Assignment

NOTE: Do not forget to see the manual page using the "--help" option in command when searching for options/commands for a particular task.

1. Install Docker, either on your native OS or on a VM. Make sure it runs. Type "docker -v" to check if it's installed.

If you can't install or configure Docker, you can use the online docker setup to do the assignment.

Step1 Goto:- https://www.katacoda.com/courses/kubernetes/playground

Step2 Click on the "Continue" button on the left panel

Step3 Click on the "launch. sh" button on the left panel

Step4 From the right panel use the top console to execute the below command:-

docker -v

Try the below commands for help

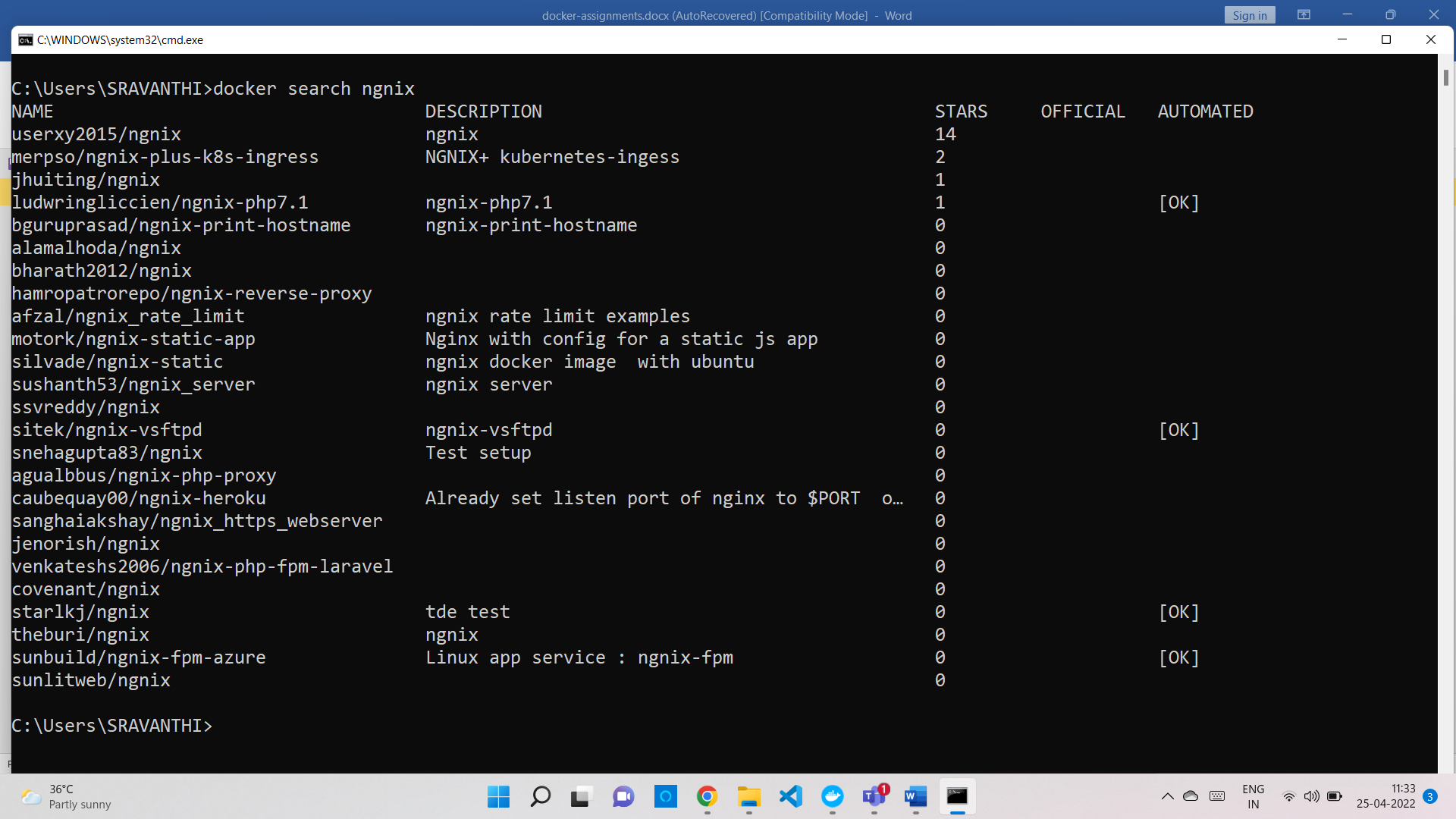
docker --help ---> This command shows all available options and commands to work with images and containers

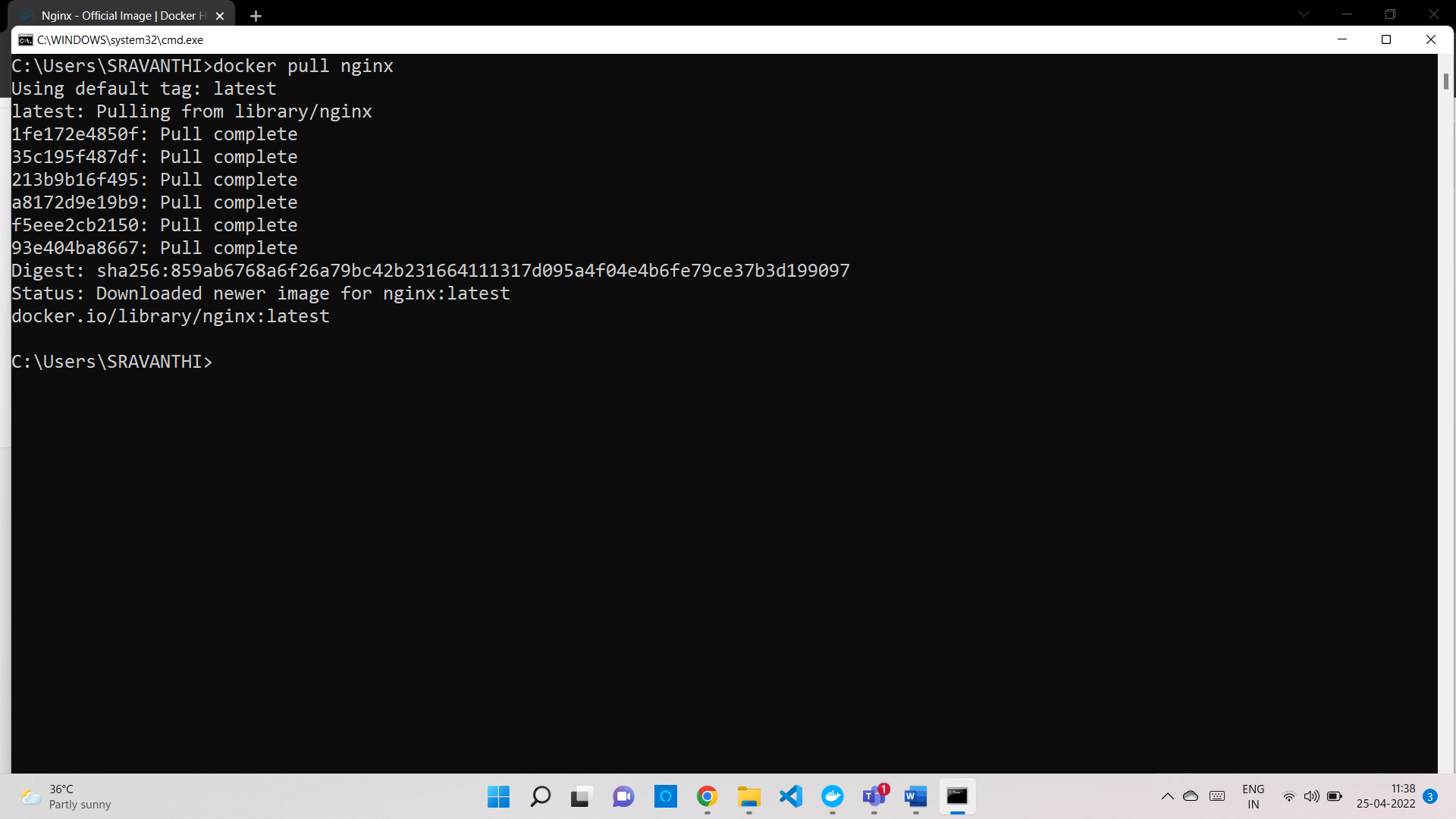
docker image --help ---> This command shows all the available options and commands to work with docker images

docker container --help ---> This command shows all the available options and commands to work with docker containers

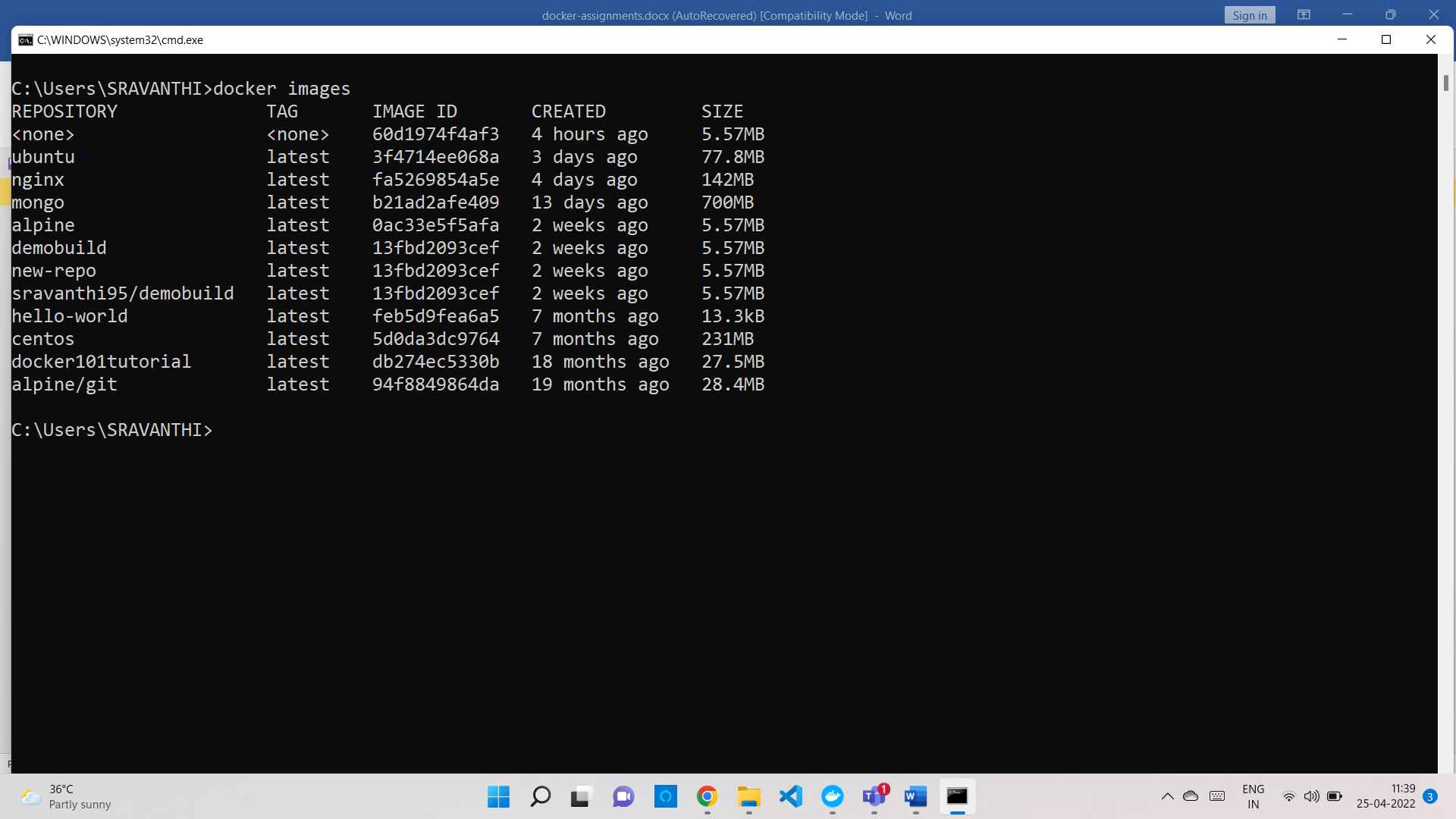
NOTE:- DO NOT TRY TO USE INTERNET TO SOLVE ASSIGNMENT, BETTER USE THE ABOVE --help OPTION TO SEE THE MANUAL OF ANY PARTICULAR COMMAND AND FIGURE OUT THE SOLUTIONS ON YOUR OWN.

2. Find an image from the docker hub of your choice(recommended: Nginx), don't use a browser, pull the official image from the docker hub : docker search ngnix





3. List all the available images in your machine/VM, make sure you see recently pulled images in the list.



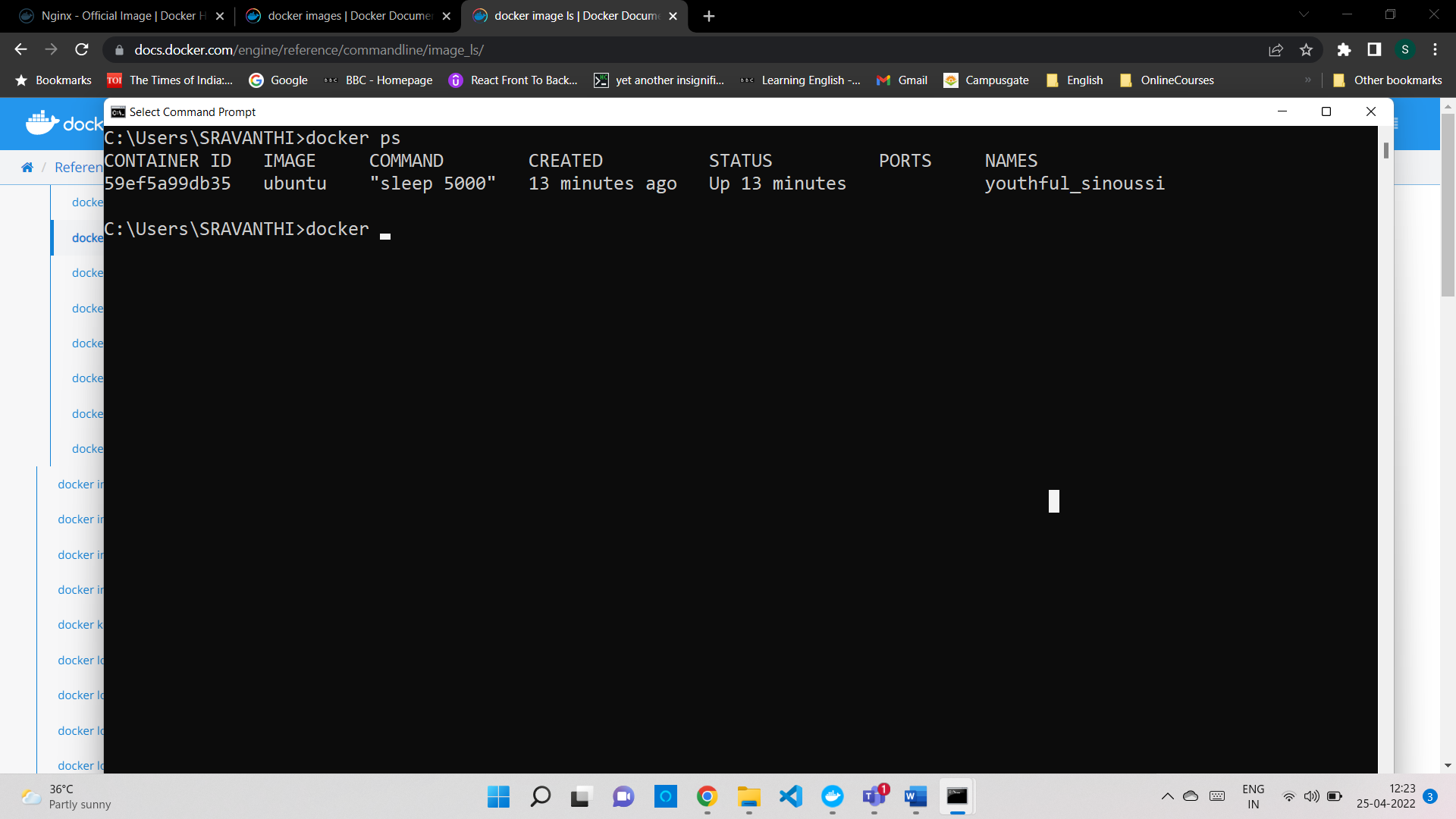
4. Find out the "Full" ImageId of the image that you pulled and write it below.

