roscore
执行 ros 核心的命令
启动 ros
ros master ros 参数服务器 rosout 日志节点
有节点 new master 之后端口是 11311 之后就
还有相应的 ros 输出
auto-starting new master
process[master]: started with pid [3675]
ROS_MASTER_URI=http://qinghuan-System-Product-Name:11311/

setting /run_id to b9ff4e38-d702-11ee-8763-092887197d13
process[rosout-1]: started with pid [3685]
started core service [/rosout]

ros 的参数服务器

rosrun 启动节点
rosrun 功能包 节点名
rosrun hello_world hello_vscode
hello_world 是我们的功能包 hello_vscode 是报名

roslaunch
执行 多个 node 的 launch
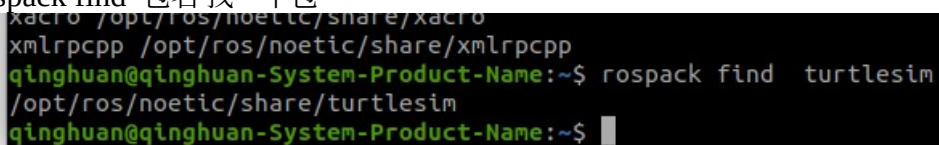roslaunch 功能包名 。laucnch 节点名

sudo apt install 安装功能包

sudo apt purge xxx 写在功能包

rospack list 列出所有的功能包

rospack find 包名 找一个包

roscd 进入功能包

roslis 包名 列出包下的文件

rosls turtlesim

功能包的文件

搜索可以下载的功能包
apt search ros-noetic-*
搜索 ros noetic 版本的功能包

| gmapping 的包
sudo apt install 把报名复制下来

修改功能包的文件
需要安装 vim
rosed turtlesim Color.msg
退出 是 ctrl shift :
ctrl 键 + shift 键 + : 冒号键

sudo apt install ros-noetic 功能包

rosrun rqt_graph rqt_graph
这个是我们在运行一个程序的节点
之后通过 ctrl alt t 一个新的终端
rosrun rqt_graph rqt_graph
功能包名是 rqt_graph
节点名是 ——rqt_graph

Node Graph

Nodes only / /

Group: 2 Namespaces Actions tf Images
Highlight Fit

Hide: Dead sinks Leaf topics Debug tf Unreachable Params

/turtle_key —— /turtle1/cmd_vel ——> /turtle_GUI

ne: ~ 80x24

aph rqt_graph

功能的文件

搜索可以下载的功能包
apt search ros-noetic-*
搜索 ros noetic 版本的功能包

| gmapping 的包
sudo apt install 把报名复制下来

qinghuan@qinghuan-System-Product-Name:~$ apt search ros-noetic-* | grep -i gmap
ping

WARNING

cob_n
e.
This
ros-noe
This
ros-noe
debug
ros-noe
The c
LAN-org
ros-noe
debug
ros-noe
slam_
ies.
qinghua

修改功能包
需要安装 vi
rosed turtles
退出是 ctrl
ctrl 键 + shif

sudo apt ins

start_turtle.launch - demo02_ws - Visual Studio Code

文件 编辑 选择 查看 转到 运行 终端 帮助

资源管理器   {} c_cpp_properties.json   hello1.py   hello_vscode_c.cpp   start_turtle.l

DEMO02_WS   src > hello_vscode > launch > start_turtle.launch
.vscode   1 <launch>
  c_cpp_properties.json 2
  {} launch.json   3 <!--执行的节点 pkg报名 type节点名字 name:自定义名字-->
  {} settings.json   4 <!--node 表示节点 pkg是包名 type是节点名字 是c中136 的参数0 name是
  {} tasks.json   5 <node pkg="turtlesim" type="turtlesim_node"   name="turtle
> build   6 <node pkg="turtlesim" type="turtle_teleop_key" name="turtle
> devel   7 <node pkg="hello_vscode" type="hello_vscode_c" name="hello"
∨ src   8 </launch>
  ∨ hello_vscode
    > include
    ∨ launch
       start_turtle.launch
    > scripts
      hello_vscode_p.py
      hello1.py
    ∨ src
      hello_vscode_c.cpp
    M CMakeLists.txt
    package.xml
  M CMakeLists.txt   问题 输出 调试控制台 终端 端口 注释   + ∨ python3
  .catkin_workspace
                             ROS_MASTER_URI=http://localhost:11311

                             setting /run_id to a480cf70-d70a-11ee-8763-092887197d13
                             process[rosout-1]: started with pid [6870]
                             started core service [/rosout]
                             process[turtle_GUI-2]: started with pid [6873]
                             process[turtle_key-3]: started with pid [6878]
                             process[hello-4]: started with pid [6879]
                             [INFO] [1709215120.791552572]: hahaha
                             [INFO] [1709215120.791574491]: 你好 scode
> 大纲   [hello-4] process has finished cleanly
> 时间线   log file: /home/qinghuan/.ros/log/a480cf70-d70a-11ee-8763-092887197d13/h

ROS1.noetic ⊗ 0 △ 0   ⌘ 0   行 1, 列 1 (已选择384) 空格: 4 UTF-8 LF XML Tabnine: Sig