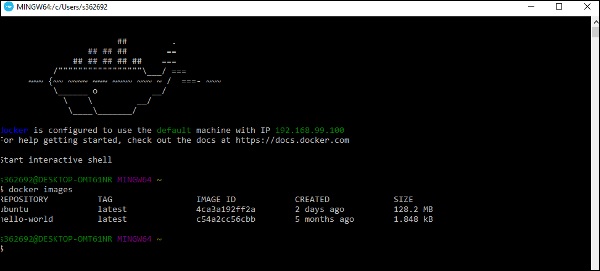
**Docker Images**

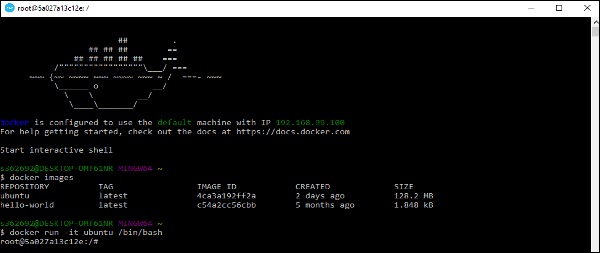
Let’s run the Docker **images** command on the Windows host.



From here, we can see that we have two images − **ubuntu** and **hello-world**.

Running a Container

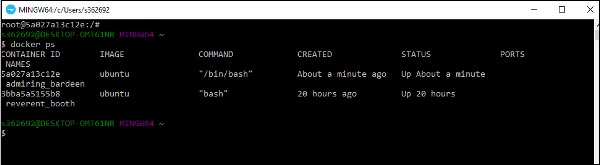
Now let’s run a container in the Windows Docker host.



We can see that by running the container, we can now run the Ubuntu container on a Windows host.

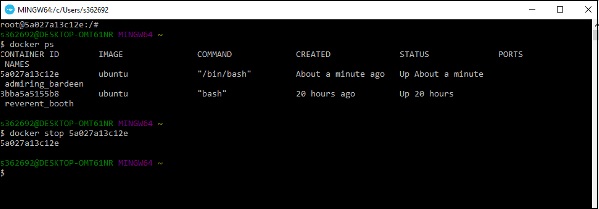
**Listing All Containers**

Let’s list all the containers on the Windows host.



**Stopping a Container**

Let’s now stop a running container on the Windows host.



So you can see that the Docker engine is pretty consistent when it comes to different Docker hosts and it works on Windows in the same way it works on Linux.

We created our Docker File in the last chapter. It’s now time to build the Docker File. The Docker File can be built with the following command −

docker build

Let’s learn more about this command.

## docker build

This method allows the users to build their own Docker images.

### **Syntax**

docker build -t ImageName:TagName dir

### **Options**

* **-t** − is to mention a tag to the image
* **ImageName** − This is the name you want to give to your image.
* **TagName** − This is the tag you want to give to your image.
* **Dir** − The directory where the Docker File is present.

### **Return Value**

None

### **Example**

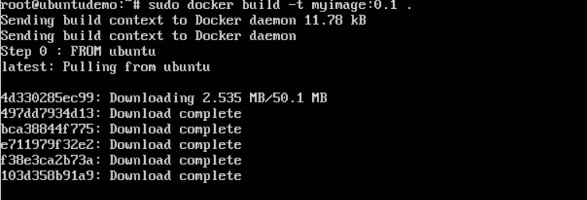
sudo docker build –t myimage:0.1.

Here, **myimage** is the name we are giving to the Image and **0.1** is the tag number we are giving to our image.

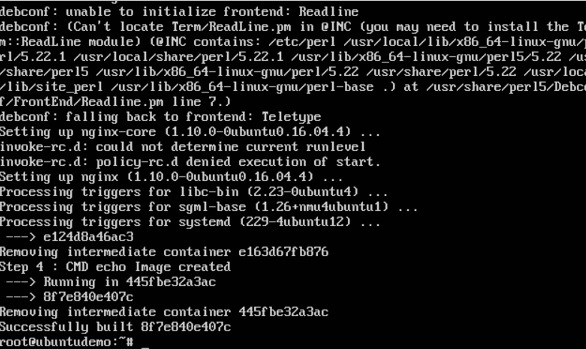
Since the Docker File is in the present working directory, we used "." at the end of the command to signify the present working directory.

### **Output**

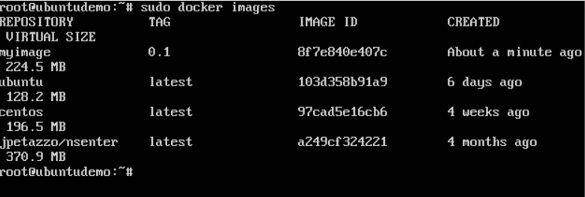
From the output, you will first see that the Ubuntu Image will be downloaded from Docker Hub, because there is no image available locally on the machine.



Finally, when the build is complete, all the necessary commands would have run on the image.



You will then see the successfully built message and the ID of the new Image. When you run the Docker **images command**, you would then be able to see your new image.



You can now build containers from your new Image.