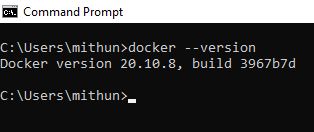
**Docker Selenium Grid Configuration**

**Prerequisite:**

* selenium/hub
* selenium/node-chrome-debug

To check Docker installed in our machine enter the following command in the Command Prompt it will show the docker version installed in your PC.

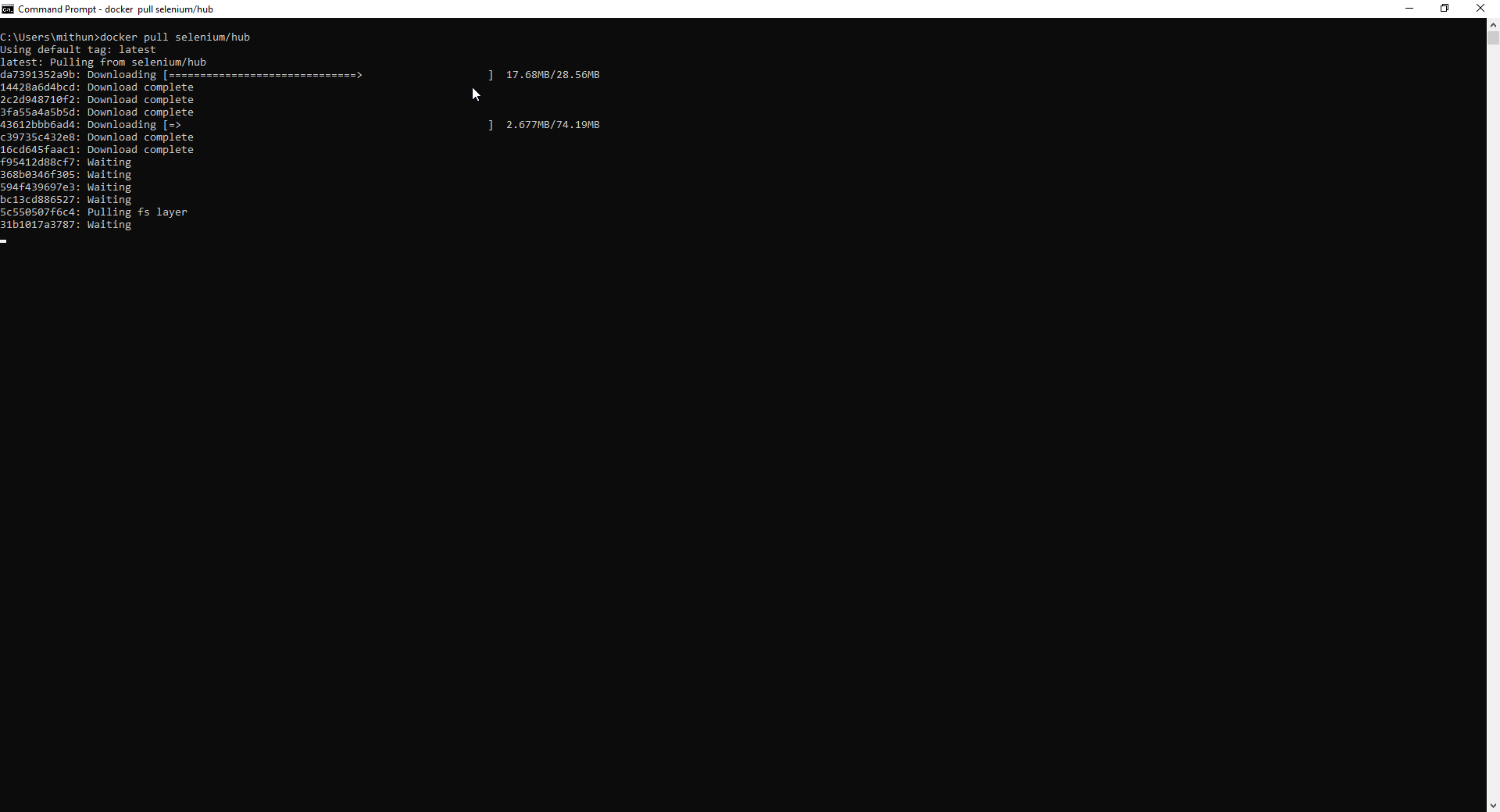
**docker –version**



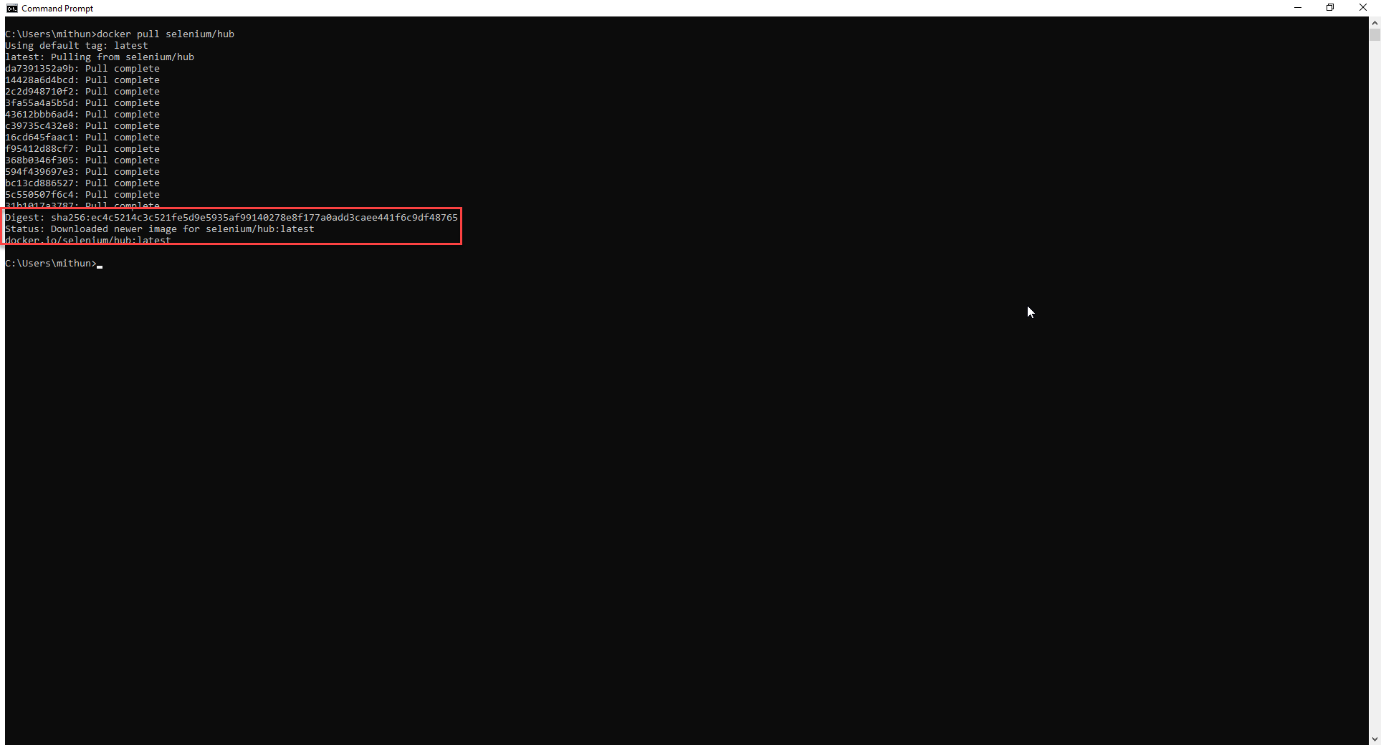