sha256:3f4714ee068a59a09d9e77de71ec1254e5916d6e5779140bc96cec7d0edea18d

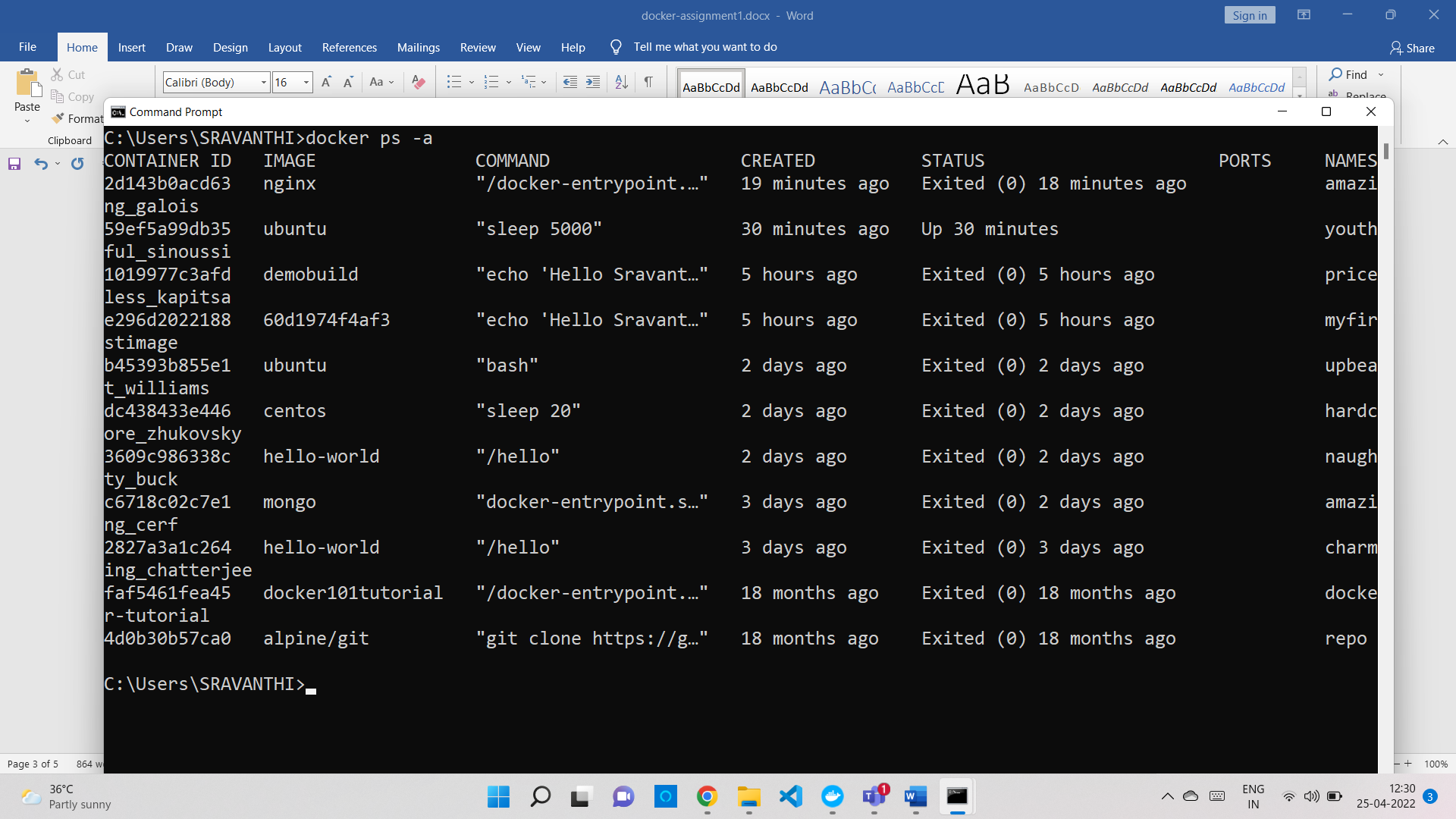
Docker images –filter = “reference name=ubuntu” –quiet –no-trunc

5. Create a container of your image

6. List all the running containers **docker ps**

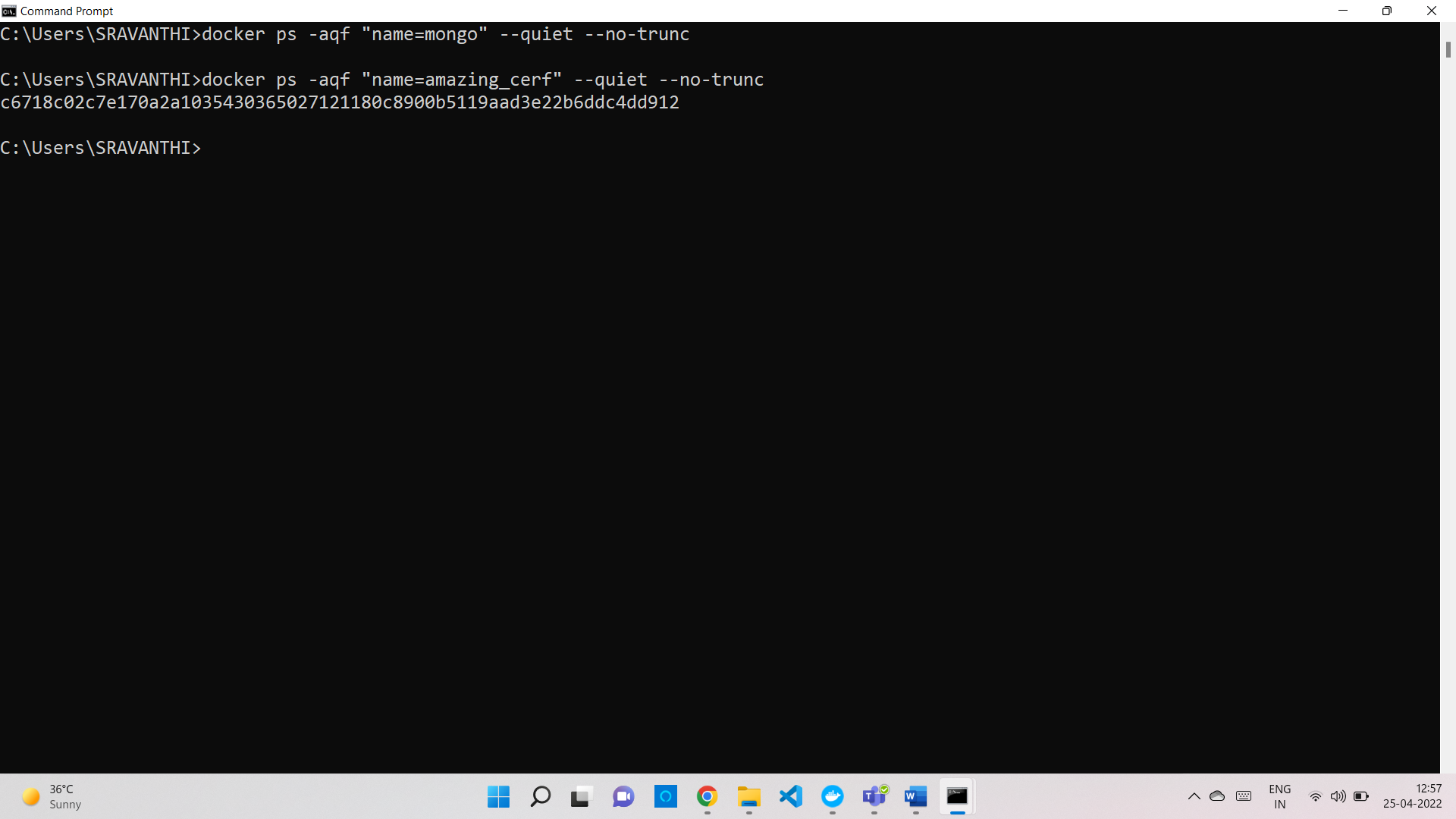


7. List all the running and stopped containers **docker ps -a**



8. Find out the "Full" containerId of the container and write it below.

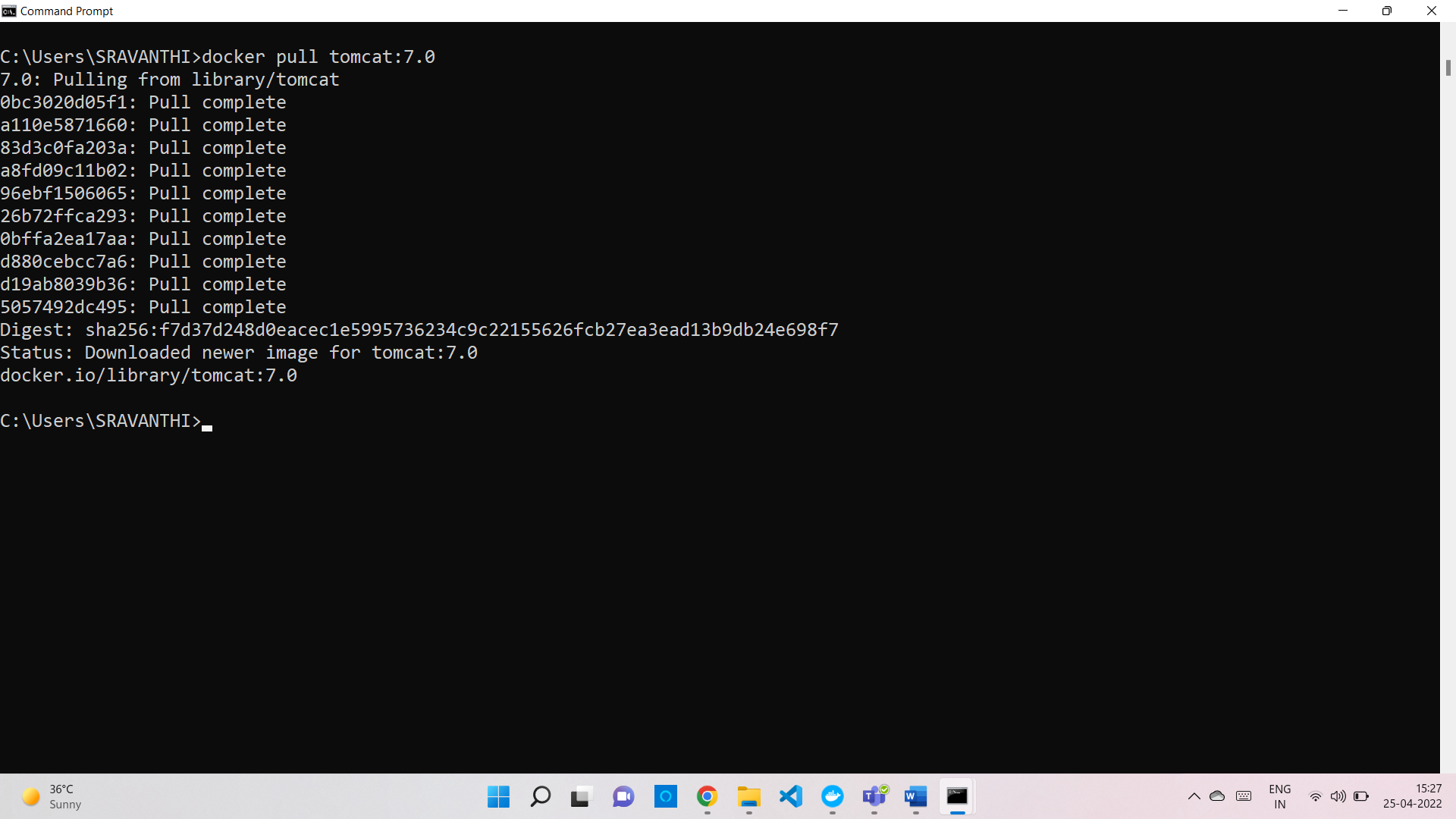
c6718c02c7e170a2a1035430365027121180c8900b5119aad3e22b6ddc4dd912



9. Find out how many image layers are used to build this image.

10. Get the Apache Tomcat 7 server image from the docker hub.

**docker pull tomcat:7.0**

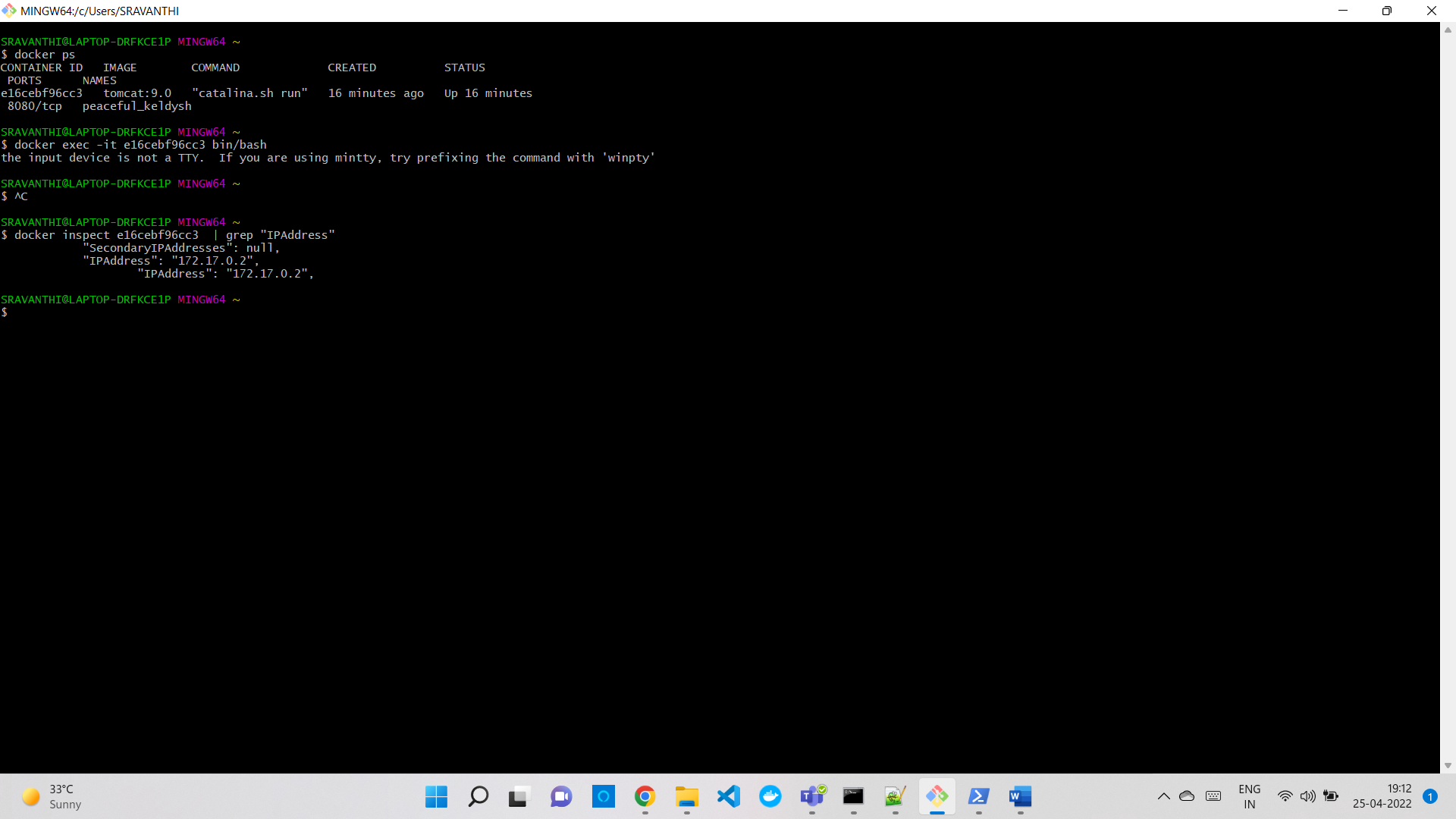


11. Run the Apache Tomcat 7, I mean create a container of Apache Tomcat.

docker run -it --rm -p 8080:8080 tomcat

12. Find out what is the IP Address of the Apache Tomcat Container that it is running on

docker inspect e16cebf96cc3 | grep "IPAddress"

