Docker command

Docker command

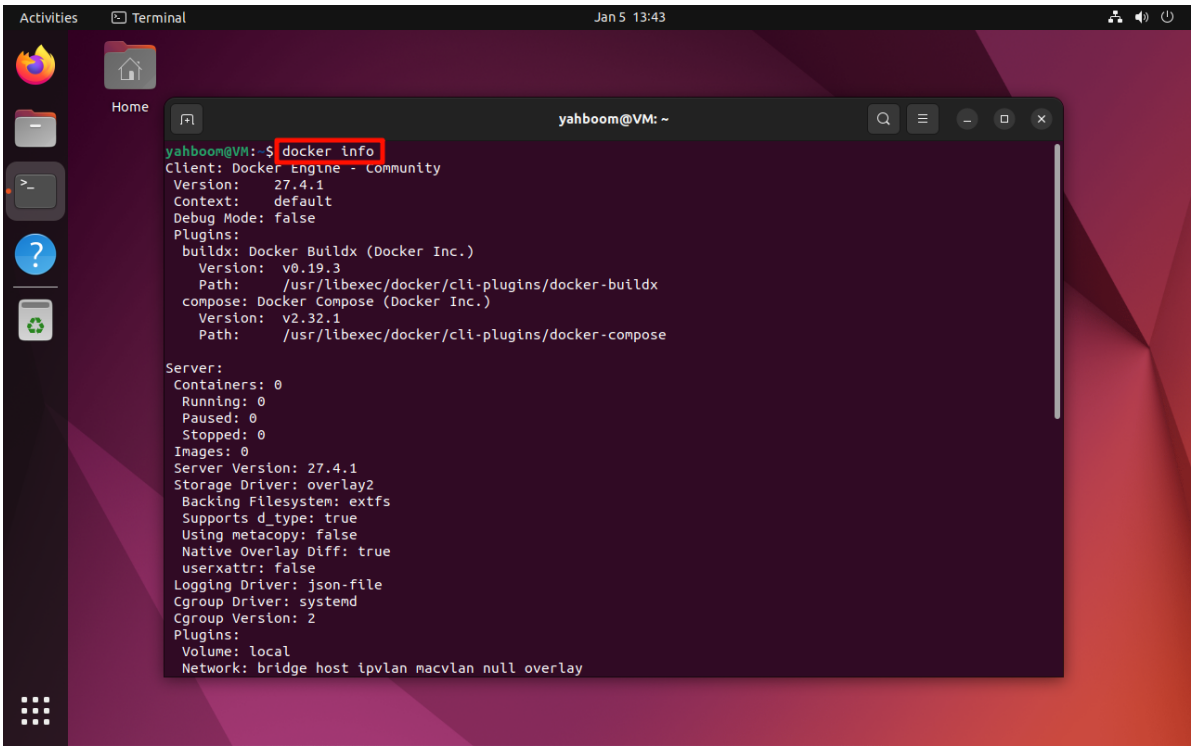
1. View detailed information
2. View the version number
3. Pull the image
4. Run the image
    4.1. View the running image
    4.2. View the running or stopped image
5. Clean up images
6. View local images
7. Delete images
8. Submit the image
9. Stop the image
10. Multiple terminals enter the same container

Docker engine includes Docker CLI, which provides command line tools for interacting with Docker daemon. This tutorial introduces the usage of common Docker commands.