**Selenium Grid Hub Configuration:**

To pull the selenium hub from the docker use the following command in the Command Prompt it will pull the selenium hub image inside the docker.

**docker pull selenium/hub**

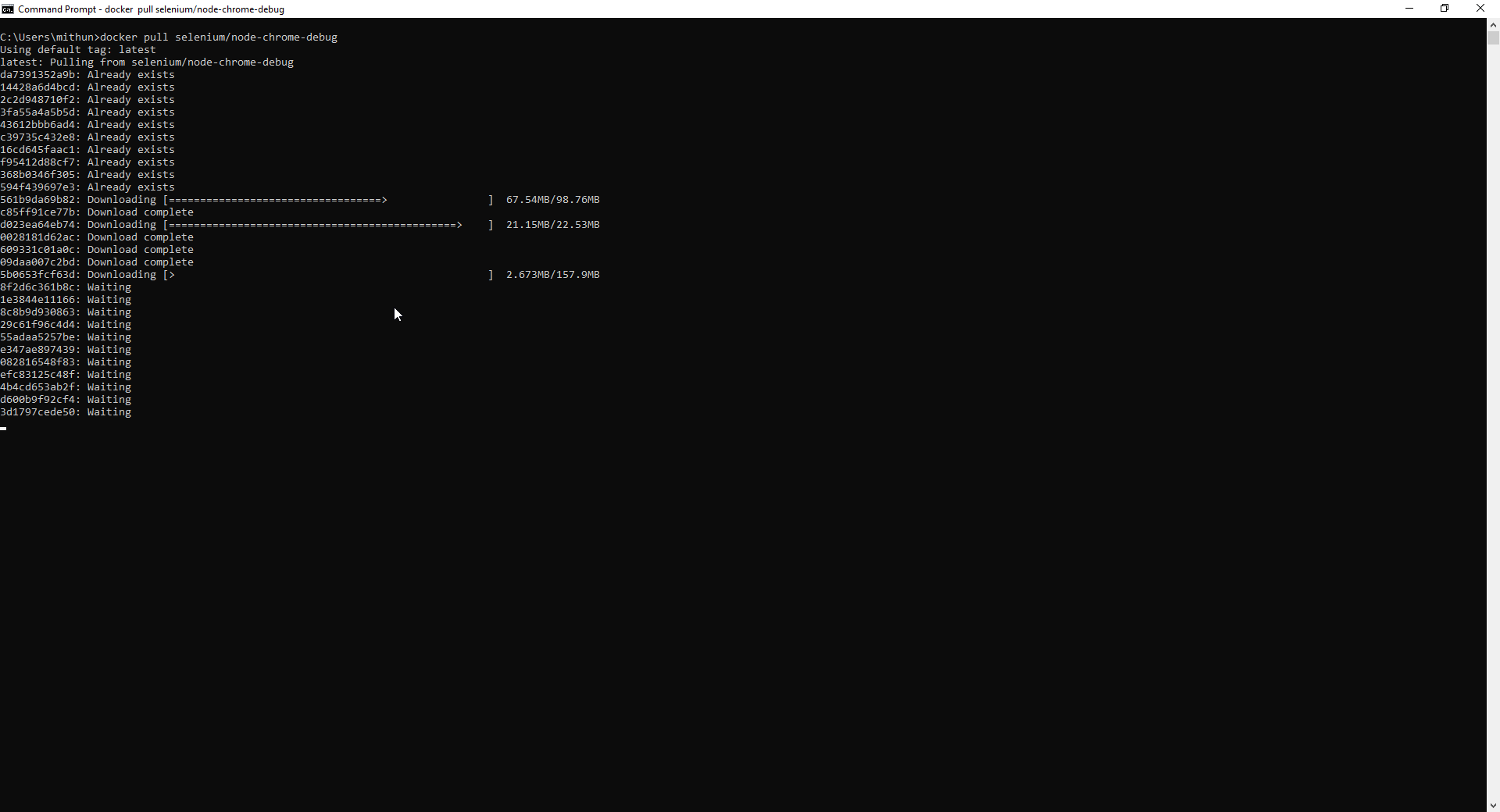


After pulling the selenium hub properly the following lines will be displayed.

