Objective:

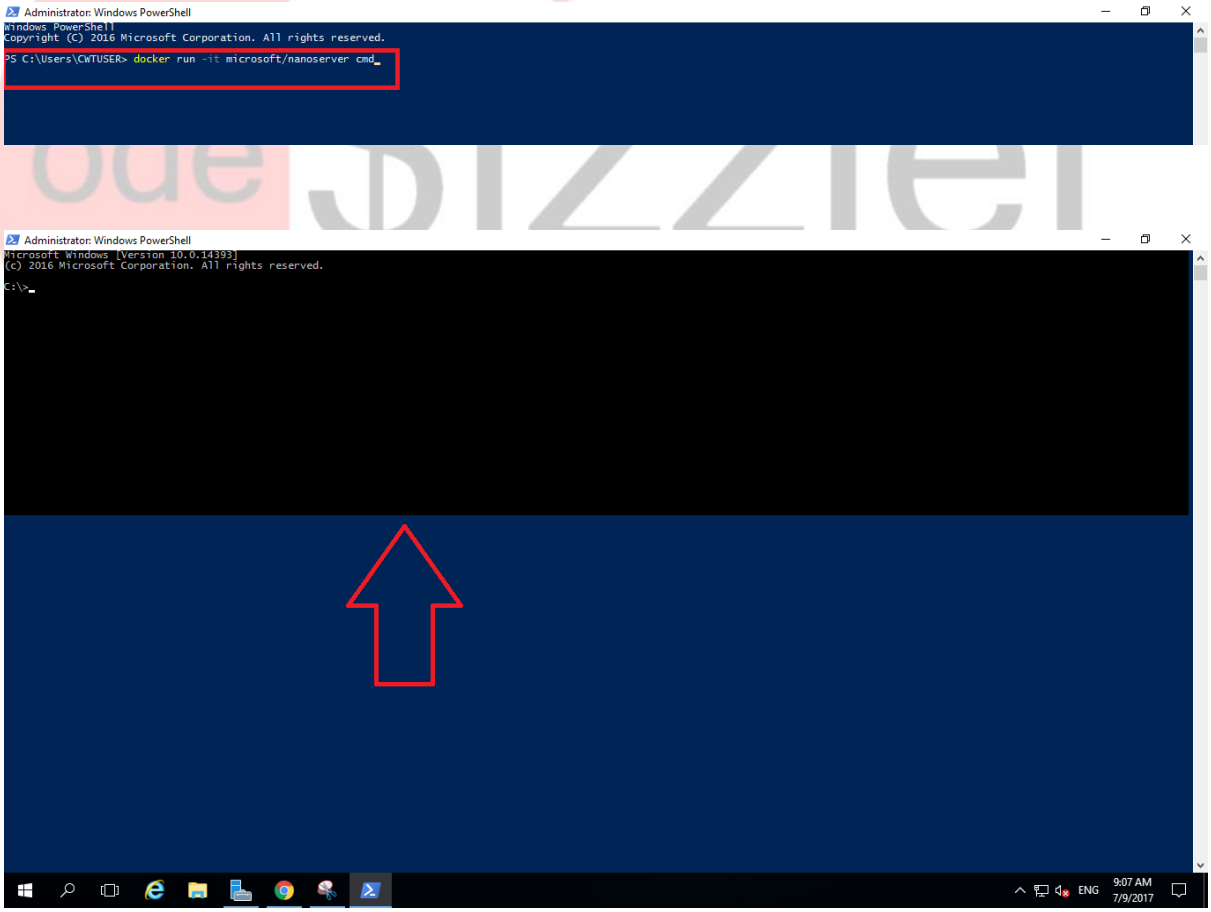
In this module, we will be seeing the different commanding techniques that can be used to work with the containers. We will check on how to start and stop a container, and to run them interactively or even in the background.

Step 1: Running an Image in the Interactive Mode

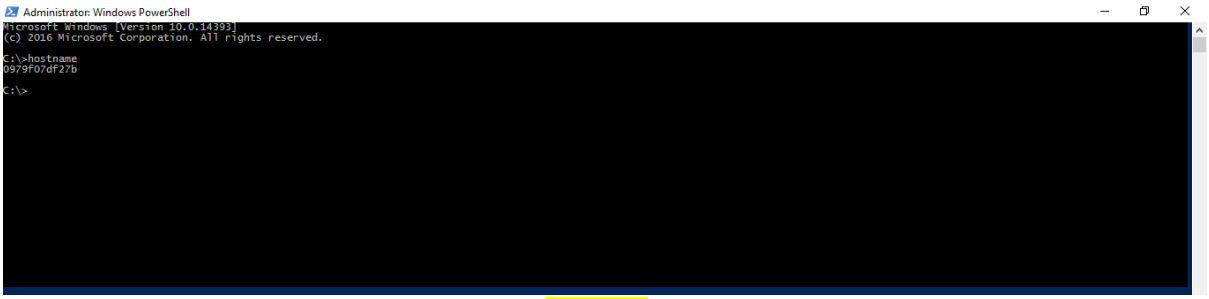
- We shall run an Image which we have downloaded in the previous lab. We shall run it in the interactive manner which means, it will not be able to work in the background. Let us open the **PowerShell** and run the following command.