1. View detailed information

> **docker info**

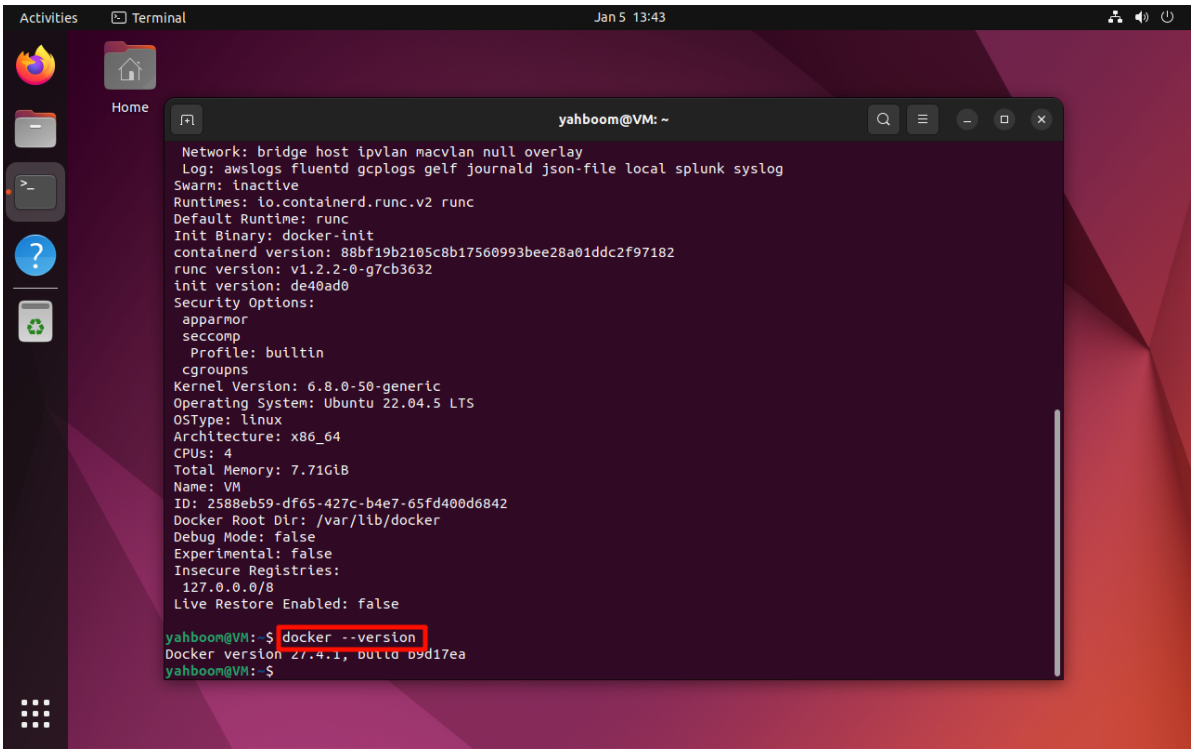
```
docker info
```



2. View the version number

> **docker --version**

```
docker --version
```



3. Pull the image

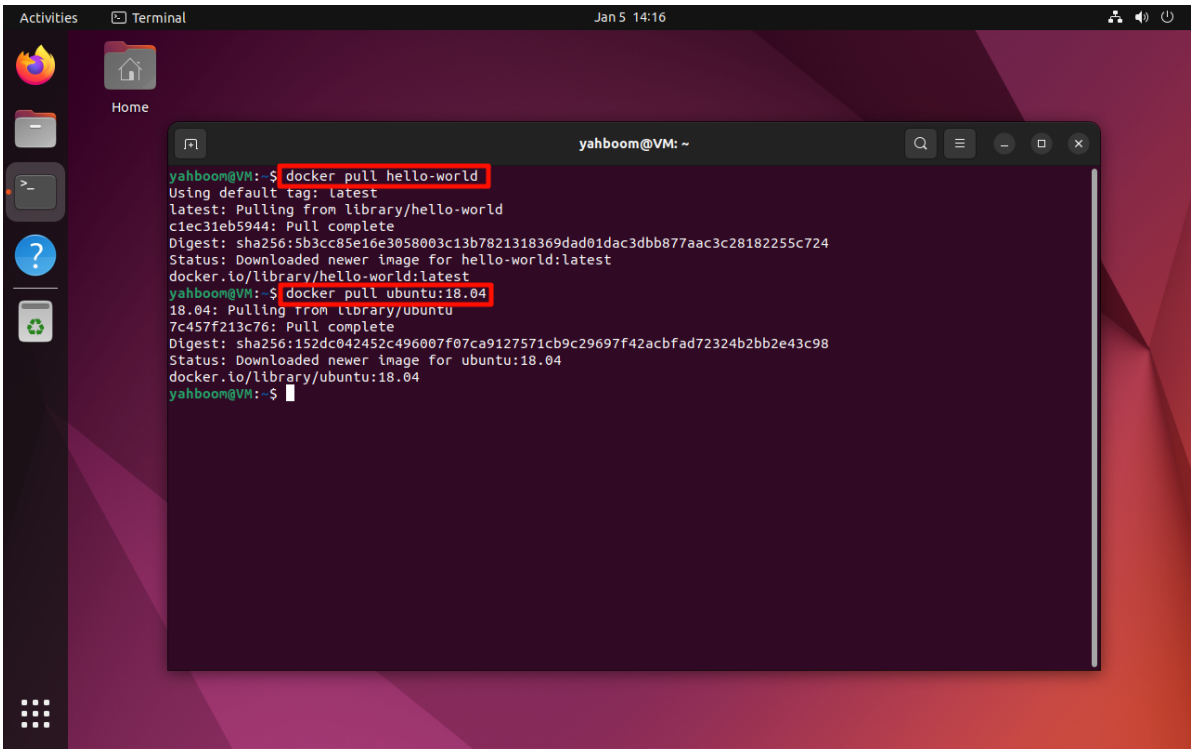
> **docker pull <image_name>**

If no tag is specified, the image with the `latest` tag will be pulled by default.

```
docker pull hello-world
```