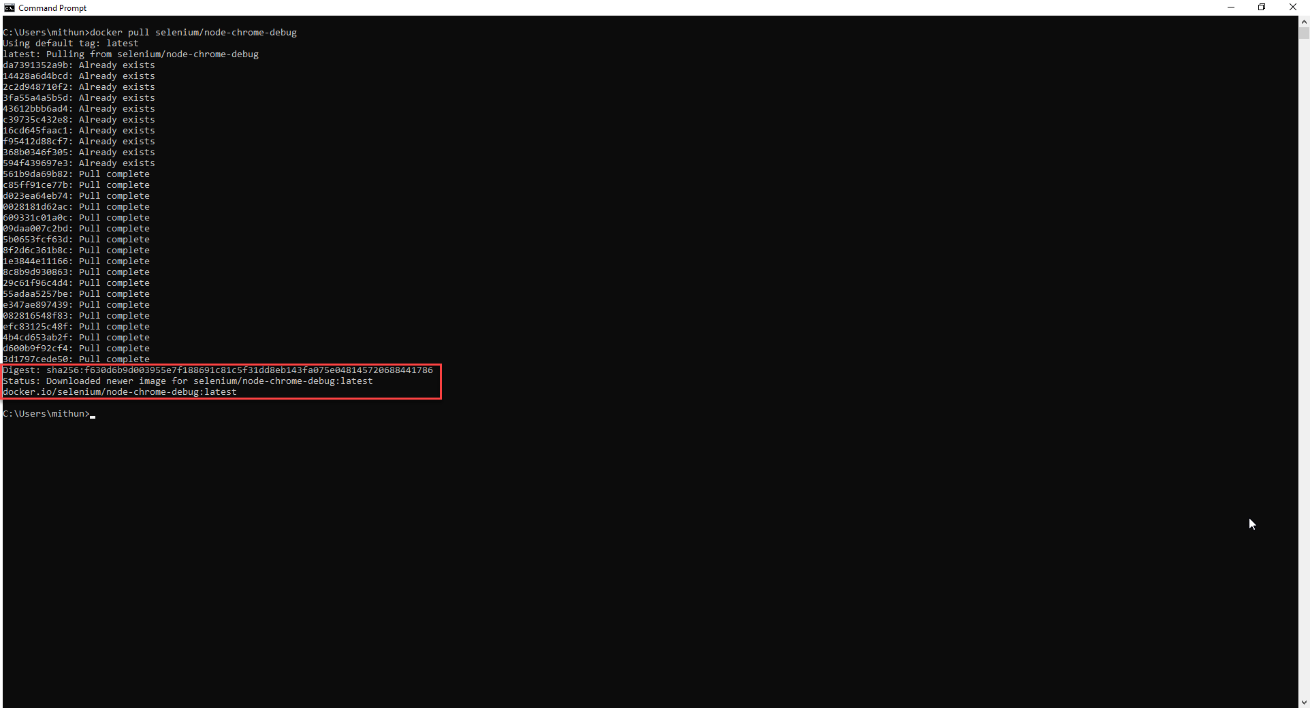


To pull the selenium chrome node from the docker use the following command in the Command Prompt it will pull the selenium chrome node image inside the docker.

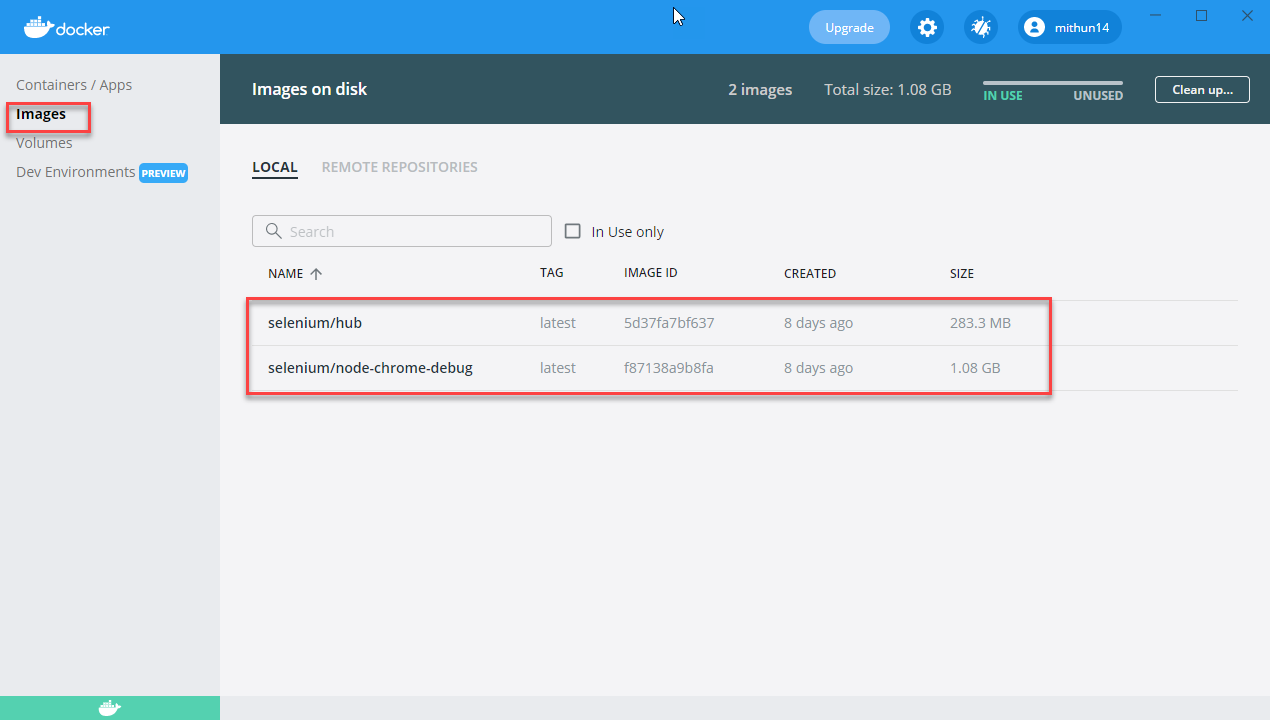
**docker pull selenium/node-chrome-debug**



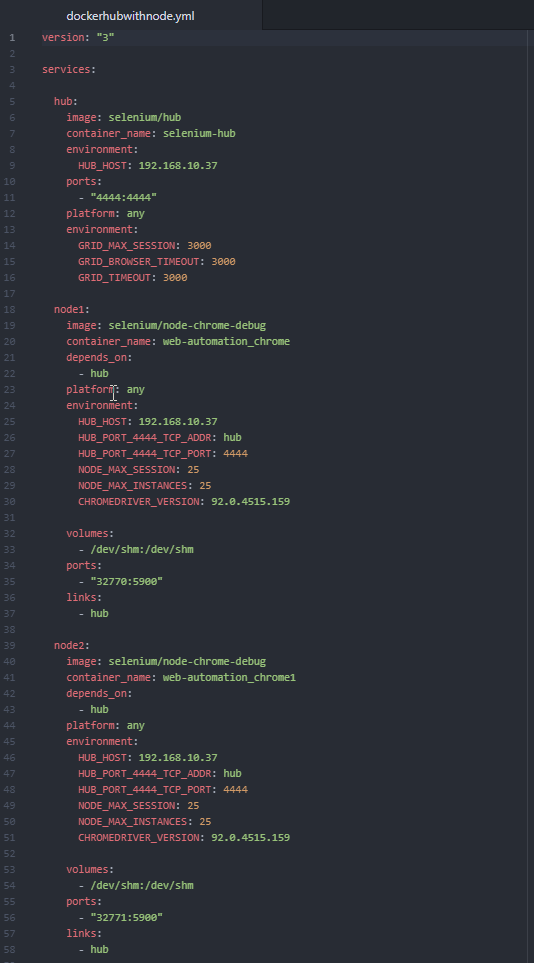
After pulling the selenium hub properly the following lines will be displayed.