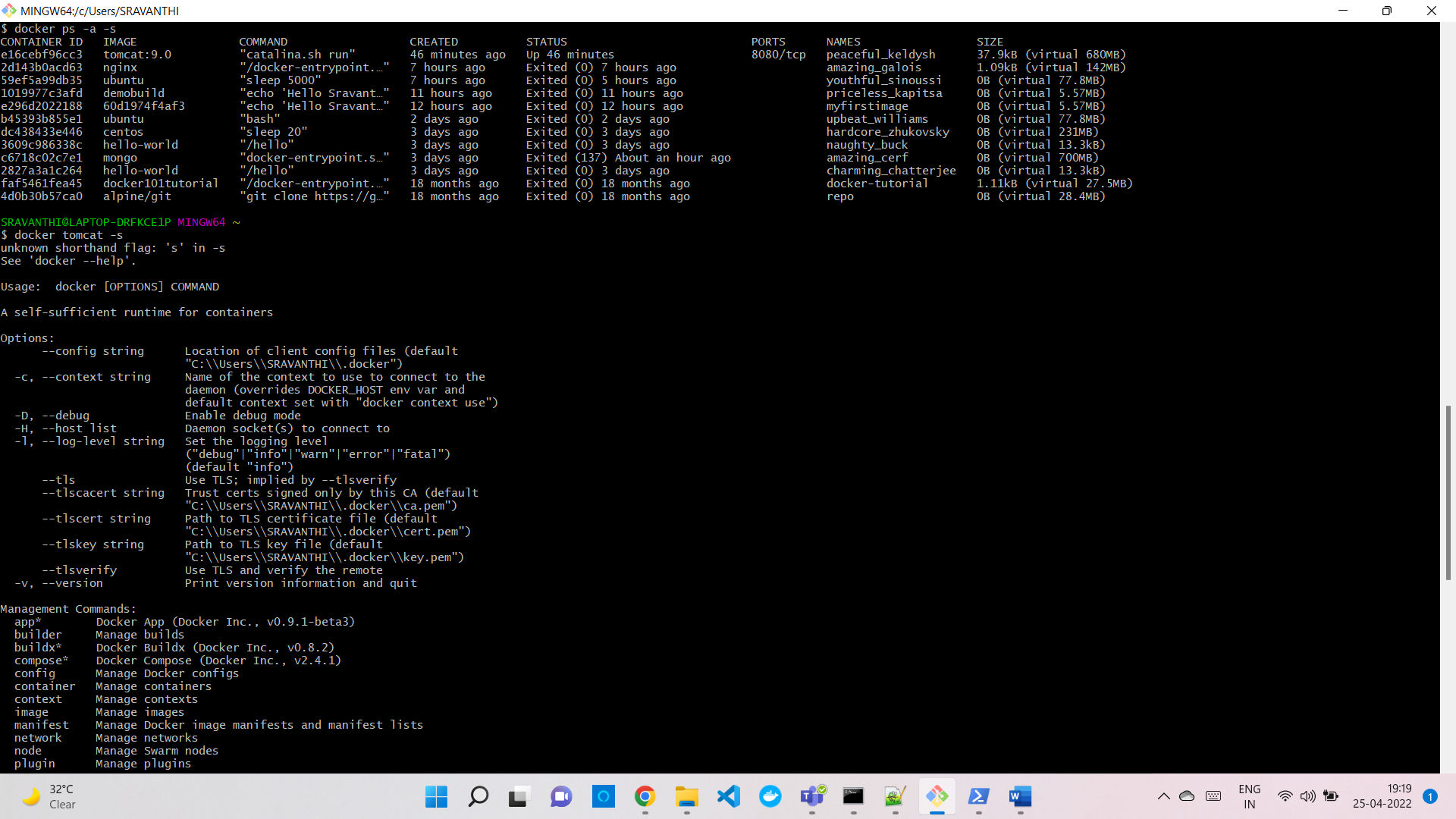


13. Which Port it is using?

14. Try to access Tomcat's home page from your machine/vm.

15. What is the disk size of the Apache Tomcat image?

docker ps -a -s



16. Find out a list of all environment variables that are configured for the tomcat image, can you see JAVA\_HOME and CATALINA\_HOME? What did you notice about it?

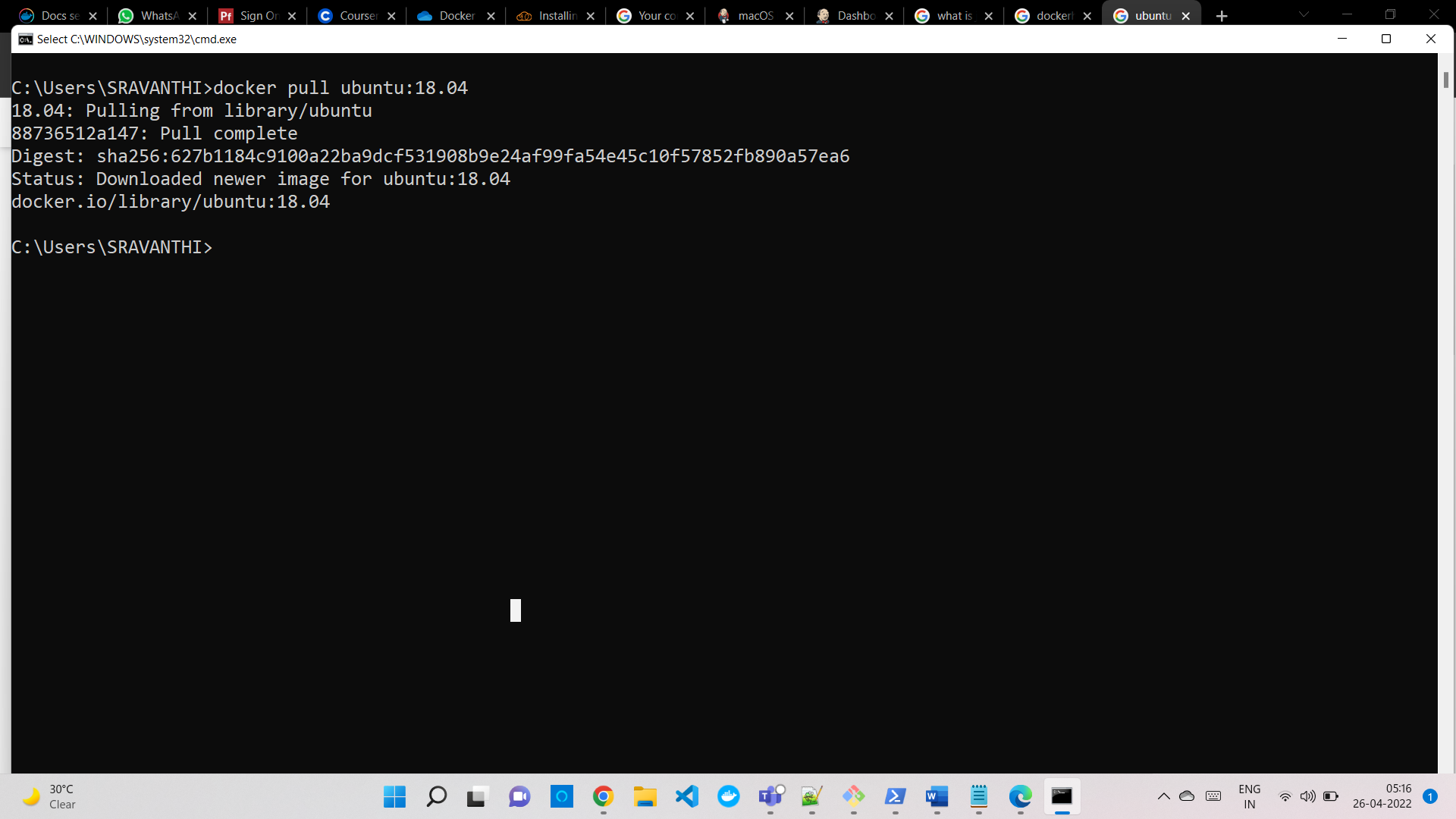
17. Find out which port is exposed for tomcat?

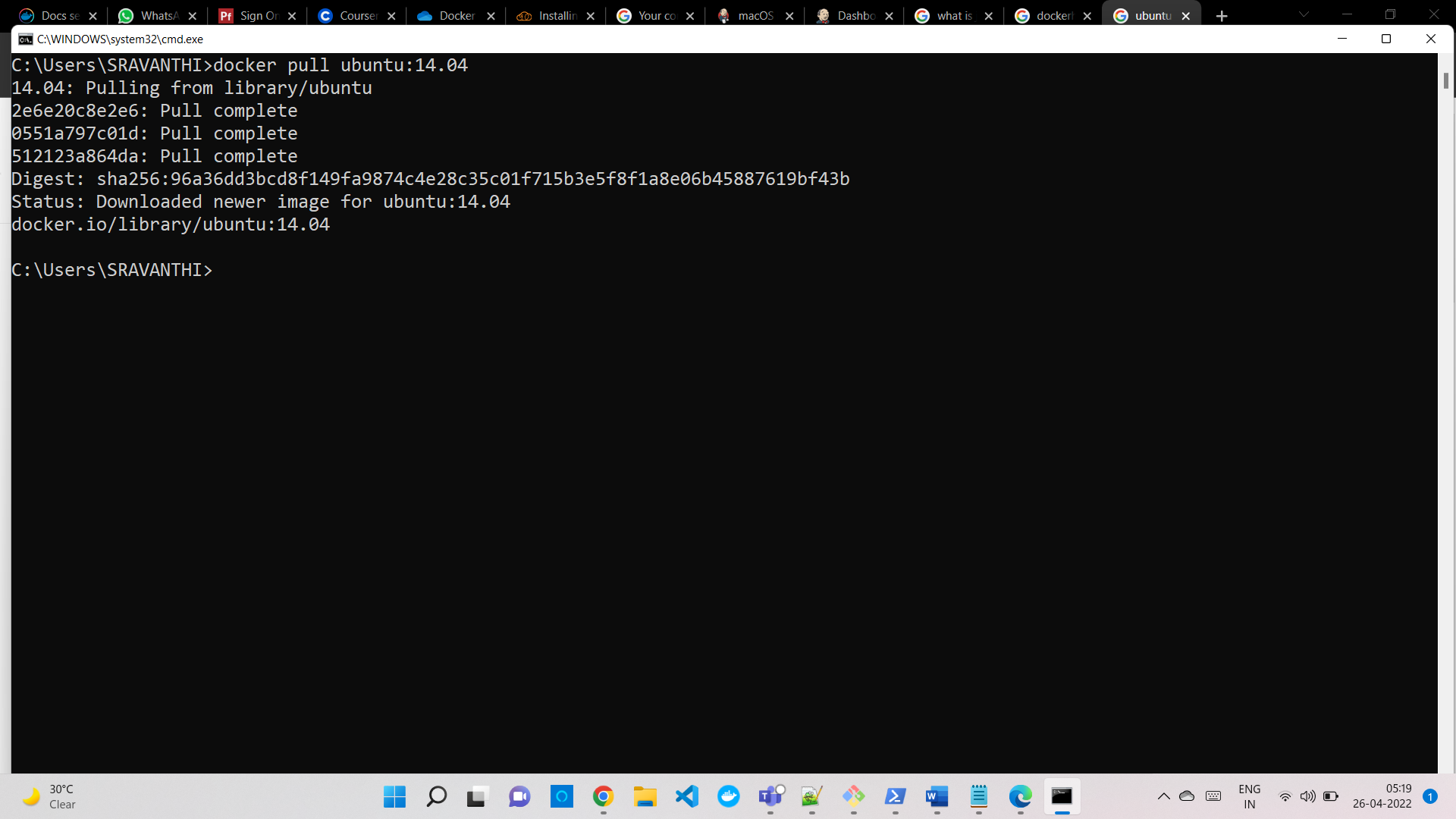
18. Run multiple containers of tomcat on a different port and access its home page.