> **docker pull <image_name>:**

```
docker pull ubuntu:18.04
```



4. Run the image

If there is no image to be run locally, docker will automatically pull the corresponding image.

> **docker run <image_name>**

```
docker run hello-world
```

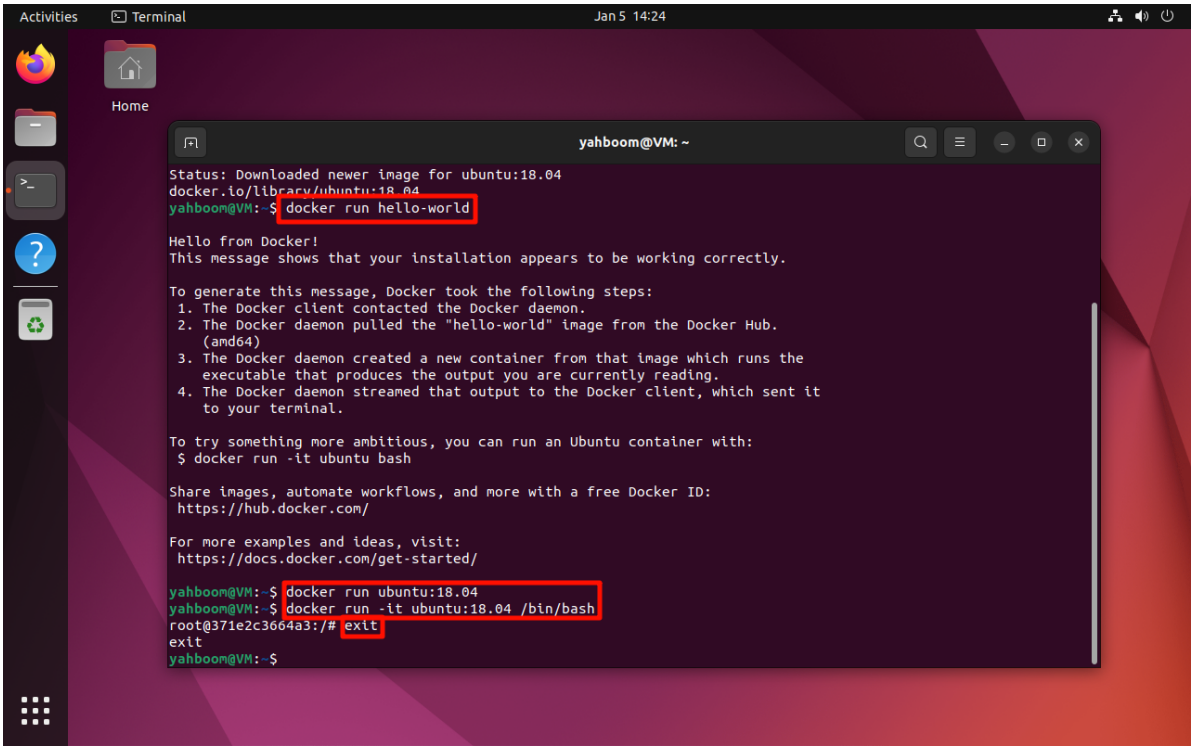
> **docker pull <image_name>:**

Start the specified container and exit:

```
docker run ubuntu:18.04
```

Start the specified container in interactive mode: Enter `exit` to exit

```
docker run -it ubuntu:18.04 /bin/bash
```