After pulling the images to the docker we can see the downloaded images displayed in the Docker Desktop Images



To Run Selenium Grid Hub along with the node, use the following config file mentioned below



To Start hub and node run the config file in command prompt enter the following command

**docker-compose -f path-of-your-file\dockerhubwithnode.yml up**

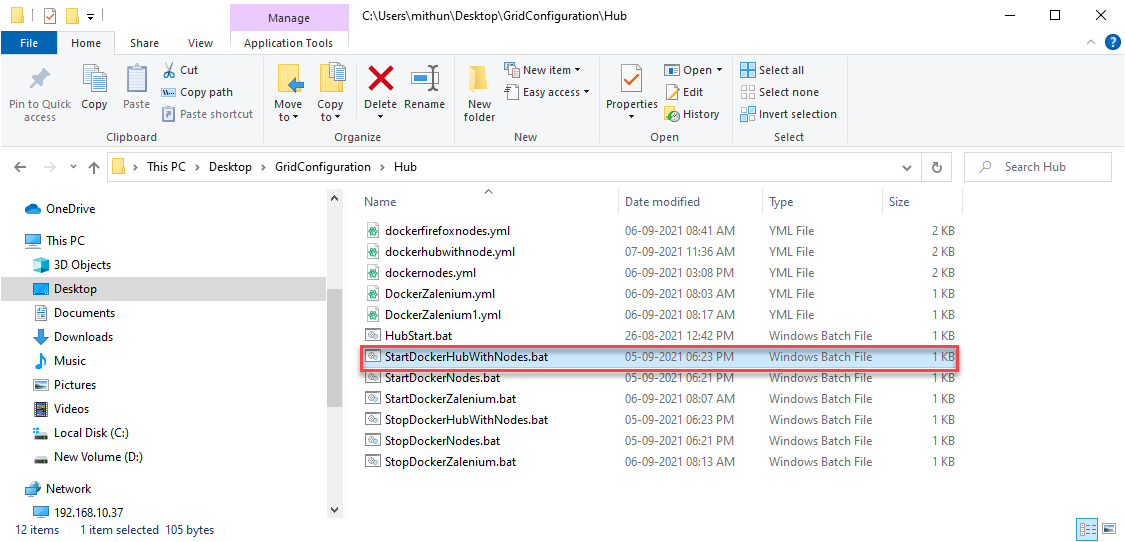
in the place of path specify your .yml file path placed in your machine.

Then to run the config file in a batch file paste the following lines in the notepad and save it as .bat file.

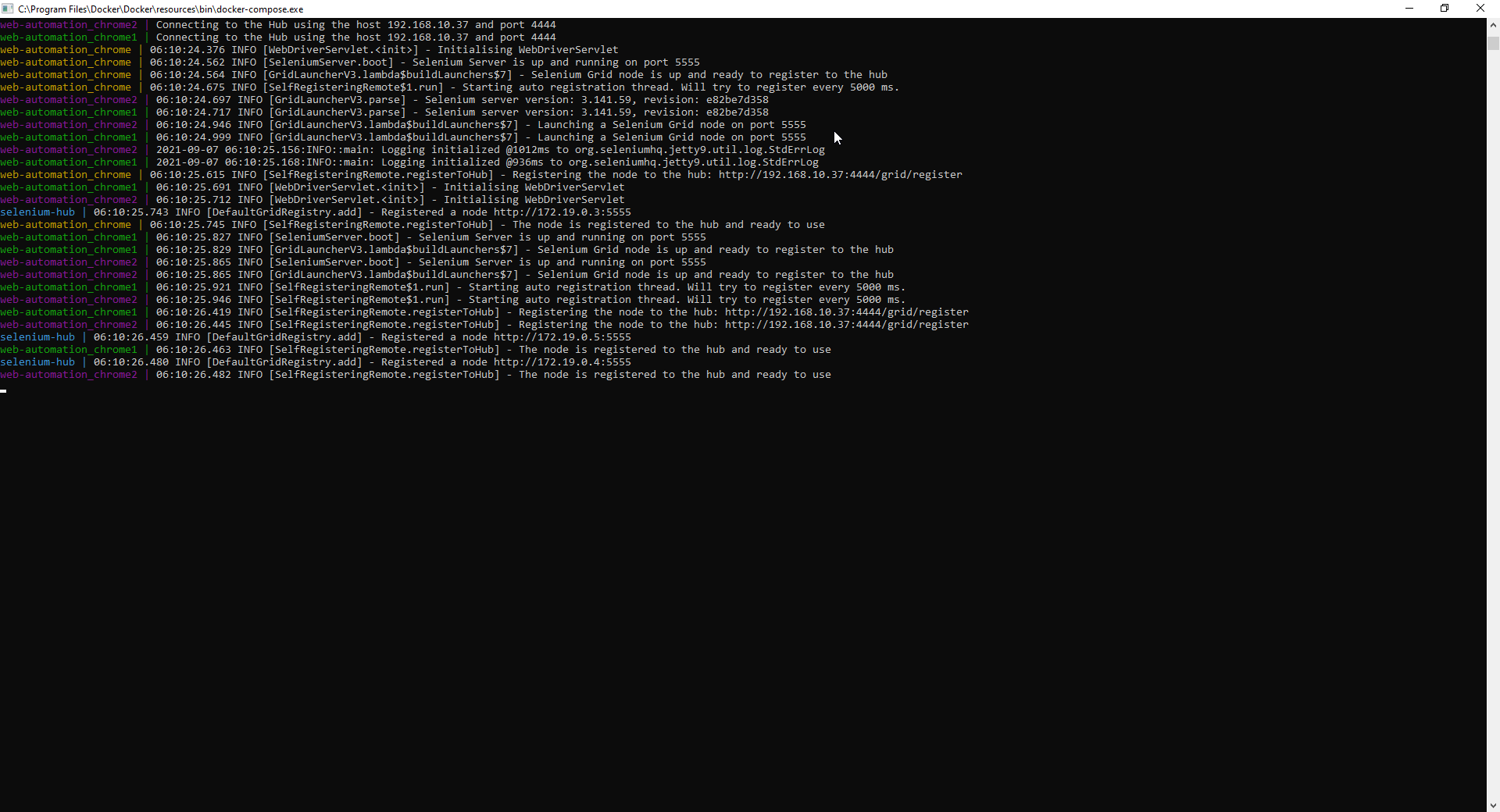
**cmd /C start/MIN docker-compose -f C:\Users\mithun\Desktop\GridConfiguration\Hub\dockerhubwithnode.yml up**

in the place of path specify your .yml file path placed in your machine.