19. Pull ubuntu os from docker hub, try to pull 2 images of ubuntu, Except the latest one.

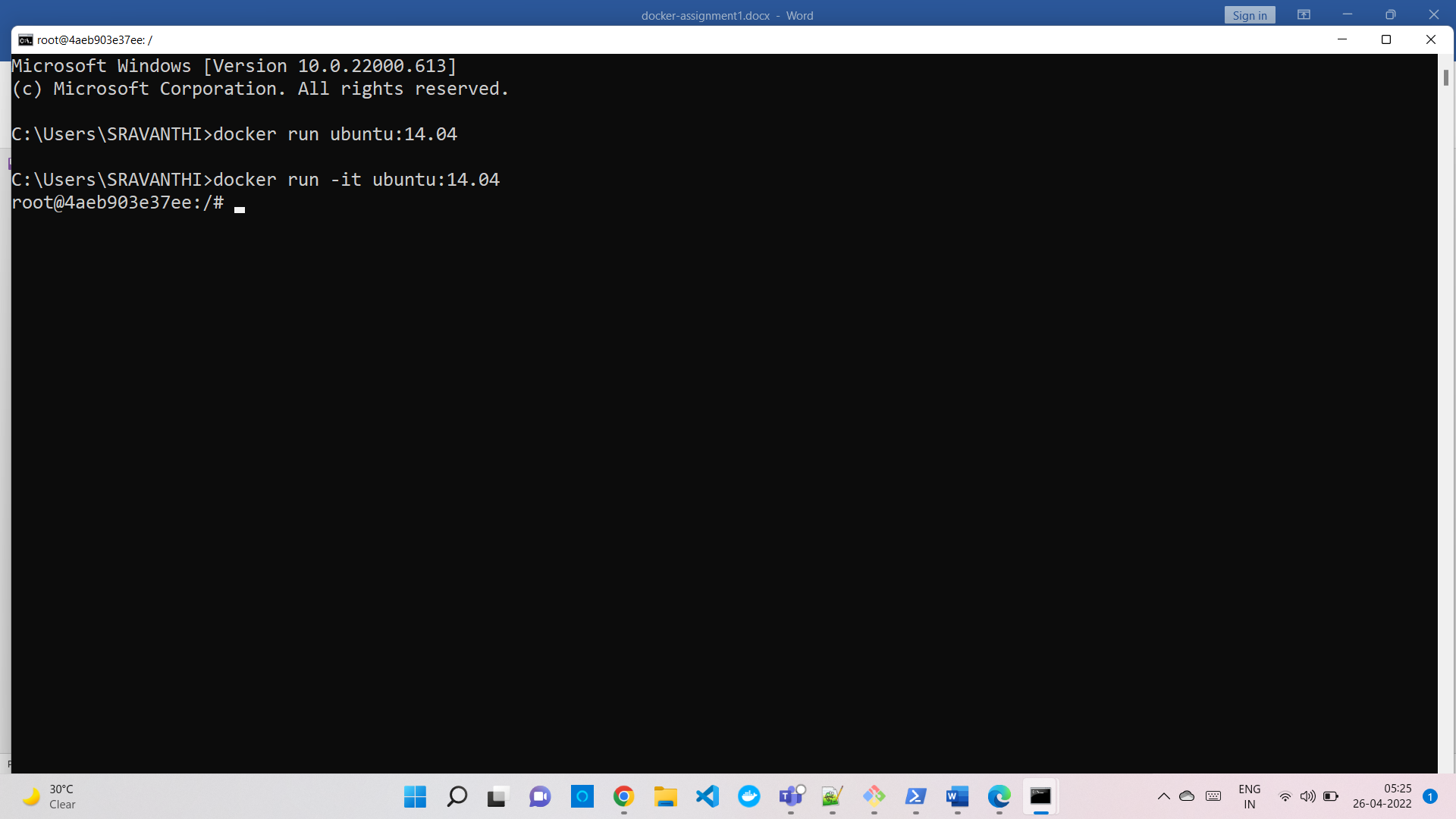
Docker pull ubuntu:18.04

Docker pull ubuntu:14.04

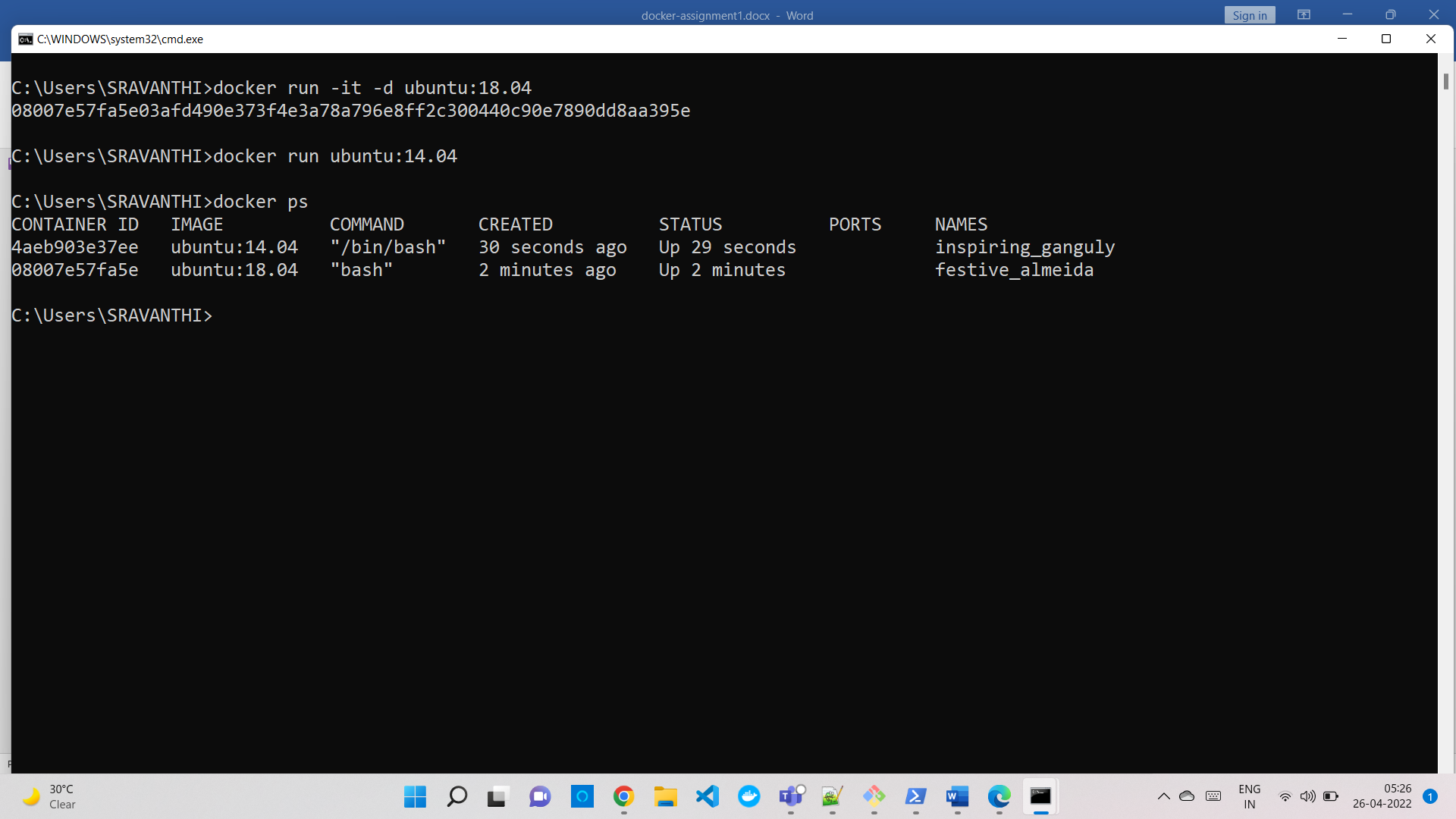




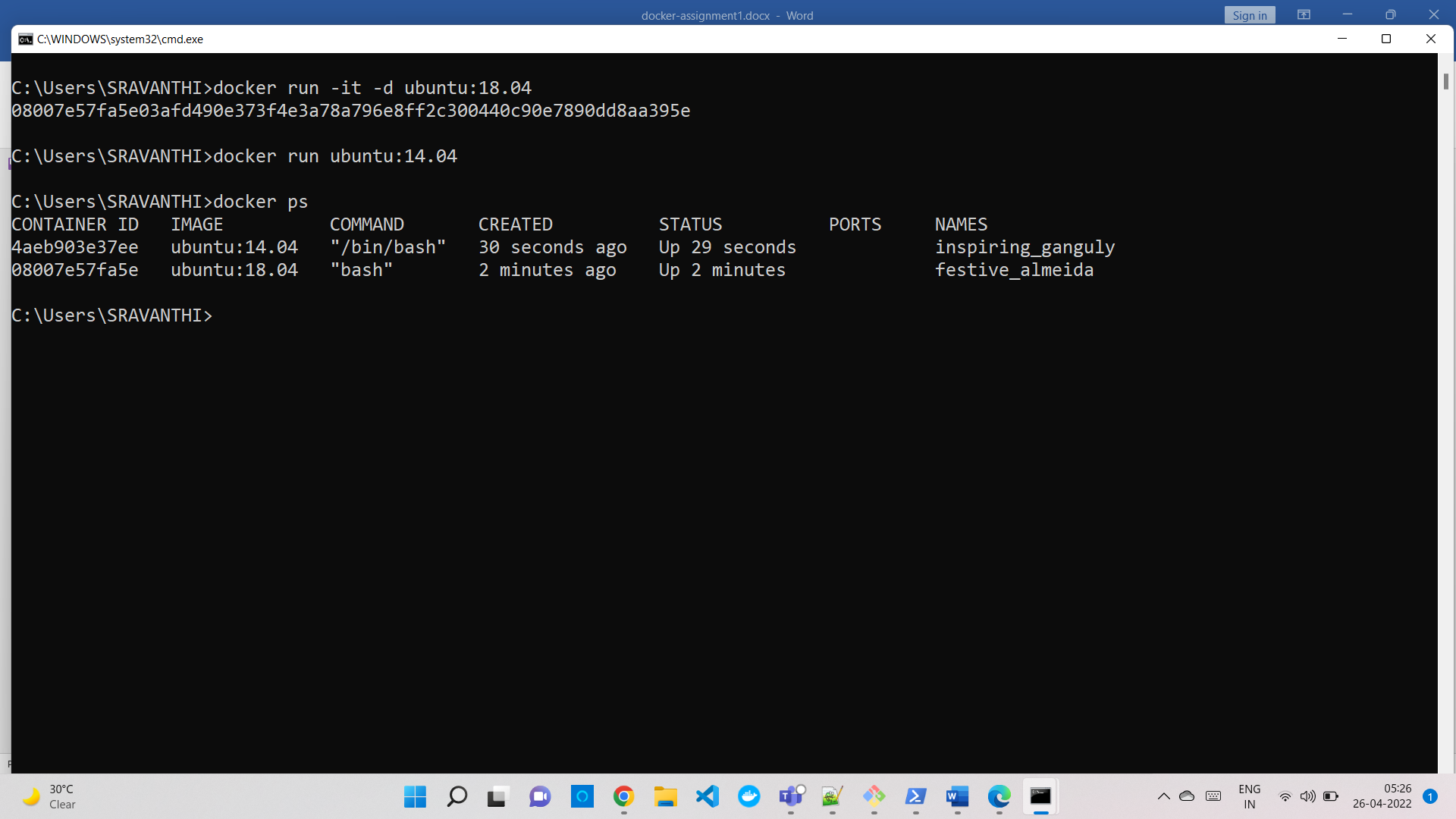
20. Run the container of ubuntu in attached mode.



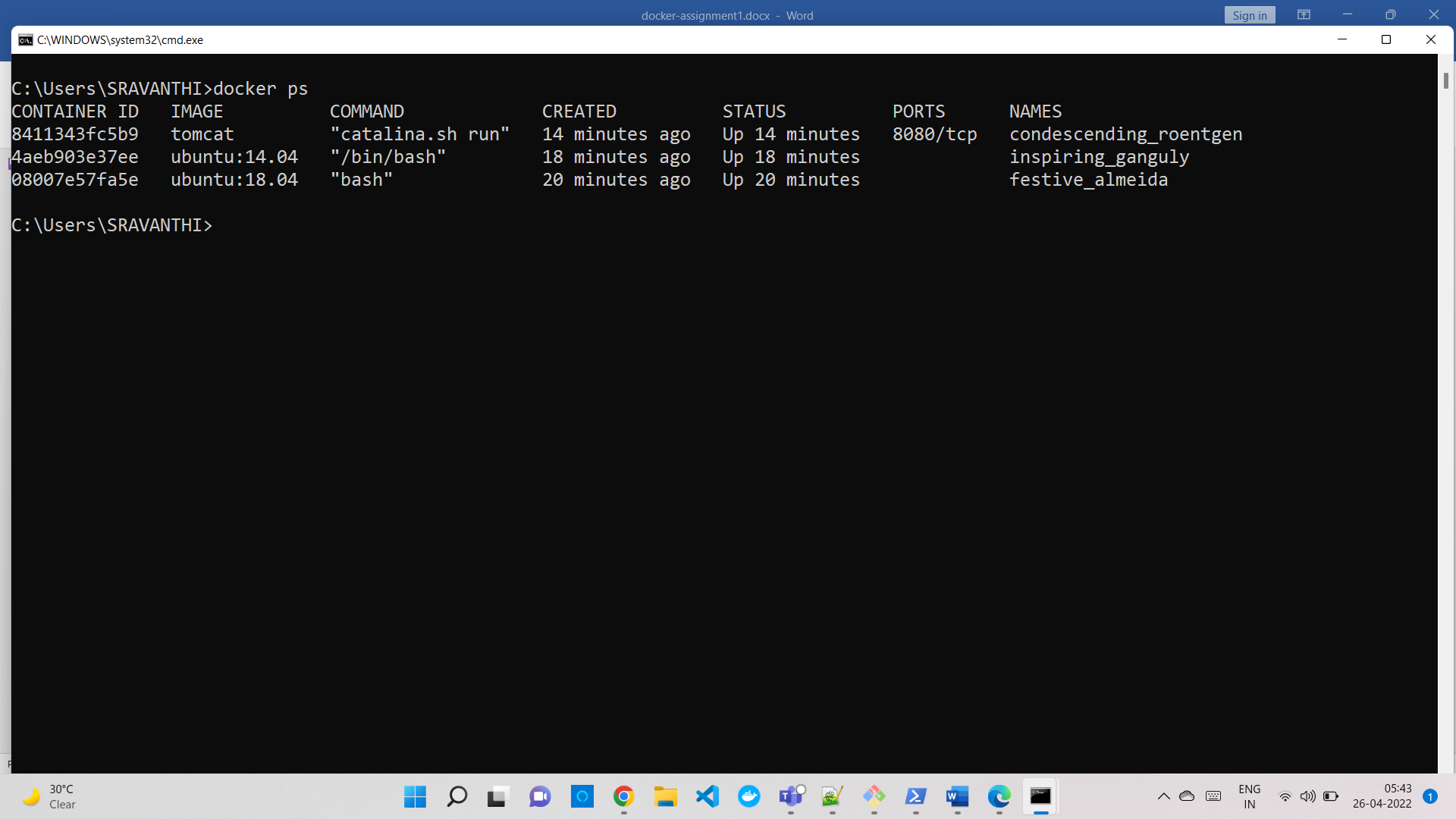
21. Run the container of another ubuntu in detached mode.