docker run -it microsoft/nanoserver cmd

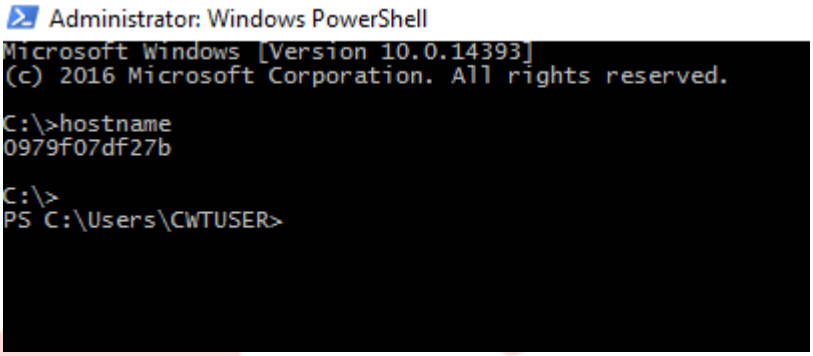
- The above command will by default run the latest version of the Nano server image that has been downloaded.



- Now we shall get the ID of the image which we are running. For that run the command, **hostname**

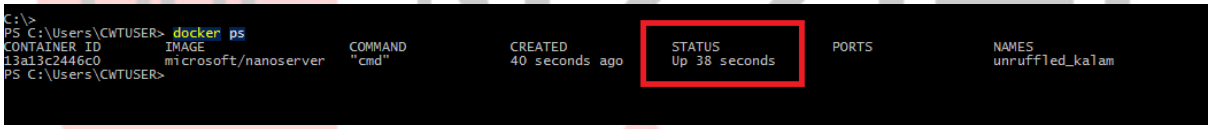


- To disconnect from the container, press **Ctrl+PQ.** This will disconnect you from the container.



- Now we shall get the list of containers that are running. When we do that, we will be able to see our Nano Server image is still running. The command is

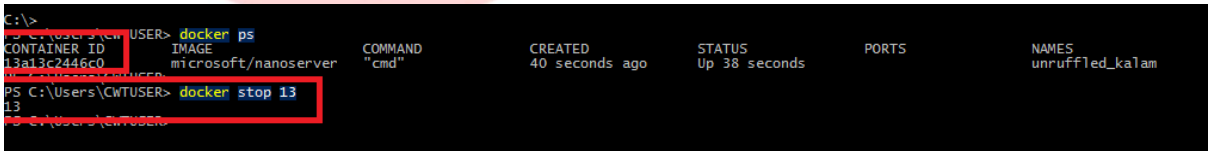
docker ps



- We can stop the container by running the command,

docker stop 13

Note: Replace the number 13 with the first two values of your image host ID



- Again, run the command **docker ps .** You will find that there will be no container running. The reason is because you have run the container in an interactive manner.

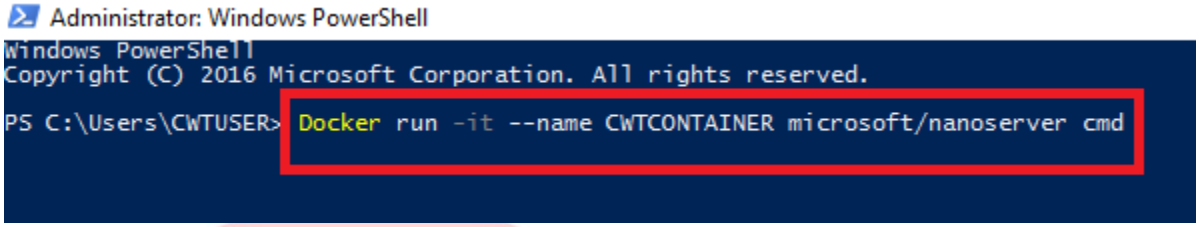
- To check the stopped containers, run the command **docker ps -a .** This will show you all the containers that were stopped.
- Again, to start these containers, run the command **docker start 13**