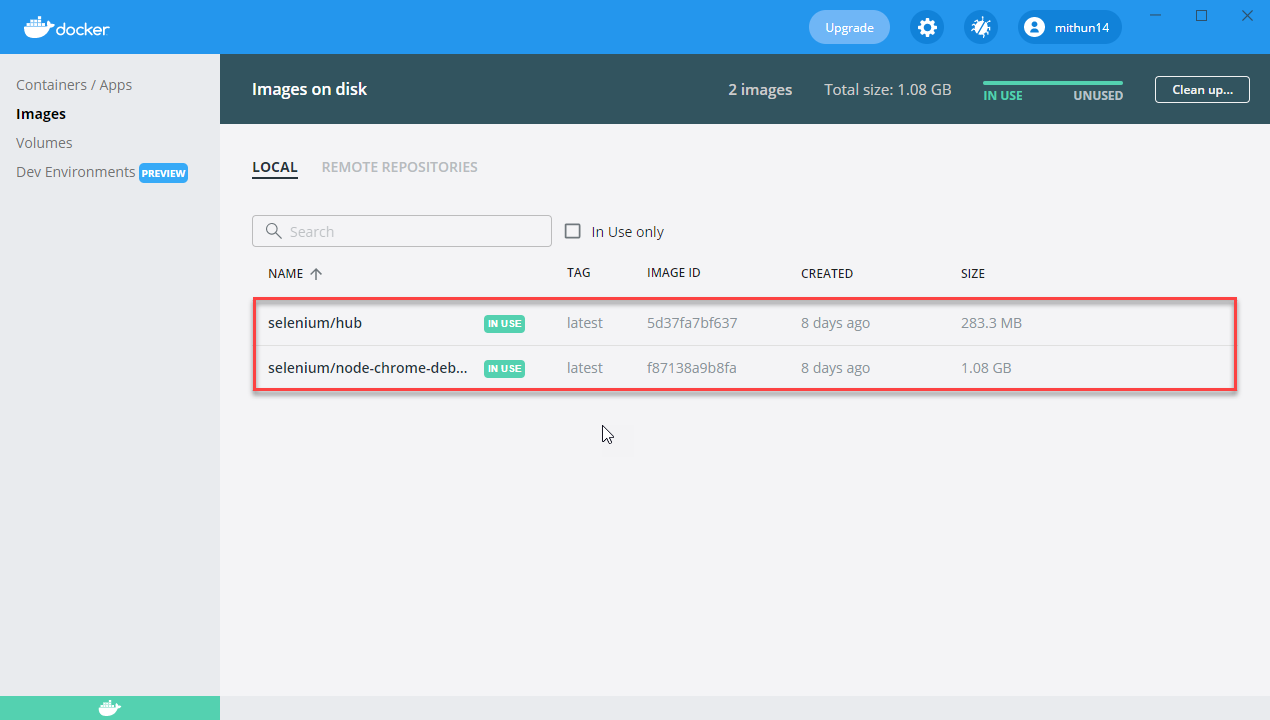
Double click your batch file created to start the grid hub with node in docker



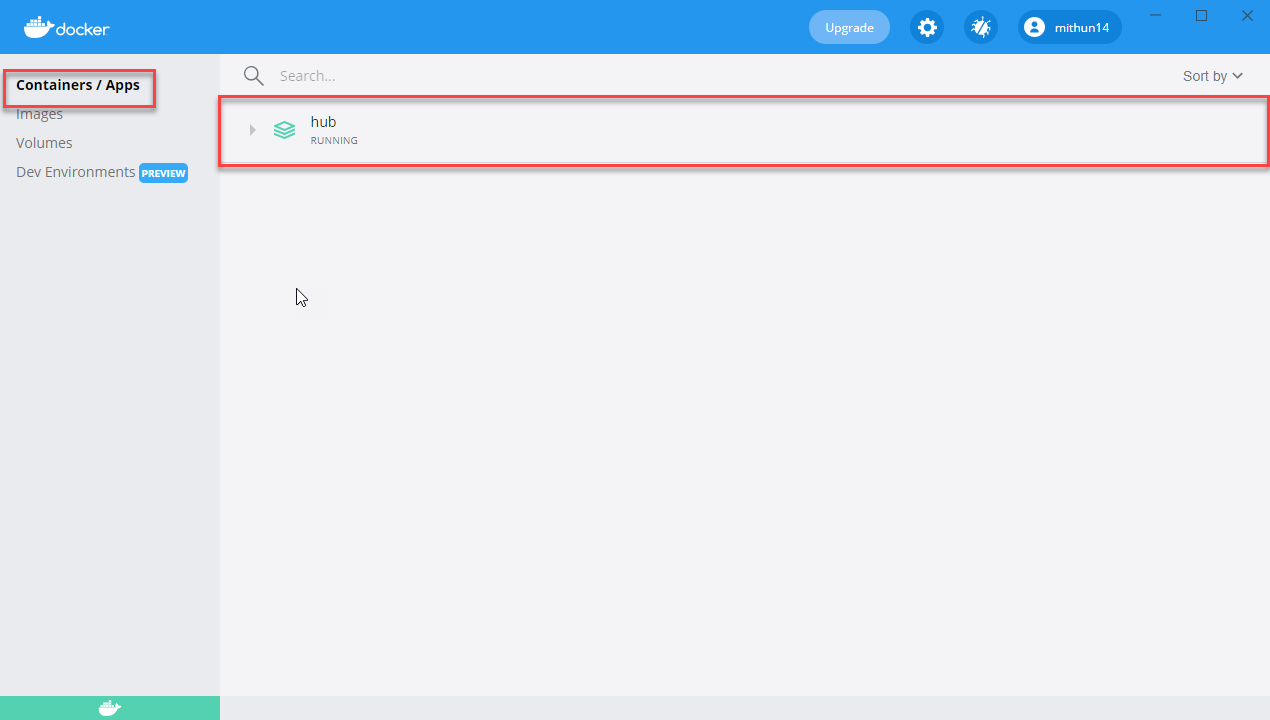
Once after started the batch file you can verify in command prompt the hub and node started and connected to hub.