22. Check how many ubuntu containers are running and stopped



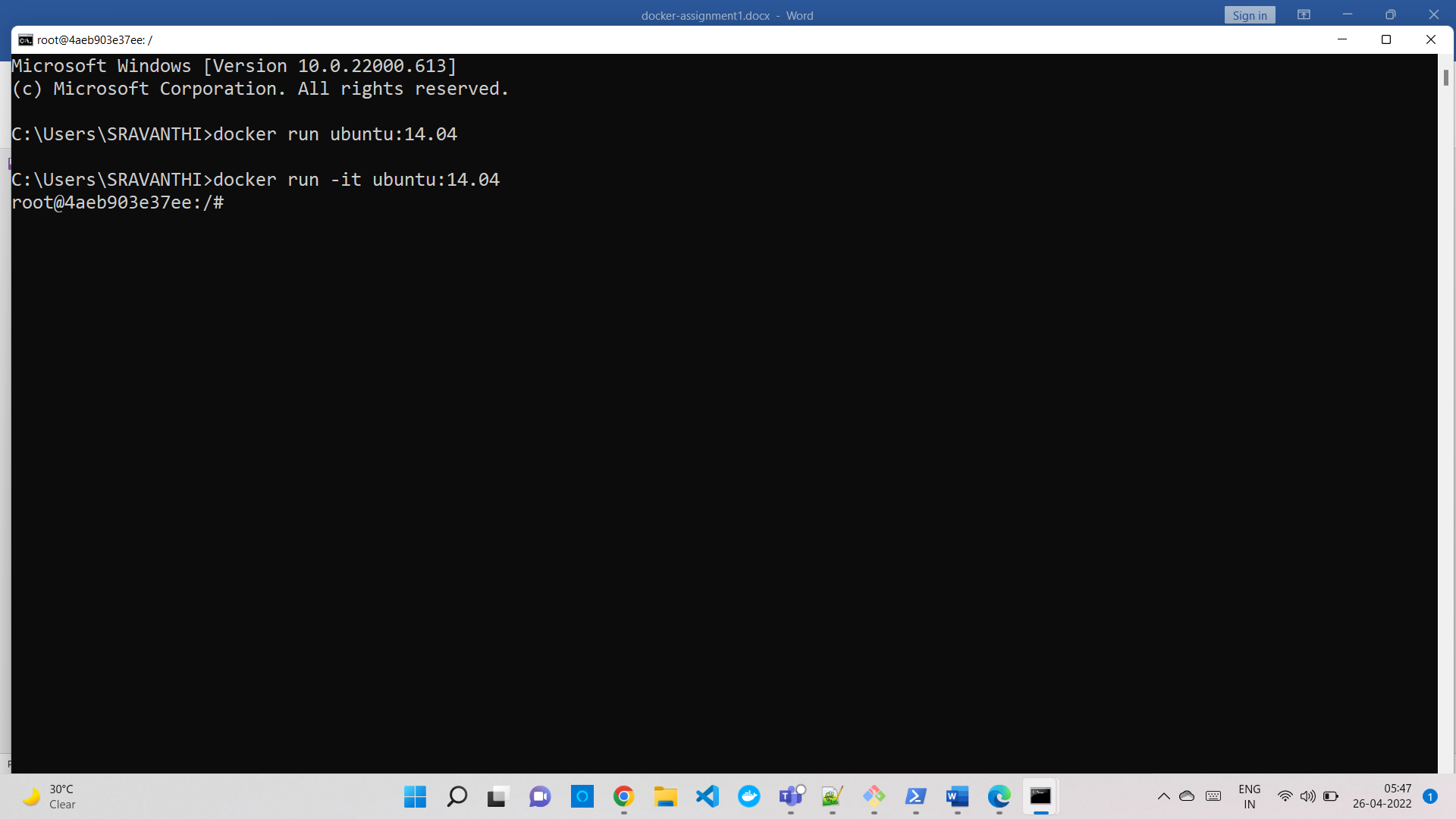
23. Is the tomcat container running? If no, start one.



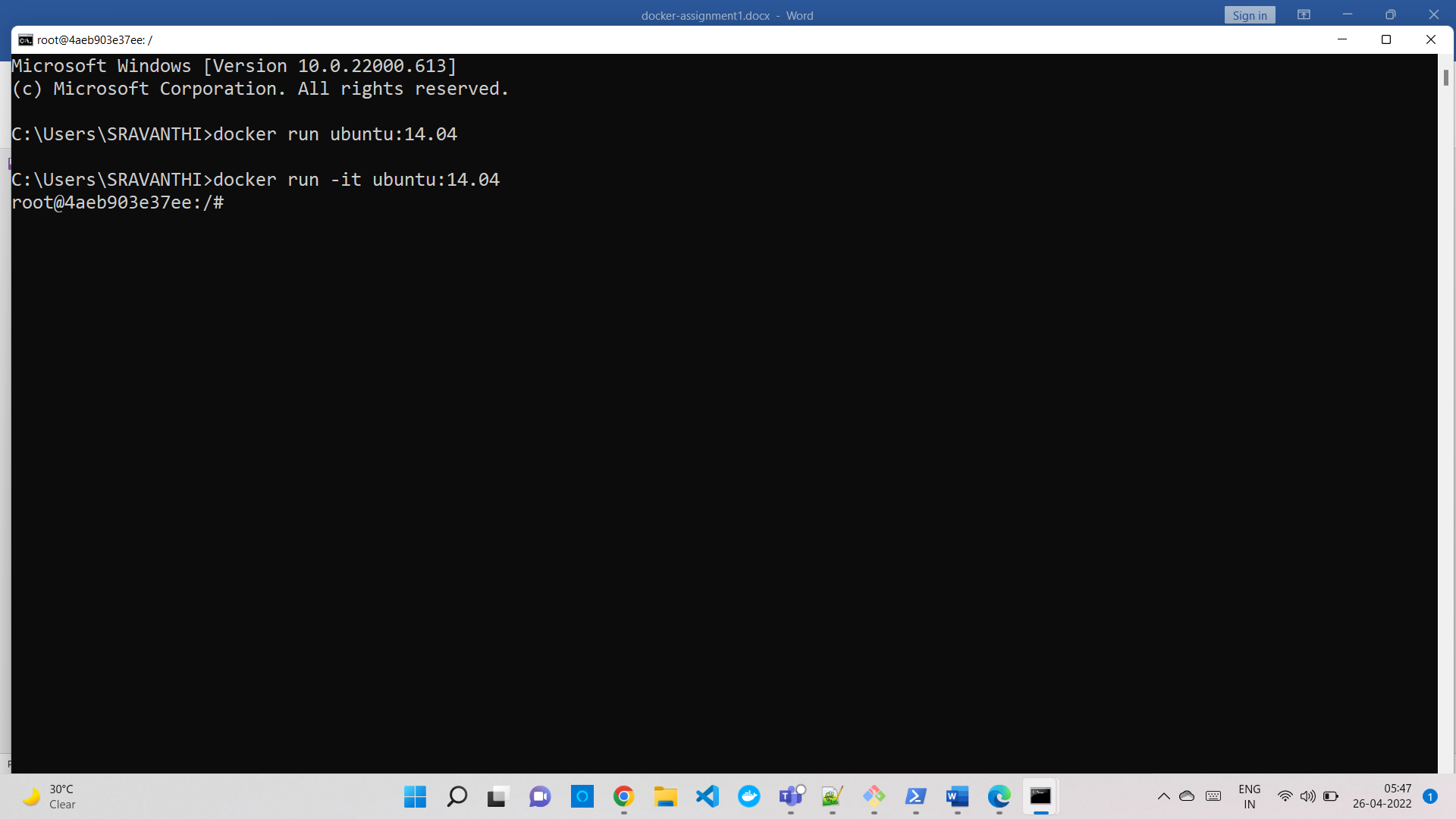
24. Check the logs, generated by the tomcat container(don't forget to make a request to tomcat's home page to see the log).



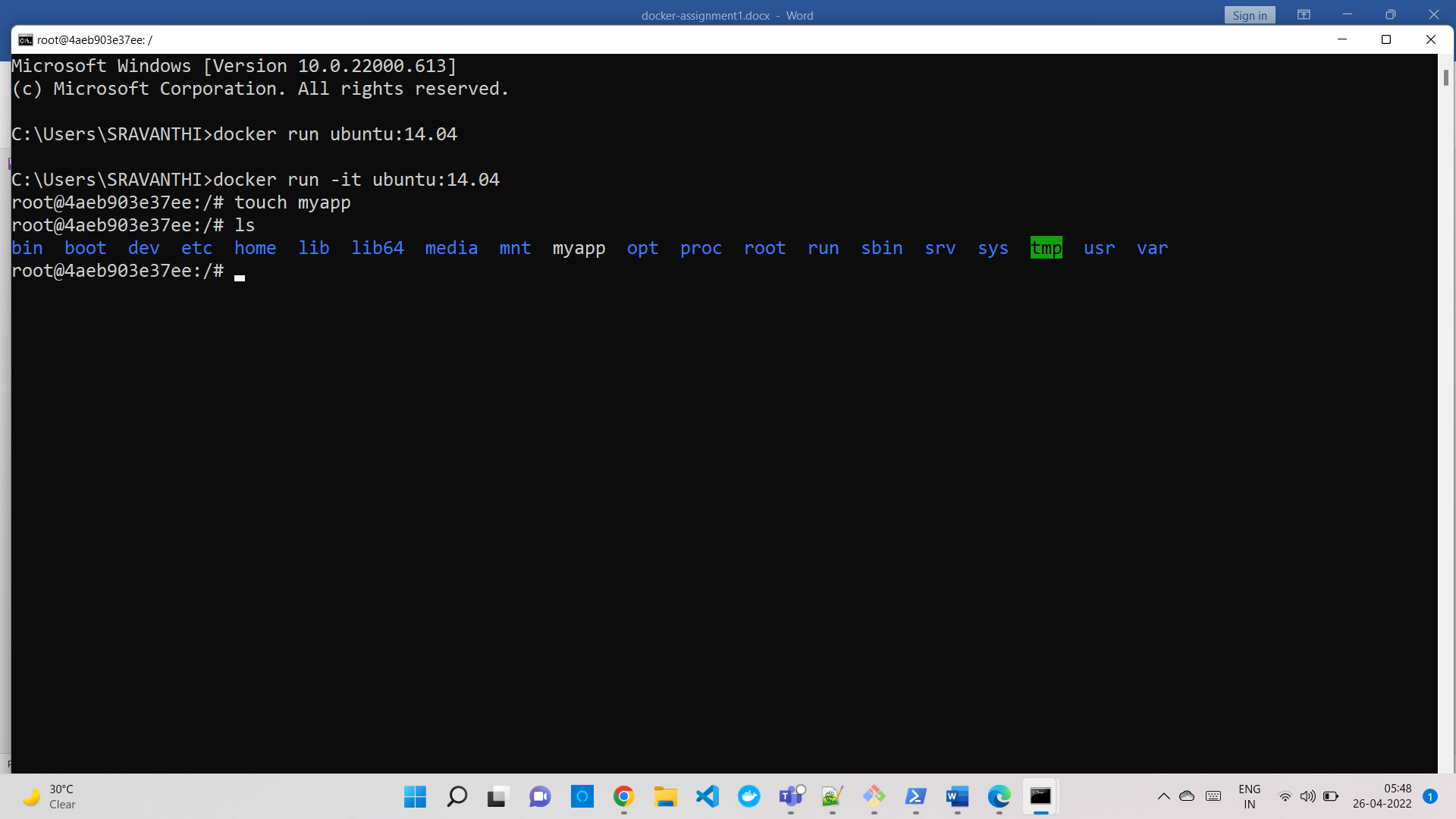
25. Check if the ubuntu container is running? If no, start one in attached mode to the terminal.



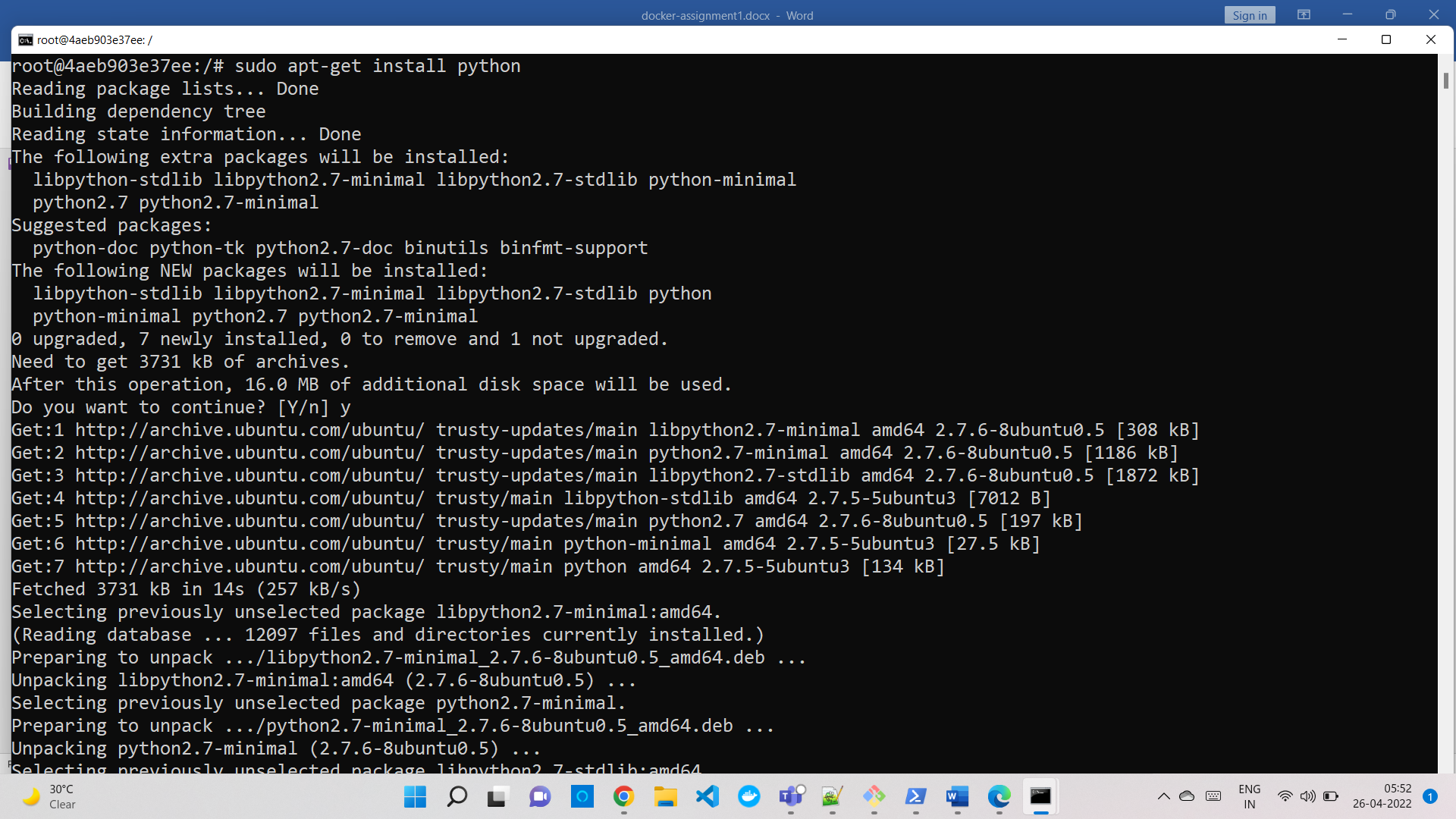
26. log in as a root user in the ubuntu container