Note: Replace the number 13 with the first two values of your image host ID

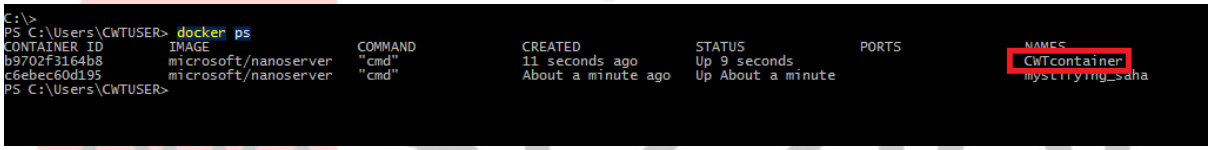
Step 2: Naming your Containers

- By default, the containers will be given some random name. We can try giving some name for our container. To do that, we have to name our containers while we run them. The command will be,

docker run -it --name CWTcontainer microsoft/nanoserver cmd



- After that again press **Ctrl+PQ** run the command **docker ps .** You will be shown with the name which you gave to your container.



- Now we can use the name of that container to take control over it. To stop the container, again run the container command to stop by using the container name. Here replace with your container name in the yellow mark.

docker stop CWTcontainer

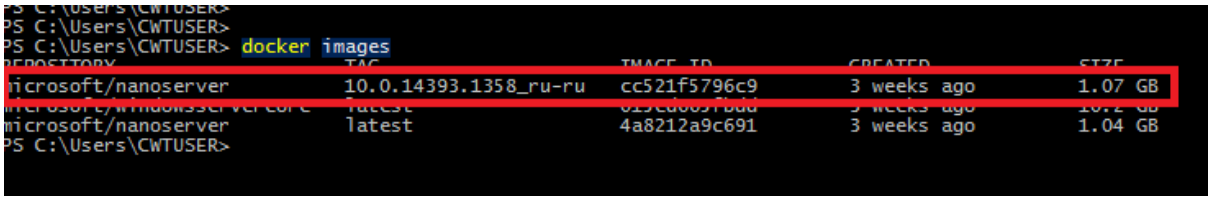
- To remove the container, run the command
  docker rm CWTcontainer
- You can now check the availability of your container by running the command, **docker ps .** It will not be available since we have completely removed it.

Step 3: Removing a running container

- It is not possible to remove a running container using the command,
  docker rm CWTcontainer
- But you can do that using the command docker rm -f CWTcontainer . This will forcefully remove the container.

Step 4: Running a custom container Image in a detached mode

- To get all the available images by running the command **docker images .** You will get all the images that you have downloaded. We shall choose the older version rather than the new version of Nano server.



- Now we shall run a command to start the container of some older version image. This is going to be run in a detached mode which means the container will be able to run even in the background.

docker run -d microsoft/nanoserver:10.0.14393.1358_ru-ru ping -t 4.2.2.3 cmd

- No check your container status using the command **docker ps .** It will running.

Step 5: Interfacing with container after starting them

- We will be able to get inside a container even after running it using **exec** command. This will allow us to start PowerShell or CMD to write some other new functions to our containers. The command is,
  docker exec -it 8b cmd

  You need to replace with the first two letters of your container ID or complete name.

- Now use the command **exit** to get from the container. Then again run the command **docker ps** and still your container will be running since it is started in a detached mode.

Step 6: Getting information about the container

- We can get the information about the container which we are running using the following command.

docker inspect cc

You need to replace with the first two letters of your container ID or complete name.

```
PS C:\Users\CWTUSER> docker inspect cc
[
    {
        "Id": "sha256:cc521f5796c9a94fde311d7fb3bda373a6090414d0ca8e917f8d026aca54c941",
        "RepoTags": [
            "microsoft/nanoserver:10.0.14393.1358_ru-ru"
        ],
        "RepoDigests": [
            "microsoft/nanoserver@sha256:9d6413a658e010e8818cca13aa3d22cb0546f64b38410e5b60c33d835cf7e527"
        ],
        "Parent": "",
        "Comment": "",
        "Created": "2017-06-13T10:38:26.6806818-07:00",
        "Container": "",
        "ContainerConfig": {
            "Hostname": "",
            "Domainname": "",
            "User": "",
            "AttachStdin": false,
            "AttachStdout": false,
            "AttachStderr": false,
            "Tty": false,
            "OpenStdin": false,
            "StdinOnce": false,
            "Env": null,
            "Cmd": null,
            "Image": "",
            "Volumes": null,
            "WorkingDir": "",
            "Entrypoint": null,
            "OnBuild": null,
            "Labels": null
        },
        "DockerVersion": "",
        "Author": "",
        "Config": {
            "Hostname": "",
            "Domainname": "",
            "User": "",
            "AttachStdin": false,
            "AttachStdout": false,
            "AttachStderr": false,
            "Tty": false,
            "OpenStdin": false,
            "StdinOnce": false,
            "Env": null,
            "Cmd": [
```

This is how we work with our containers in different manners.