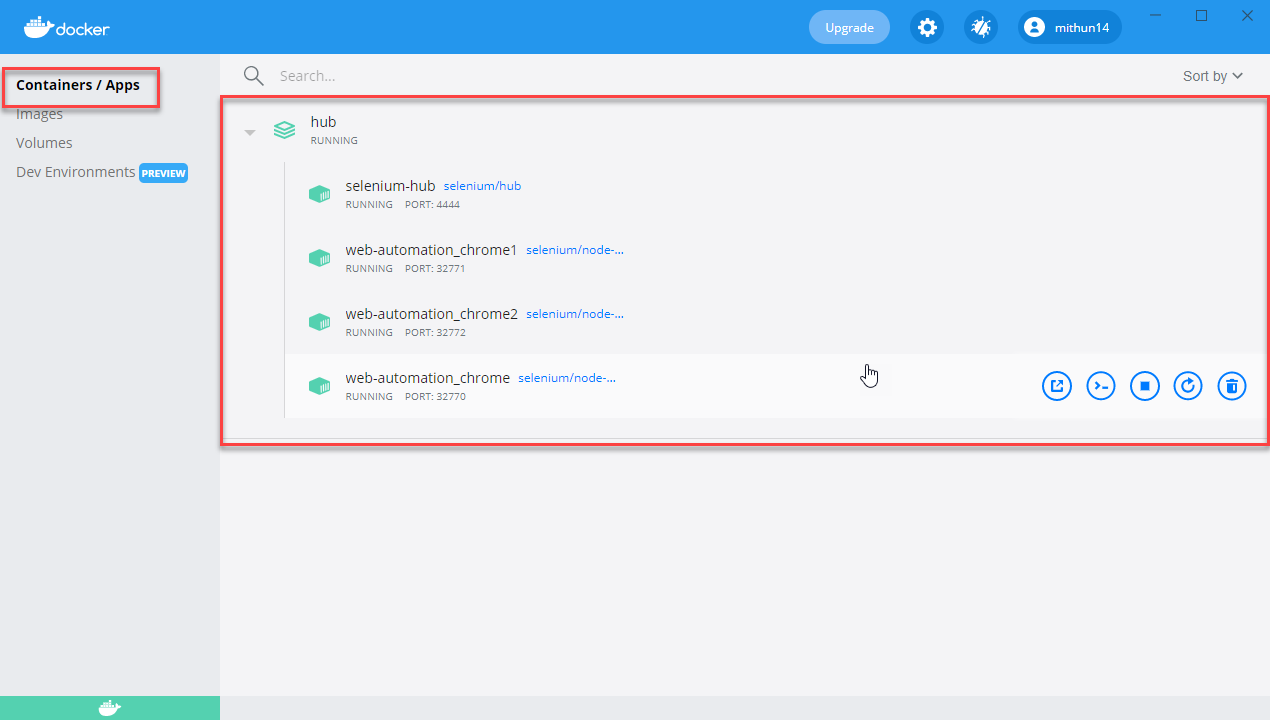
Once after started the batch file you can verify that both the hub and node are in use in the Docker Desktop.



And in the Containers/Apps in the Docker Desktop you can see the Container Running based on the .yml file config, It will take the folder name where your .xml file have been kept as a container name



If you expand the container you can check all the nodes listed along with the hub based on the .yml configuration



Change the following lines in our Playback.java class to run the scripts in Docker

**case** "Chrome": {

**final** String nodeUrl = "http://192.168.10.37:4444/wd/hub";

System.*setProperty*("webdriver.chrome.driver","/usr/bin/chromedriver");

DesiredCapabilities chromeOption = **new** DesiredCapabilities();

chromeOption.setBrowserName("chrome");

chromeOption.setPlatform(Platform.***LINUX***);

ChromeOptions option = **new** ChromeOptions();

option.merge(chromeOption);

option.addArguments("--no-sandbox");

option.addArguments("disable-infobars");

option.addArguments("--disable-extensions");

option.addArguments("--disable-setuid-sandbox");

option.addArguments("--headless");

option.addArguments("--start-maximized") ;

option.addArguments("--disable-dev-shm-usage");

option.addArguments("--disable-gpu");

option.addArguments("--disable-popup-blocking");

option.addArguments("--disable-notifications");

option.addArguments("disable-infobars");

option.addArguments("--disable-translate");

**try** {

Playback.*driver* = **new** RemoteWebDriver(**new** URL(nodeUrl),option);

}

**catch** (MalformedURLException e) {

e.printStackTrace();

}

Playback.*driver*.manage().deleteAllCookies();

Playback.*driver*.manage().window().maximize();

Playback.*driver*.manage().timeouts().implicitlyWait(60, TimeUnit.***SECONDS***);

Playback.*driver*.manage().timeouts().pageLoadTimeout(60, TimeUnit.***SECONDS***);

**break**;

}

Then you can execute your scripts from eclipse it will run the scripts in the docker container and we can check our results in the share folder based on extend report configuration.

