Enter the robot's docker container

5.1. Related concepts

1. What is the docker host

```
The host is the server where we call commands to create containers using images.
This refers to the main control on our robot (jetson nano, etc.), and the host
mentioned below refers to this.
```

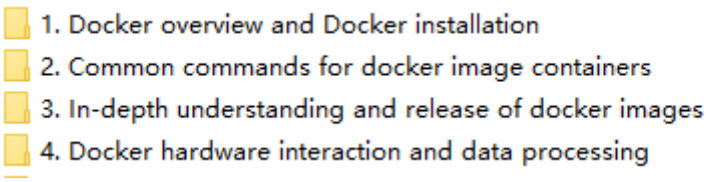
2. What is a GUI

```
GUI is a graphical user interface, which mainly refers to: the image window
displayed by opencv, the rviz interface, the rqt interface, etc.
```

3. What is the robot's docker container

```
The robot here is the Jetcobot robotic arm, and the docker container refers to
the robot container that has been configured with various development dependent
environments.
```

4. Before operating this chapter tutorial, please make sure that you have mastered the knowledge of the following chapters, otherwise you may find it difficult to learn. In this case, please check the following pre-knowledge content repeatedly, and you will feel very relaxed after mastering it. Come on, you are the best!



5.2, Generate Docker container

To facilitate the generation of Docker container, edit the command to generate Docker container into a shell script. The factory image will come with a startup script in the host system, named [new_jetcobot_docker.sh].

If it is a self-edited script, you need to modify [VERSION] to the latest version according to the actual situation.

```
#!/bin/bash
xhost +

VERSION='3.1.1'

docker run -it \
--privileged \
--net=host \
--ipc=bridge \
--ipc=host \
--pid=host \
```

```
--env="DISPLAY" \
--env="QT_X11_NO_MITSHM=1" \
-v /tmp/.X11-unix:/tmp/.X11-unix \
-v /dev/bus/usb:/dev/bus/usb \
-v ~/temp:/root/temp \
yahboomtechnology/jetcobot_noetic:$VERSION /bin/bash
```

Run the following command to generate a new version of the docker container.

```
sh new_jetcobot_docker.sh
```

Then open another terminal and enter the following command to view the ID and name of the currently started docker container

```
docker ps
```

5.3. Operate Docker container

To facilitate the operation of Docker containers, edit the commands for operating Docker into shell scripts. The factory image will come with a startup script in the host system, named [start_docker.sh].

If it is a self-edited script, you need to modify [DOCKER_ID] to the container ID number or container name that needs to be started according to the actual situation.

```
#!/bin/bash

xhost +

# Modified based on the CONTAINER ID or NAMES
DOCKER_ID="zealous_maxwell"

INPUT_PARM=$1
CHECK_ENABLE="enable"
CHECK_START="start"
CHECK_DISABLE="disable"
CHECK_STOP="stop"
CHECK_JUPYTERLAB="lab"
CHECK_RESTART="restart"

if [ $# = 0 ]; then
    # echo "count:" $#
    echo "exec docker:"
    docker exec -it $DOCKER_ID /bin/bash -c "cd /root/ && /bin/bash"
fi

if [ "$INPUT_PARM" = "$CHECK_START" ] || [ "$INPUT_PARM" = "$CHECK_ENABLE" ];
then
    echo "start docker:"
    docker start $DOCKER_ID
    docker exec -it $DOCKER_ID /bin/bash -c "cd /root/ && /bin/bash"
fi

if [ "$INPUT_PARM" = "$CHECK_STOP" ] || [ "$INPUT_PARM" = "$CHECK_DISABLE" ];
then
```

```
    echo "stop docker:"
    docker stop $DOCKER_ID
fi

if [ "$INPUT_PARM" = "$CHECK_JUPYTERLAB" ]; then
    echo "start jupyter:"
    docker start $DOCKER_ID
    docker exec $DOCKER_ID /bin/bash -ic "jupyter lab &"
    # docker exec $DOCKER_ID /bin/bash -ic "/root/start_jupyterlab.sh"
fi

if [ "$INPUT_PARM" = "$CHECK_RESTART" ]; then
    echo "restart docker:"
    # docker restart $DOCKER_ID
    docker stop $DOCKER_ID
    sleep 2
    docker start $DOCKER_ID
    docker exec $DOCKER_ID /bin/bash -ic "jupyter lab &"
    # docker exec $DOCKER_ID /bin/bash -ic "/root/start_jupyterlab.sh"
fi
```

Start docker container

```
sh start_docker.sh start
# or this command
sh start_docker.sh enable
```

Start and open the jupyterlab service in the docker container

```
sh start_docker.sh lab
```

Enter the docker container

```
sh start_docker.sh
```

Shut down the docker container

```
sh start_docker.sh stop
# or this command
sh start_docker.sh disable
```

Restart the docker container

```
sh start_docker.sh restart
```

5.4, Summary

Without changing the docker image version, you only need to use the [new_jetcobot_docker.sh] script once to generate a container, and then update the generated container ID in the [start_docker.sh] file [DOCKER_ID]. After that, you only need to use the [start_docker.sh] script to start and enter the script. The advantage of this is that it avoids repeated container generation, and the modified data in the container can be saved. After restarting the host, use the [start_docker.sh] script again to enter the container, and the modified data can be saved.