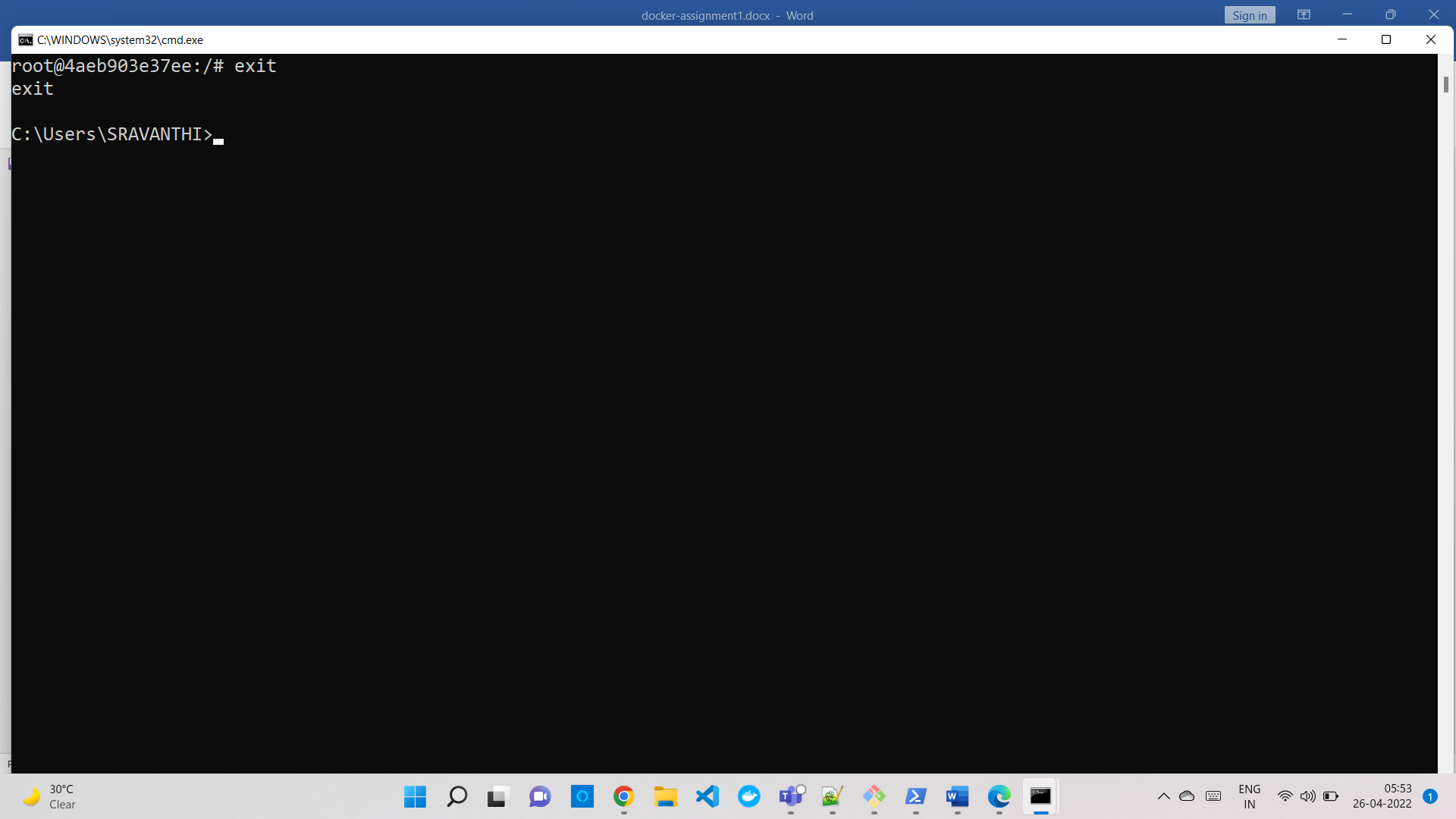
27. Create a file with any name in the root directory



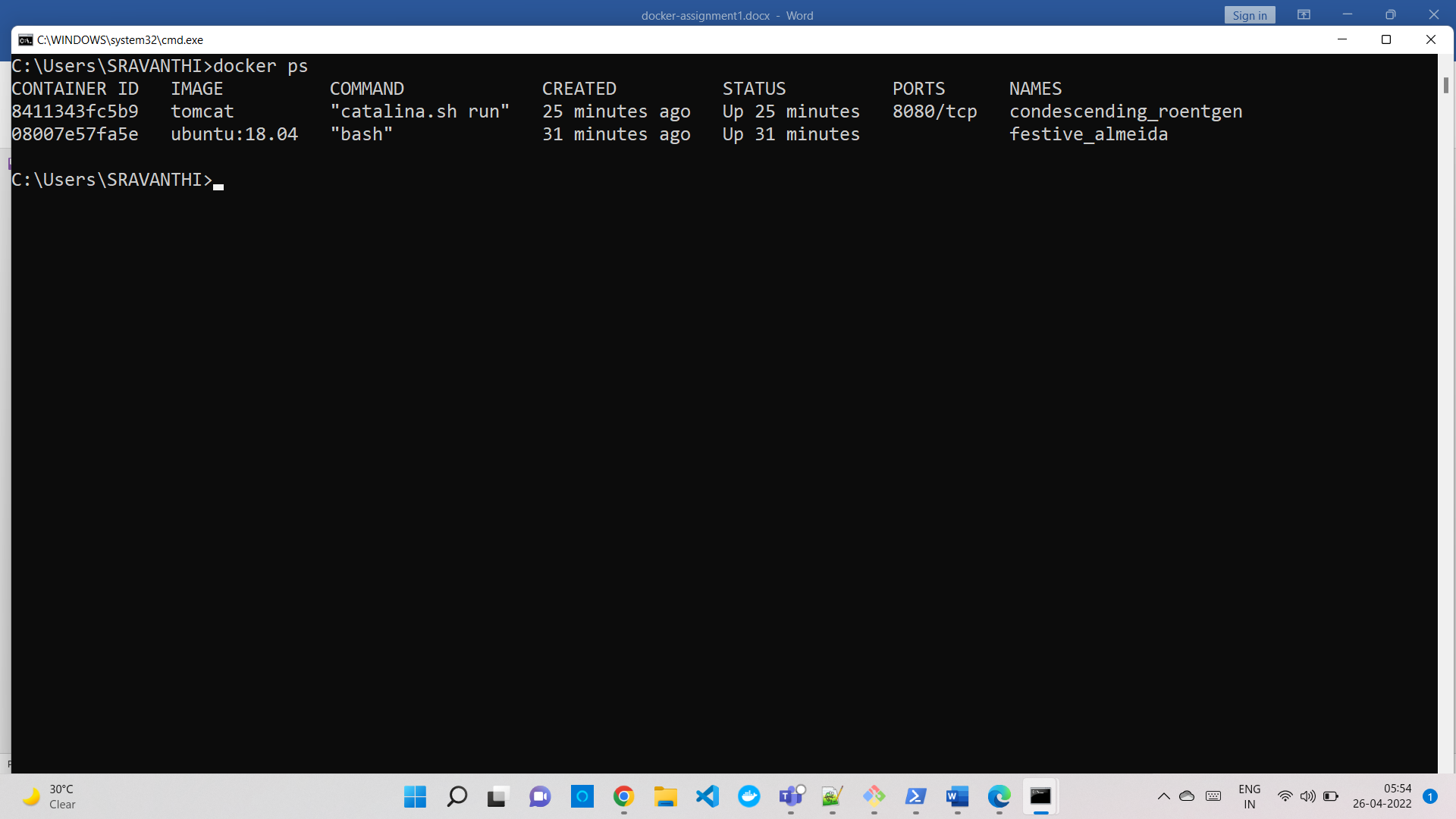
28. Install the software of your choice in the ubuntu container using "apt-get install"



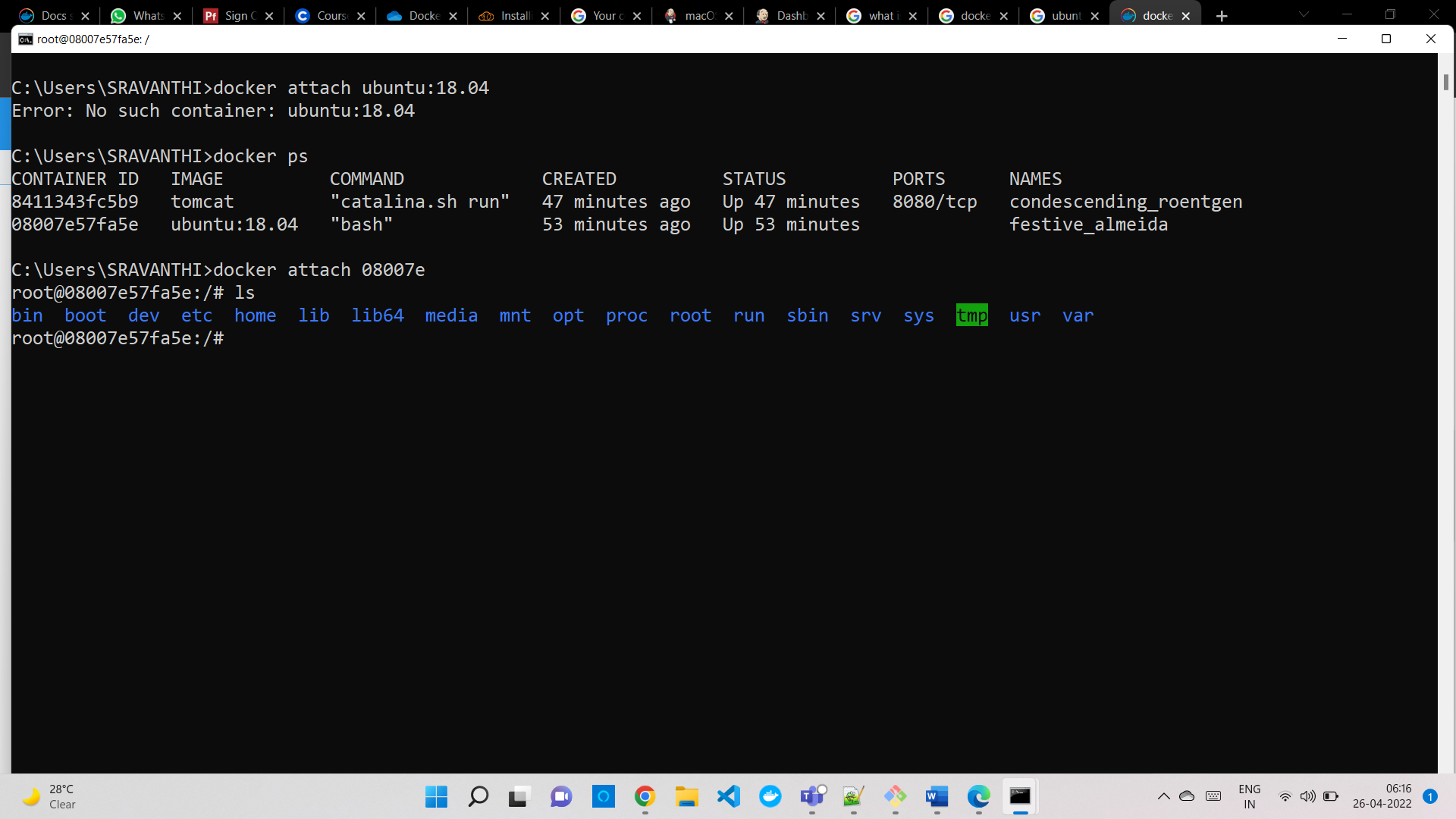
29. Now exit the ubuntu shell, are you back to your host machine, if not, come back to the host machine.



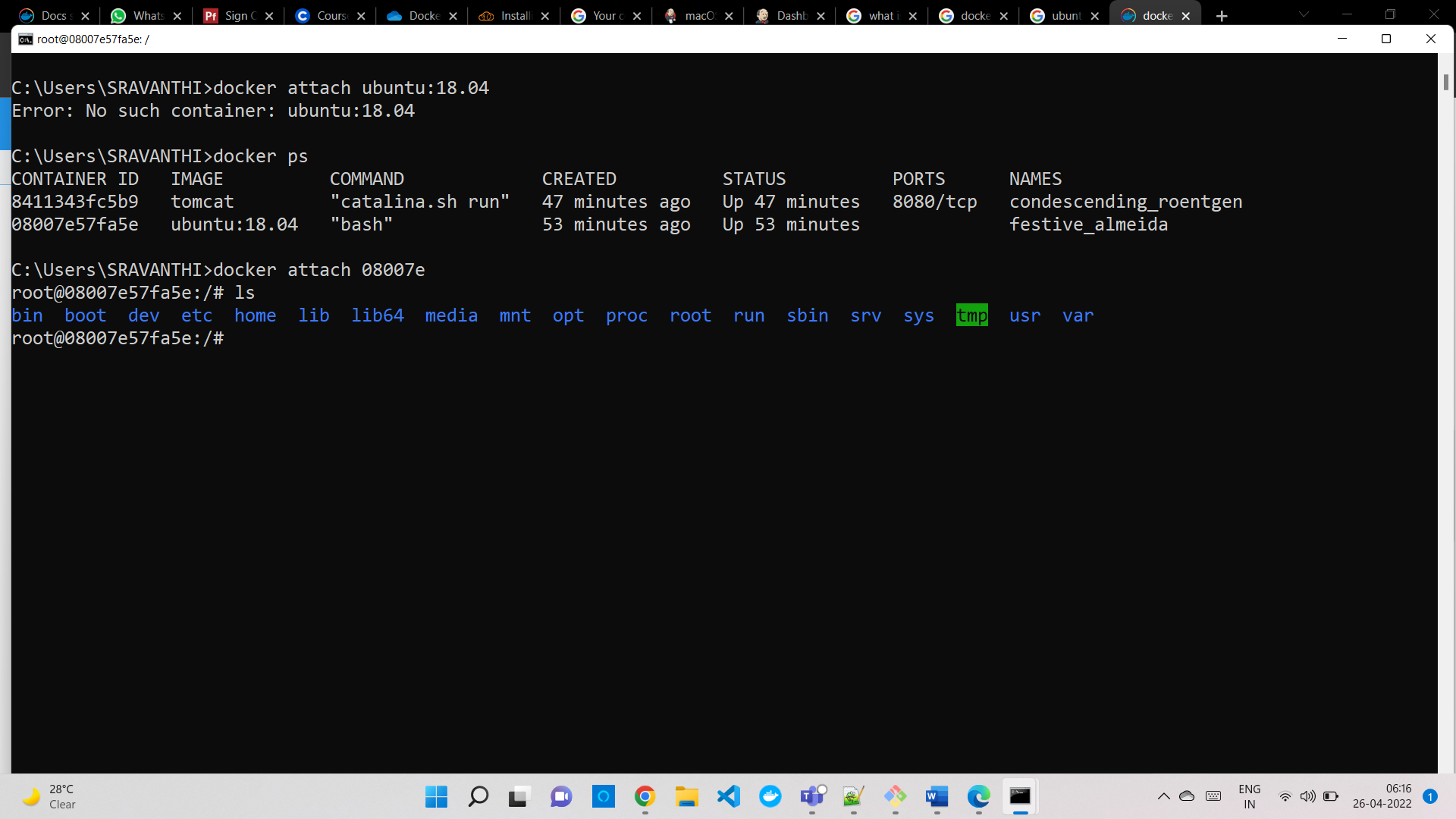
30. Check if the ubuntu container is running.