To Stop hub and node run the config file in command prompt enter the following command

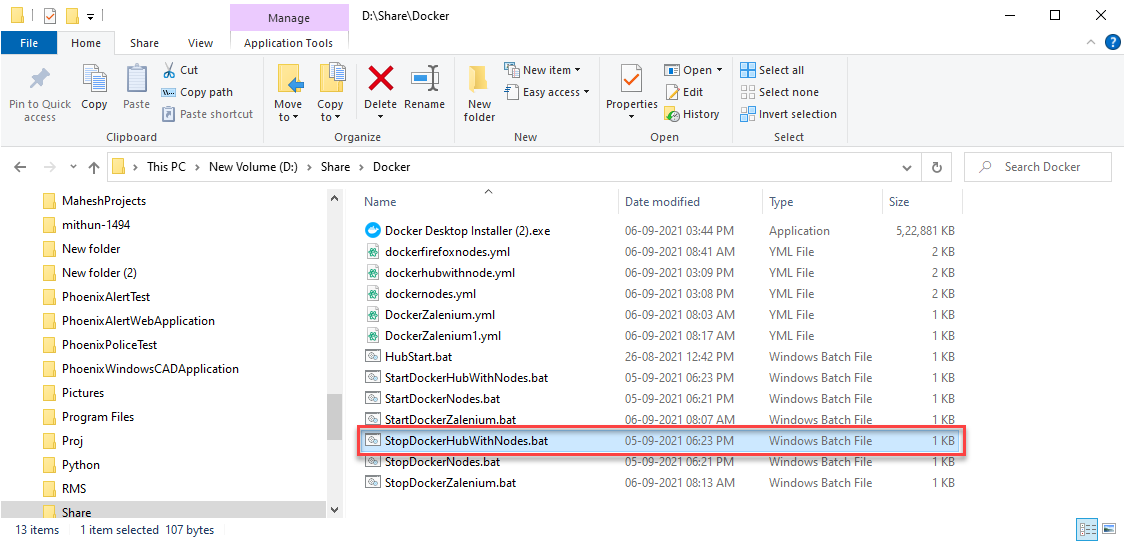
**docker-compose -f path-of-your-file\dockerhubwithnode.yml down**

in the place of path specify your .yml file path placed in your machine.

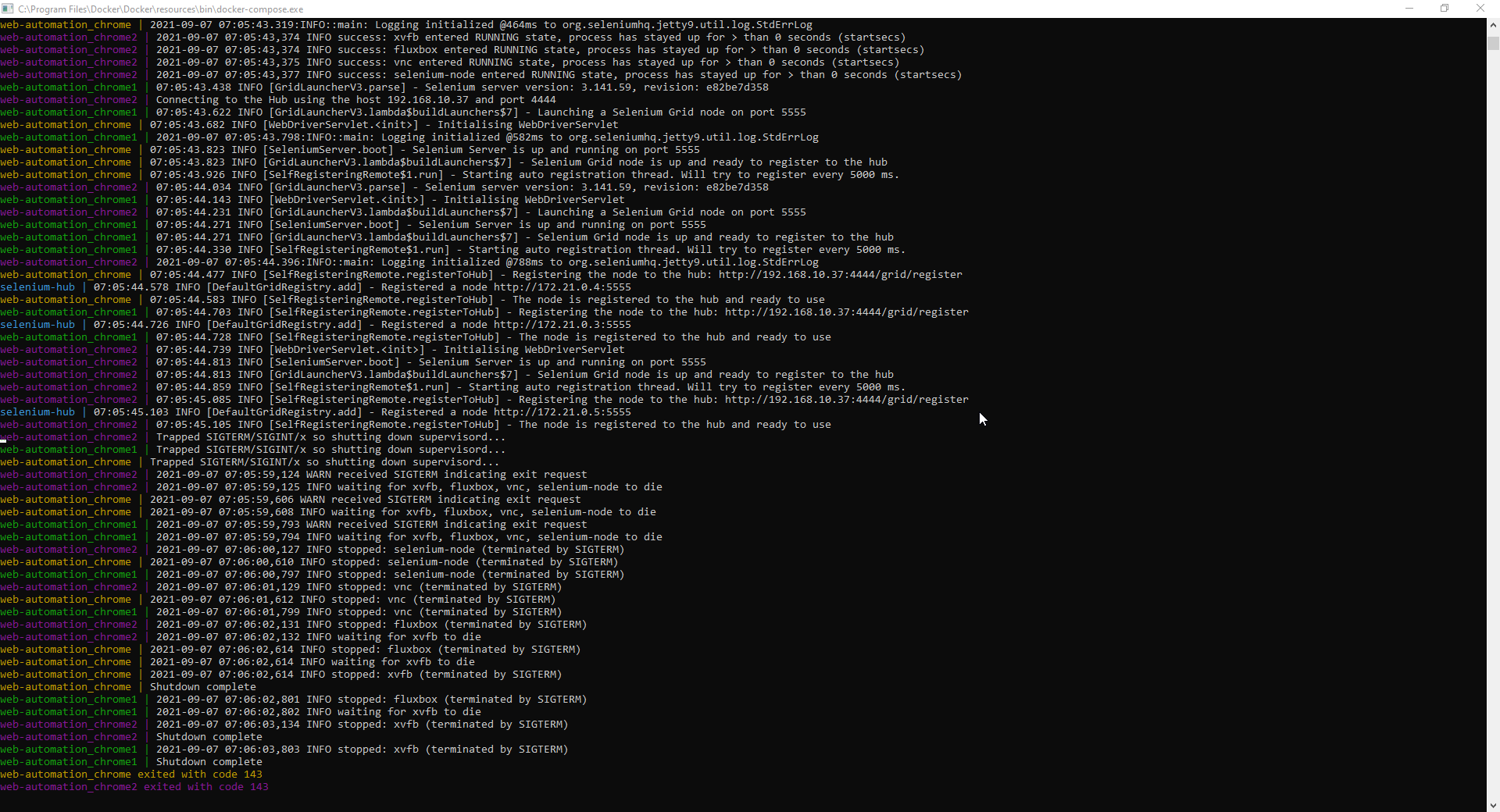
To stop the hub and node running in docker batch file paste the following lines in the notepad and save it as .bat file.