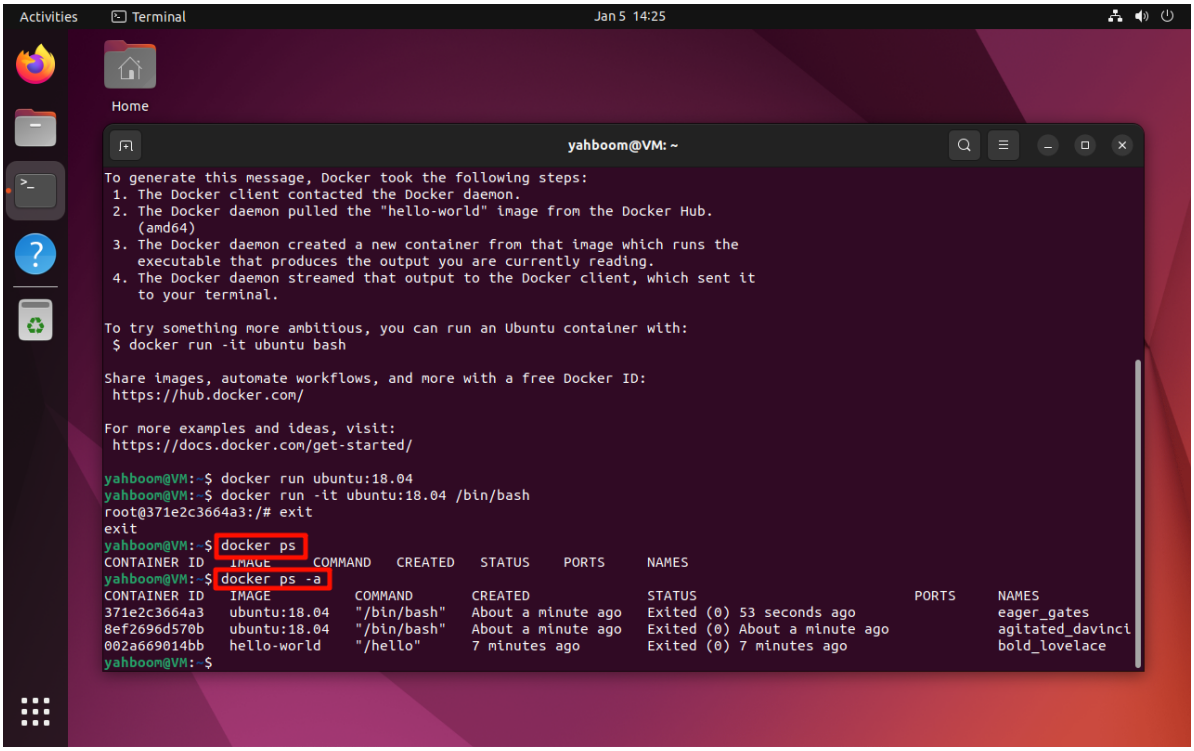
4.1. View the running image

> **docker ps**

```
docker ps
```

4.2. View the running or stopped image

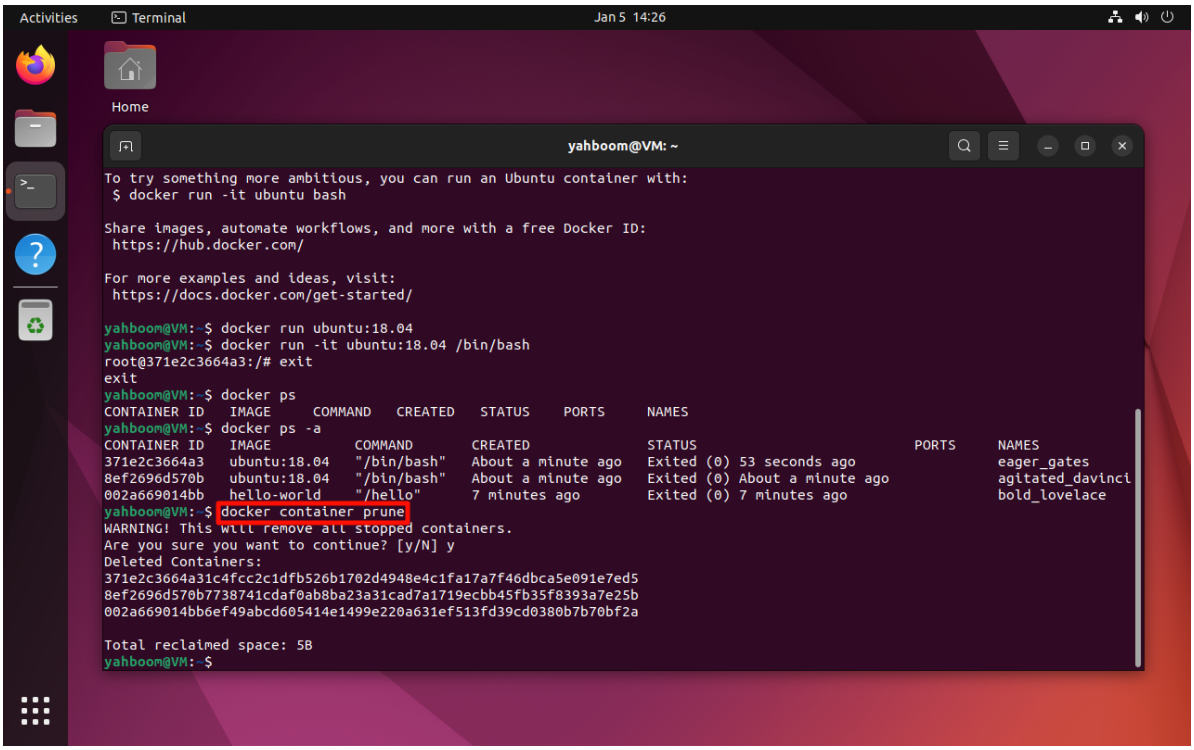
```
docker ps -a
```



5. Clean up images

> docker container prune