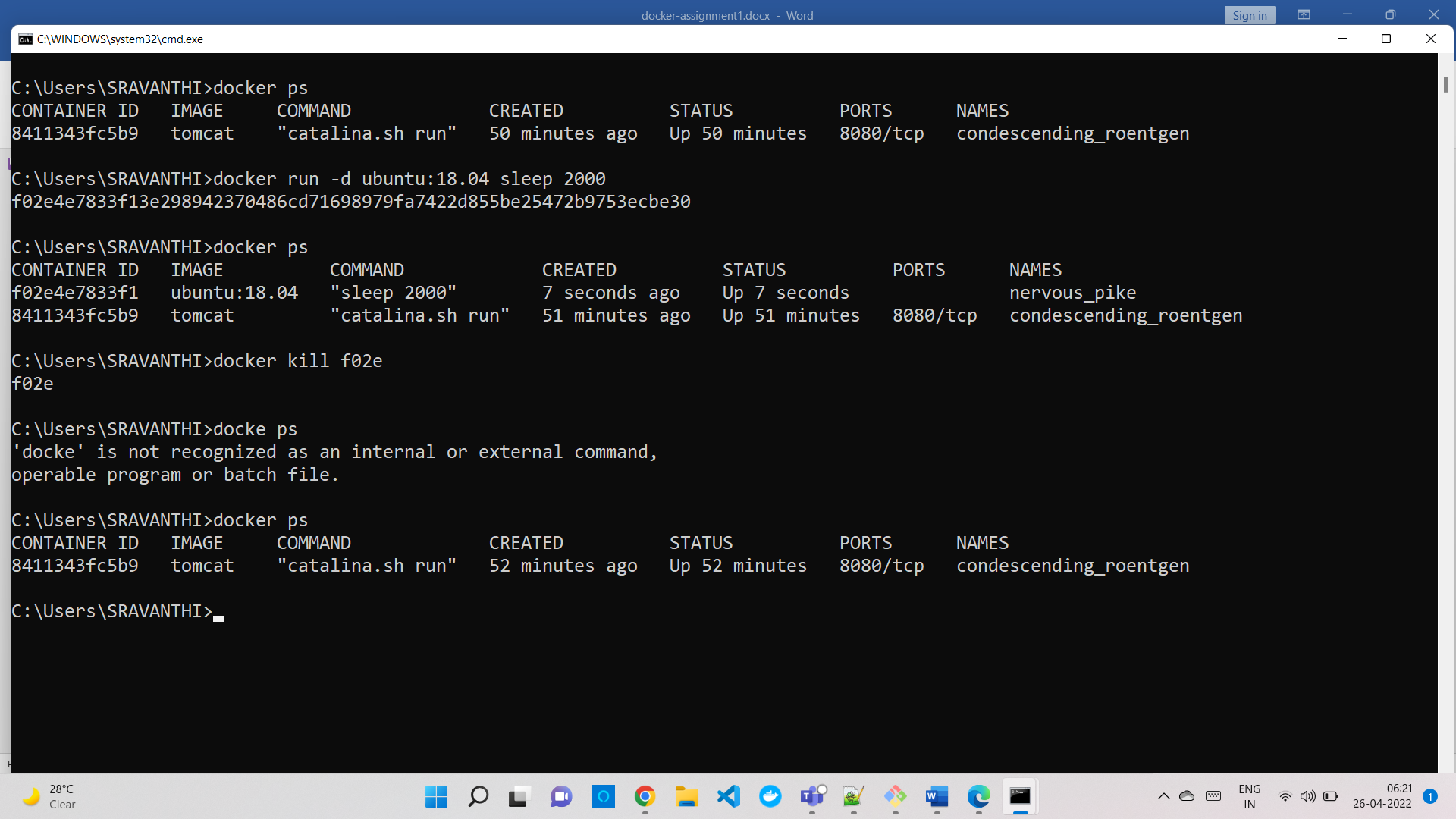
31. Create a new ubuntu container out of the same image as that previous container in attached mode.



32. log in as a root user



33. Check if you can see the file created in the previous container, you will not see the file as well as the software that you installed in the previous container. Now kill this Container.



34. Do you have the previous ubuntu container where you created the file and installed the software? If no repeat step 25 to 29.

Yes

35. Create an Image out of the existing container.

36. Now Create a Container out of this image and login into it to see if you can see the file and software installed by you in the previous container.

37. Do you have a running tomcat container? If yes, Stop it and kill all tomcat containers.

38. Create an index.html file with the following code in it:-

<h1>This is Tomcat Container</h1>

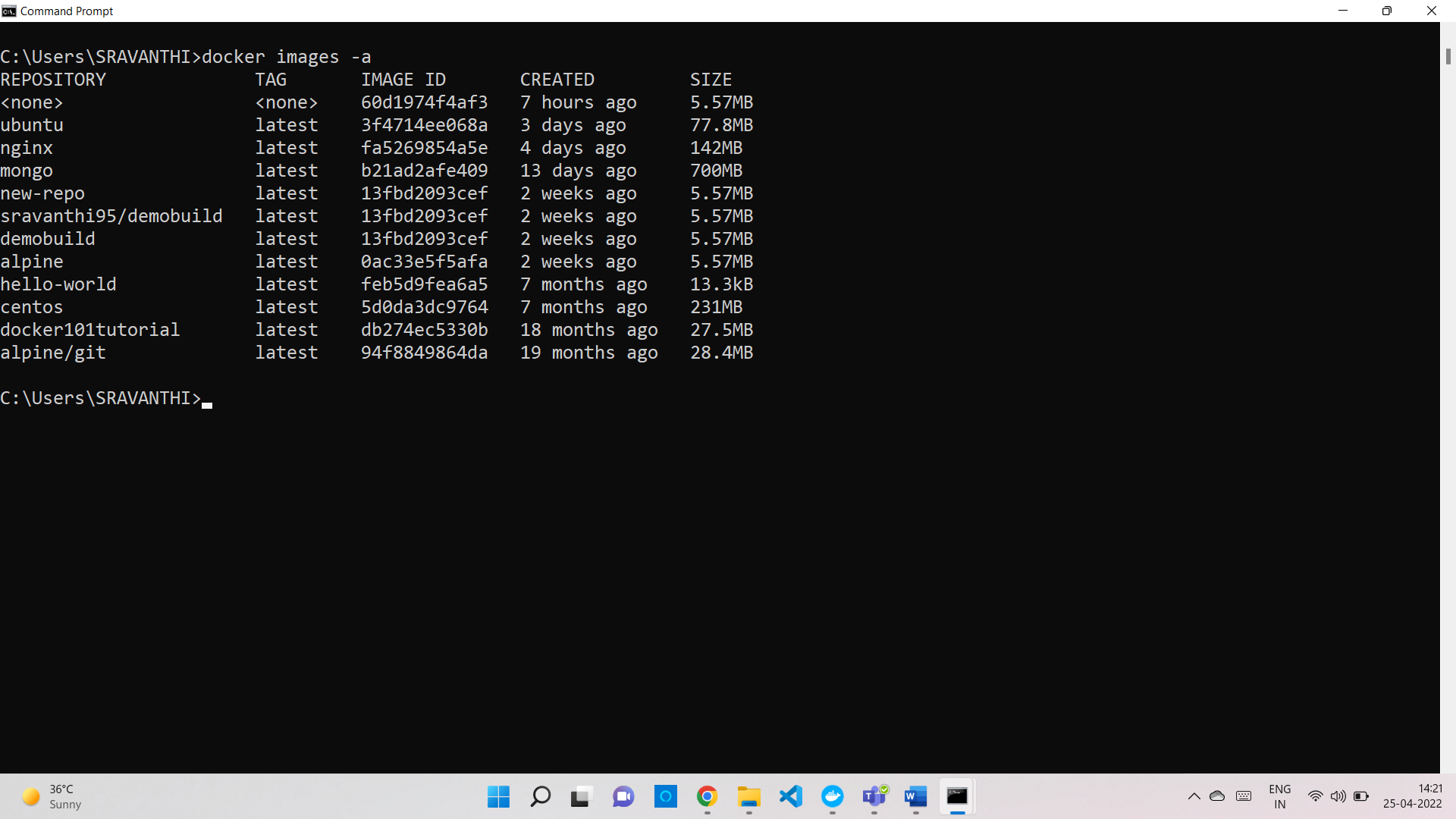
Now, Start a tomcat container in such a way that on hitting its URL for the home page it should show the above HTML page.

39. type below command:-

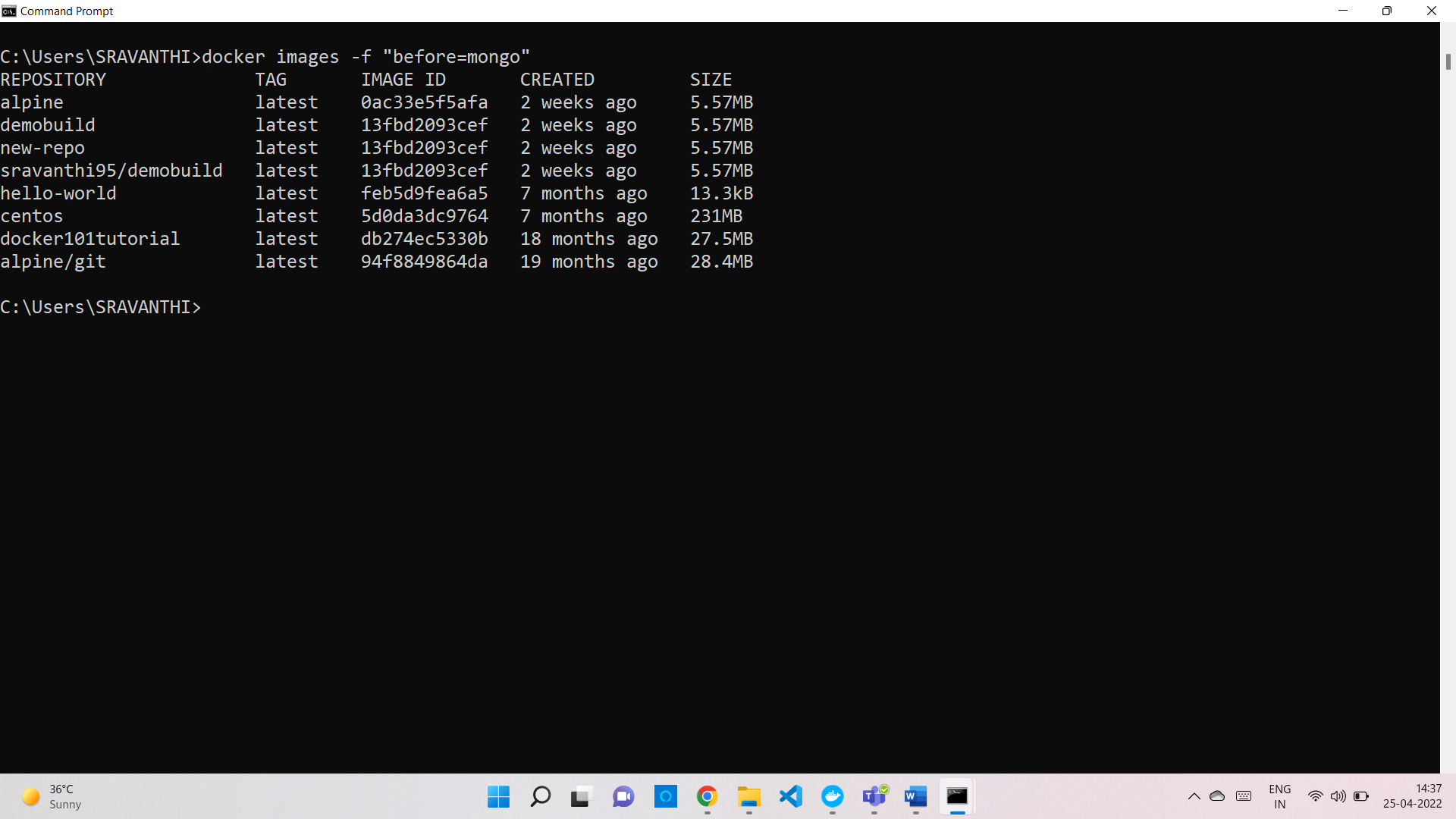
docker images --help

Now, try to run a command that proves the concept of the following three options:-

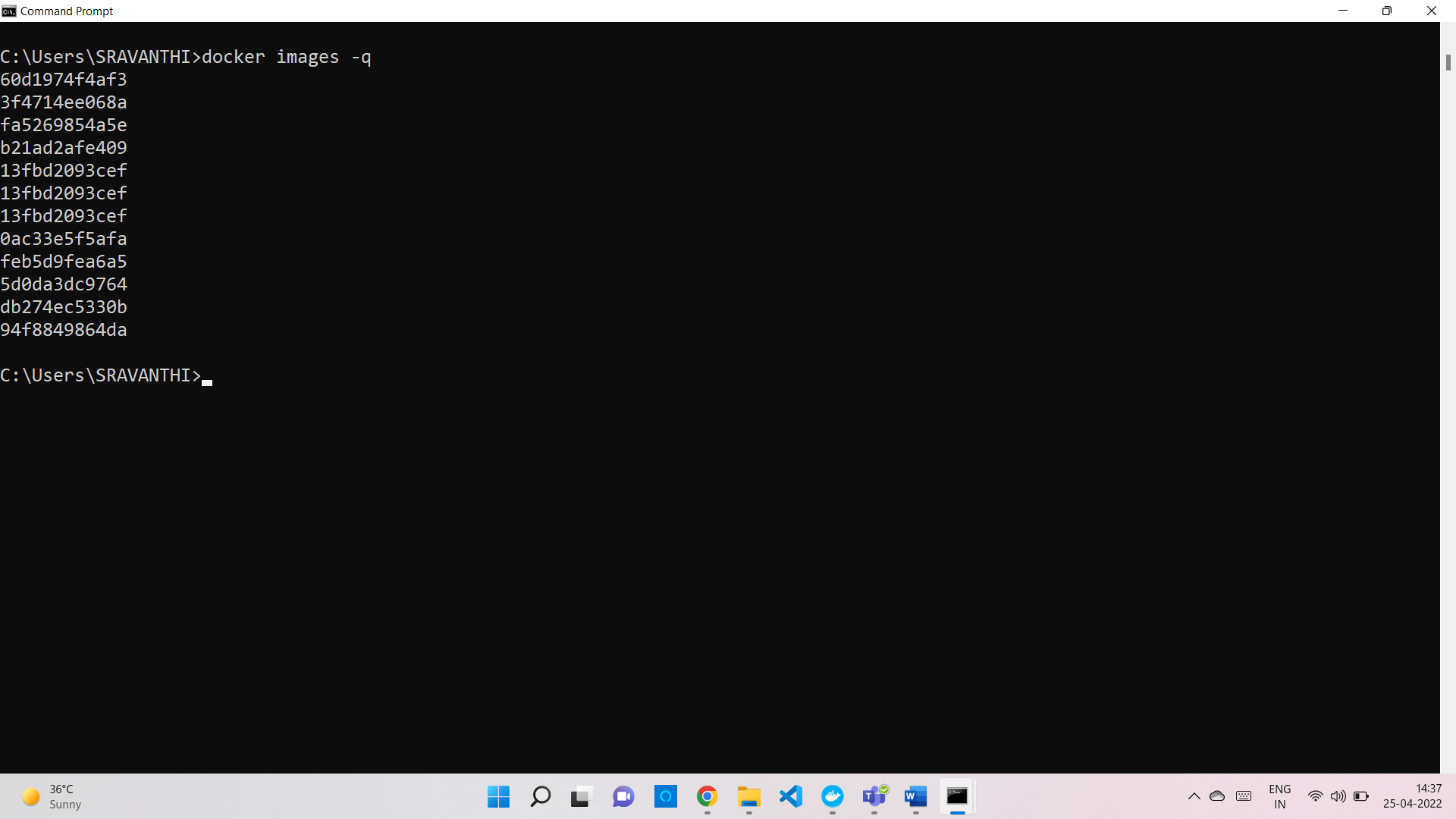
1. -a docker images -a (Gives the list of all images)



1. -f docker images -f “before = mongo”