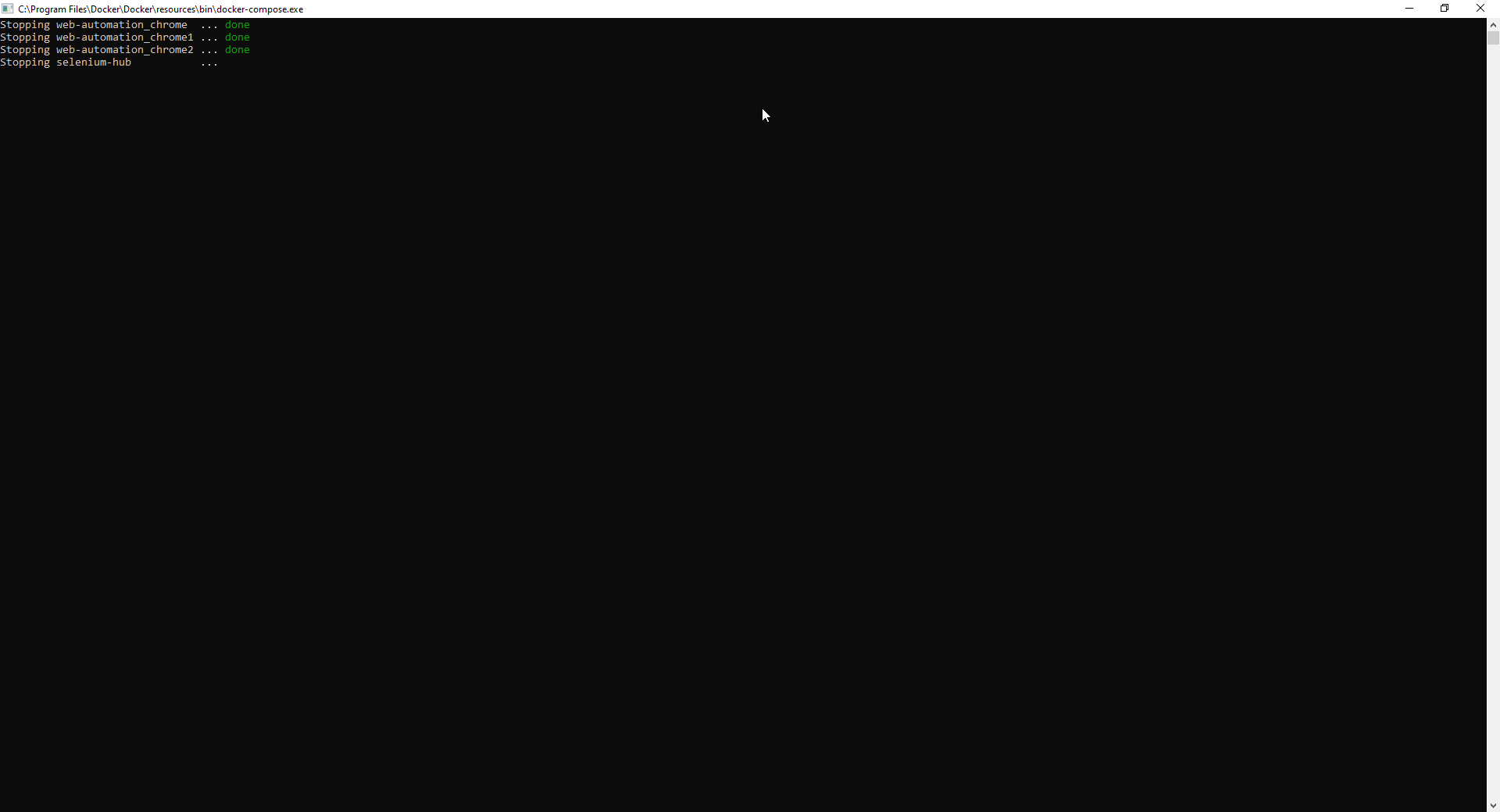
**cmd /C start/MIN docker-compose -f C:\Users\mithun\Desktop\GridConfiguration\Hub\dockerhubwithnode.yml down**

Double click your batch file created to stop the grid hub with node in docker

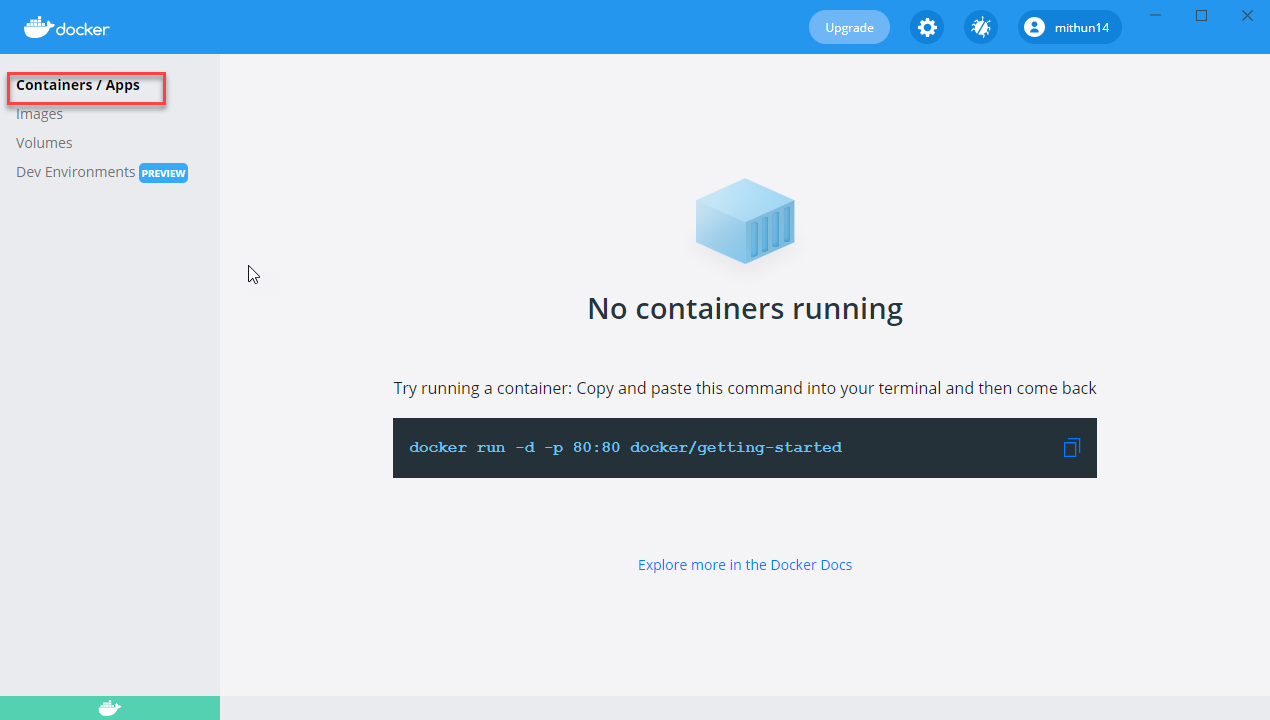


Once after started the batch file you can verify in command prompt the hub and node stopped and container instances will be removed.





Once the batch file executed you can see the container has been removed from the docker Desktop App



And you can check Images in use tag has been removed