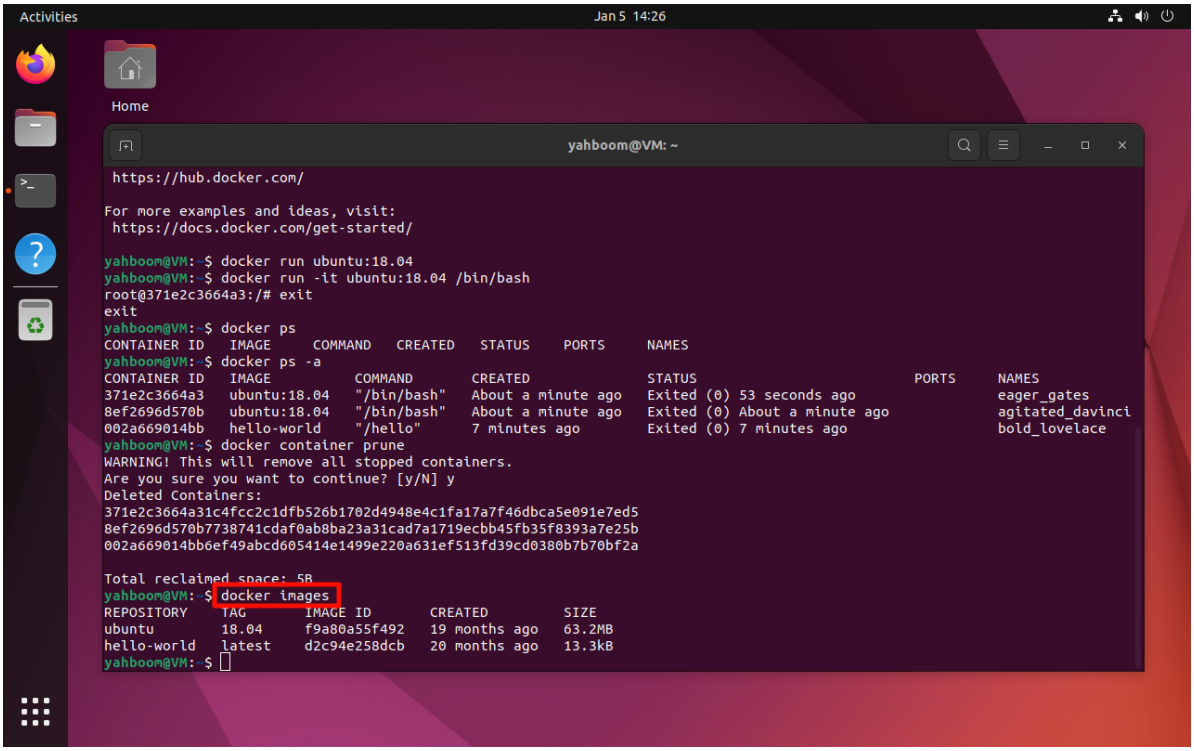
```
docker container prune
```



6. View local images

> docker images

```
docker images
```



7. Delete images

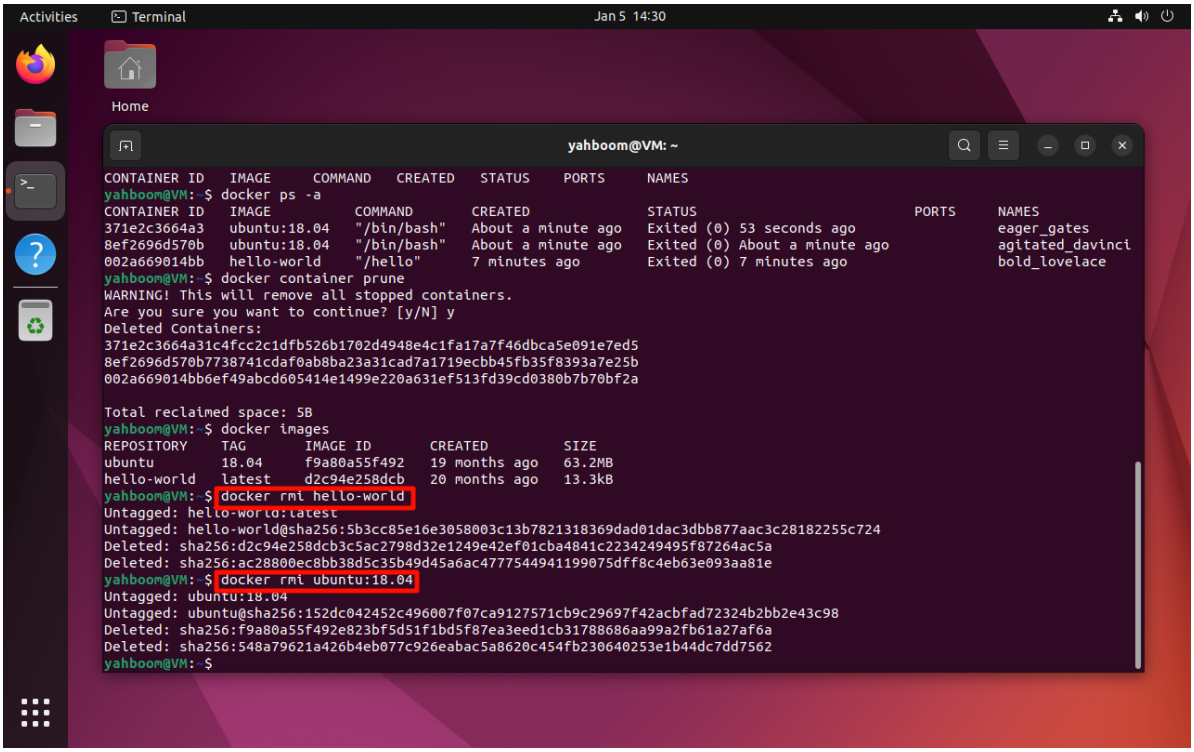
Note: The image to be deleted needs to be stopped and cleaned up

> **docker rmi <image_name>**

