stack services.

```
$ docker stack ps <the_Service_name>
```

Note: The above command will list what instances are running what services that the application is using.