1. -q docker images -q



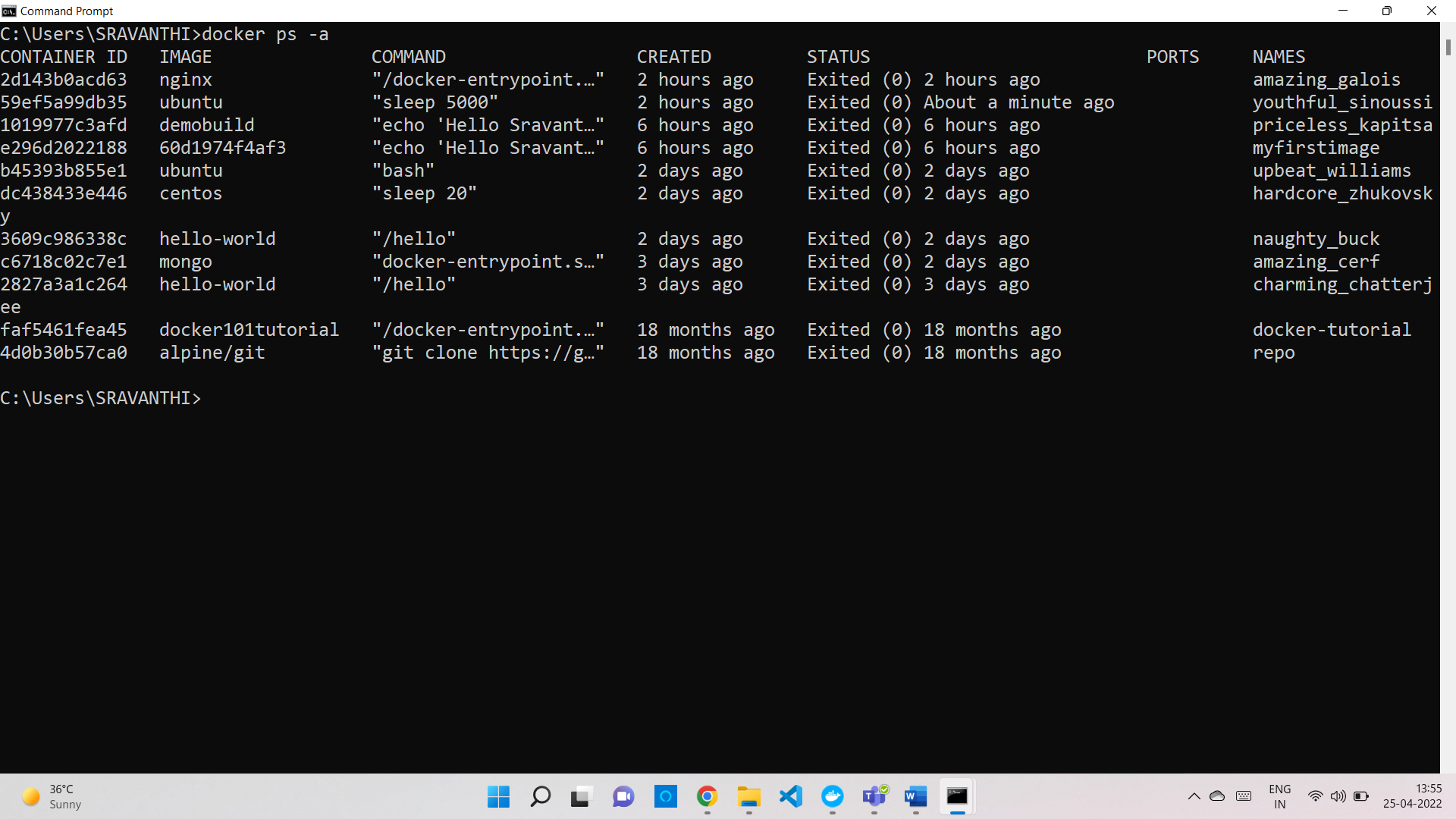
write at least 1 command using each option above and prove their concepts as described in the --help.

40. type below command:-

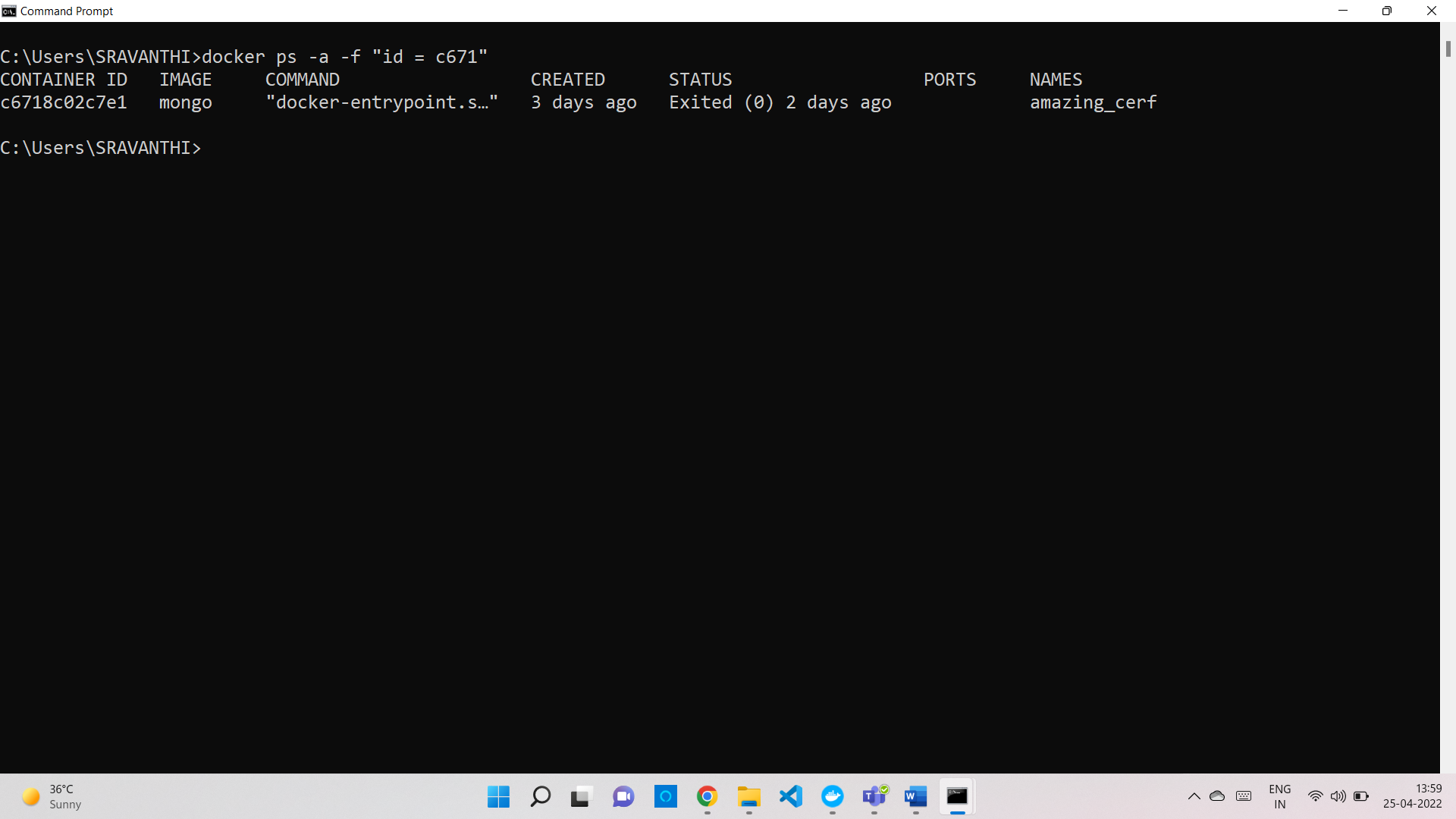
docker ps --help

Now, try to run a command that proves the concept of the following six options:-

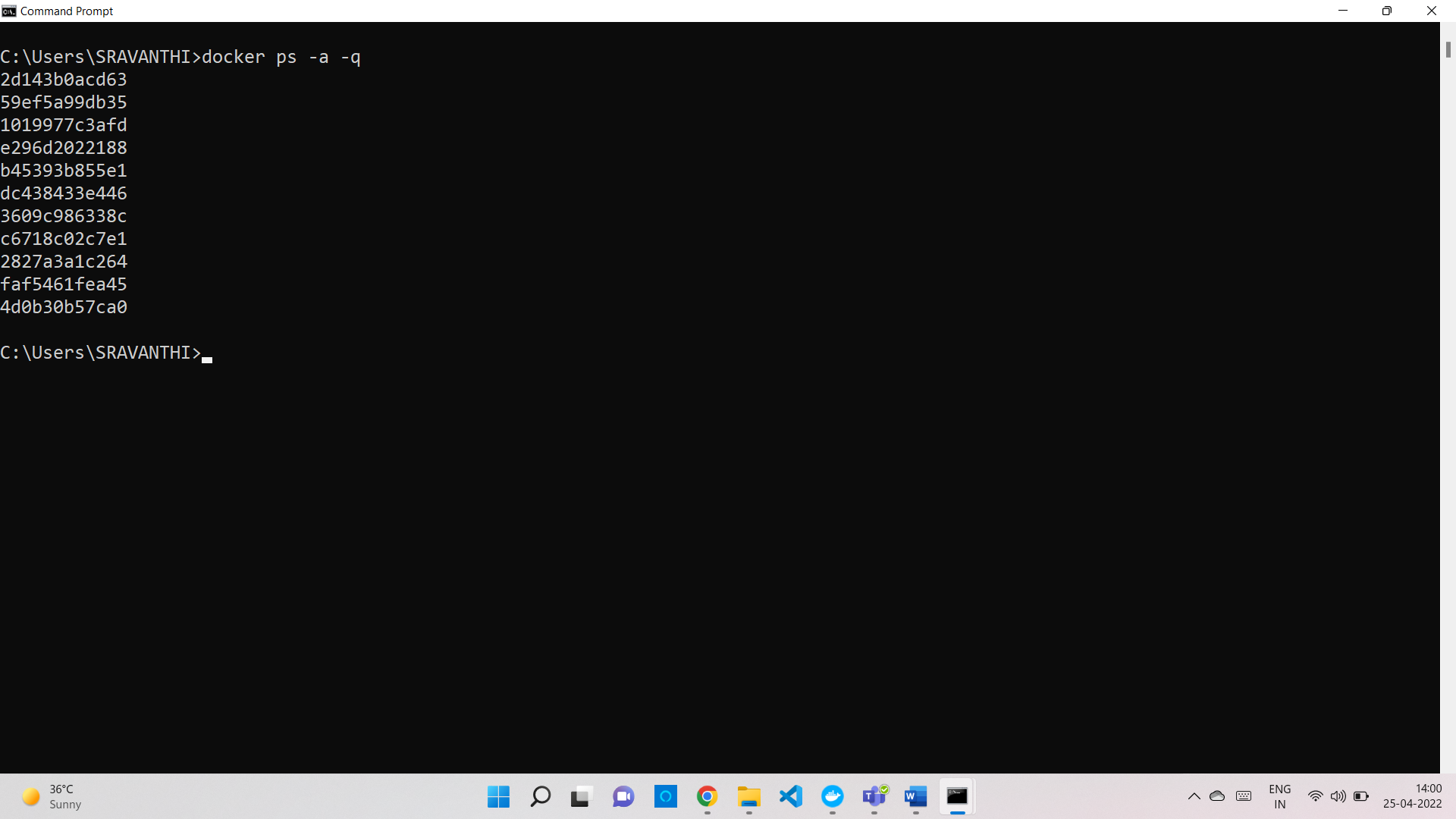
1. -a docker ps -a



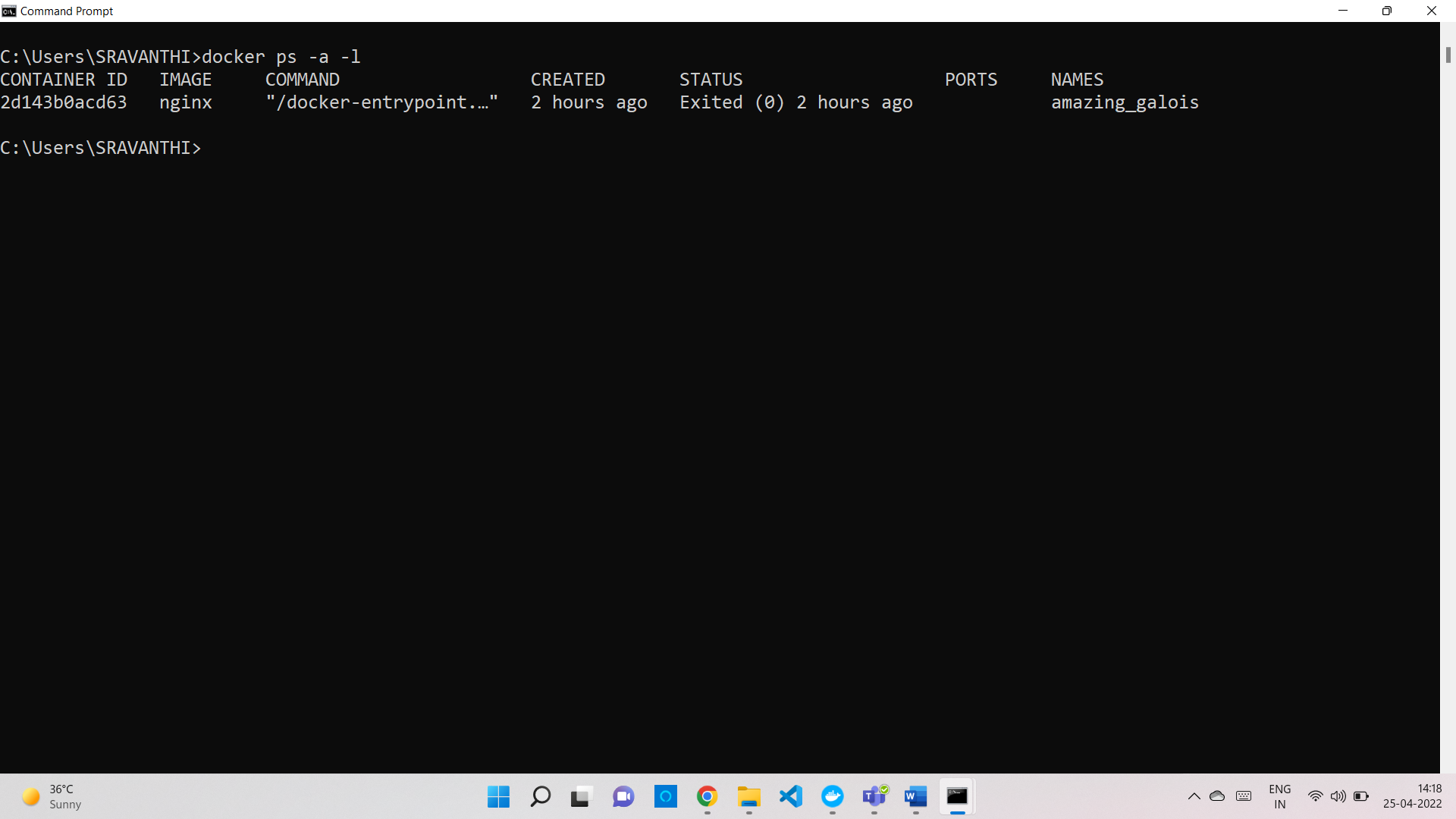
1. -f docker ps -a -f “id=c671”



1. -q docker ps -a -q (list out the all the container id’s)



1. -n
2. -l docker ps -a -l (Give the latest results)



6. -s docker ps -a -s