```
docker rmi hello-world
```

> **docker rmi <image_name>:**

```
docker rmi ubuntu:18.04
```



8. Submit the image

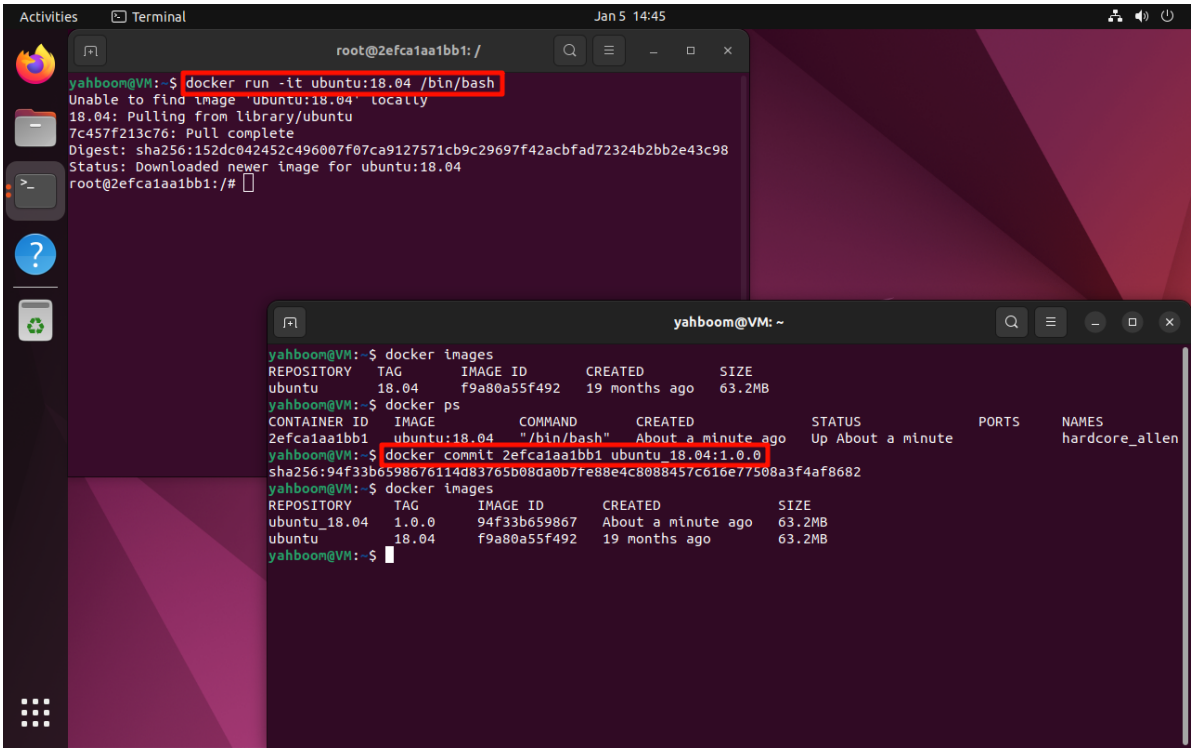
Pull and run the ubuntu:18.04 image in interactive mode

```
docker run -it ubuntu:18.04 /bin/bash
```

> **docker commit <image_name>:**

Note: Modify according to the actual `CONTAINER ID`

```
docker commit 2efca1aa1bb1 ubuntu_18.04:1.0.0
```



9. Stop the image

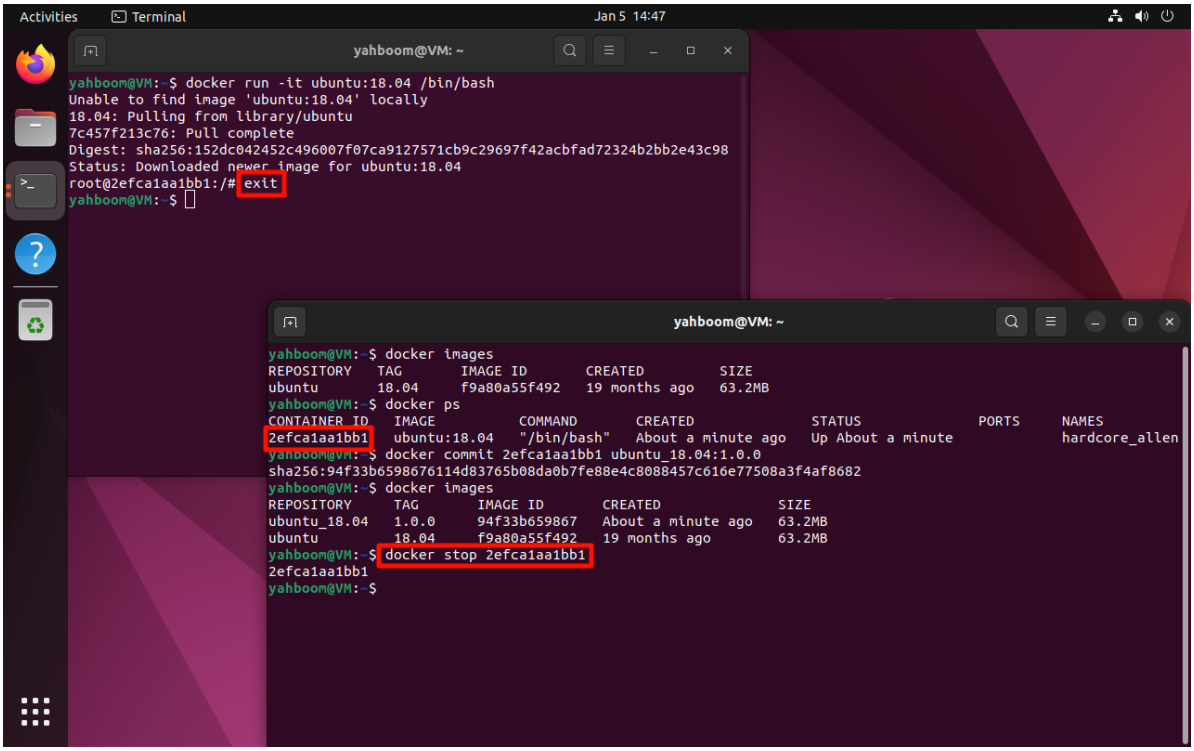
If you run the container in interactive mode and the terminal enters the container, you can enter `exit` in the container to stop the container;

If you close the container externally, you can use the `docker stop` command.

> **docker stop**

Note: Modify according to the actual `CONTAINER ID`

```
docker stop 2efca1aa1bb1
```



10. Multiple terminals enter the same container

Containers are isolated from each other. Directly using the command to run the image will start different containers; if you need to perform operations in the same container, you need to use the command to enter the same container.

Run the ubuntu:18.04 image in interactive mode:

```
docker run -it ubuntu:18.04 /bin/bash
```

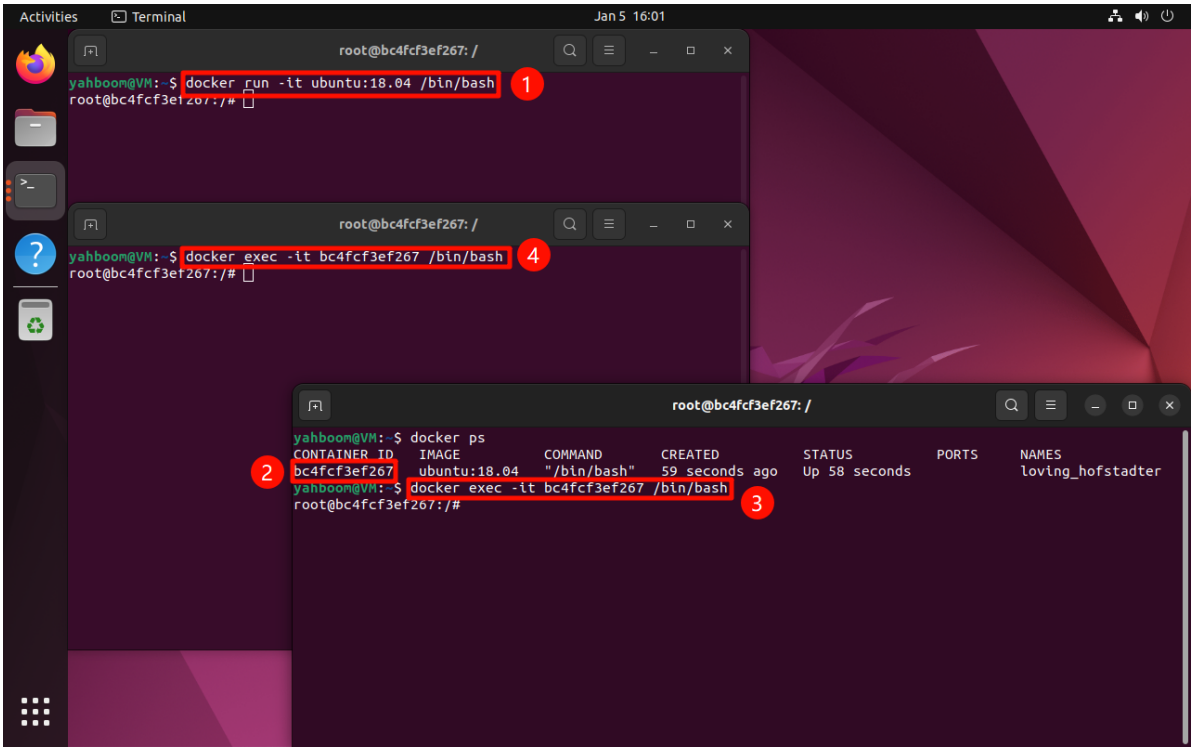
View the running container:

```
docker ps
```

> ### docker exec

Enter the running container in interactive mode: modify according to the actual `CONTAINER ID`

```
docker exec -it bc4fcf3ef267 /bin/bash
```


Activities
Terminal
Jan 5 16:01
root@bc4fcf3ef267: /
yahboom@VM:~$ docker run -it ubuntu:18.04 /bin/bash
root@bc4fcf3ef267:/#
1
root@bc4fcf3ef267: /
yahboom@VM:~$ docker exec -it bc4fcf3ef267 /bin/bash
root@bc4fcf3ef267:/#
4
root@bc4fcf3ef267: /
yahboom@VM:~$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
bc4fcf3ef267 ubuntu:18.04 "/bin/bash" 59 seconds ago Up 58 seconds loving_hofstadter
yahboom@VM:~$ docker exec -it bc4fcf3ef267 /bin/bash
root@bc4fcf3ef267:/#
2
3