

6、ROS2节点

1、节点简介

在通信时，不论采用何种方式，通信对象的构建都依赖于节点(Node)，在ROS2中，一般情况下每个节点都对应某一单一的功能模块(例如：雷达驱动节点可能负责发布雷达消息，摄像头驱动节点可能负责发布图像消息)。一个完整的机器人系统可能由许多协同工作的节点组成，ROS2中的单个可执行文件(C++程序或Python程序)可以包含一个或多个节点。

2、创建节点流程

1. 创建程序文件
2. 导入相关ros库
3. 编写节点功能
4. 编写配置文件
5. 编译运行

3、Hello World节点案例

这里以python功能包为例讲解

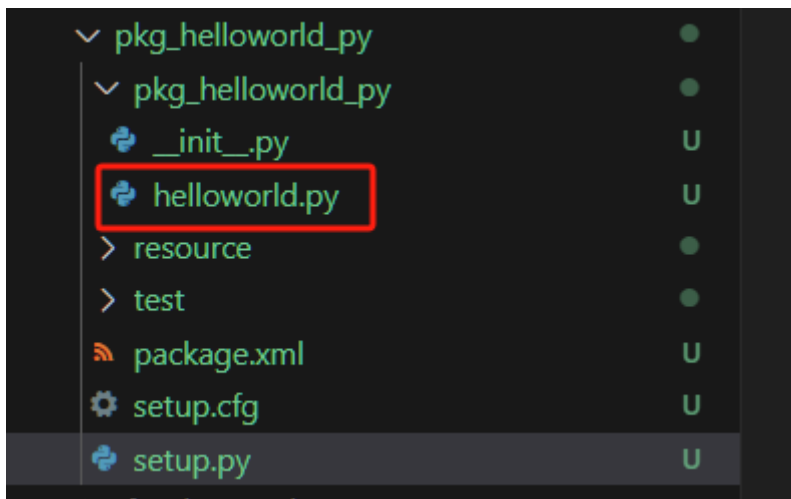
3.1、创建python功能包

- workspace替换成实际的工作空间路径

```
cd workspace/src
ros2 pkg create pkg_helloworld_py --build-type ament_python --dependencies rclpy
--node-name helloworld
```

3.2、编写代码

执行上面命令后会创建pkg_helloworld_py，同时会创建helloworld.py文件来编写节点：



删除原本 helloworld.py 的代码，编写如下代码：

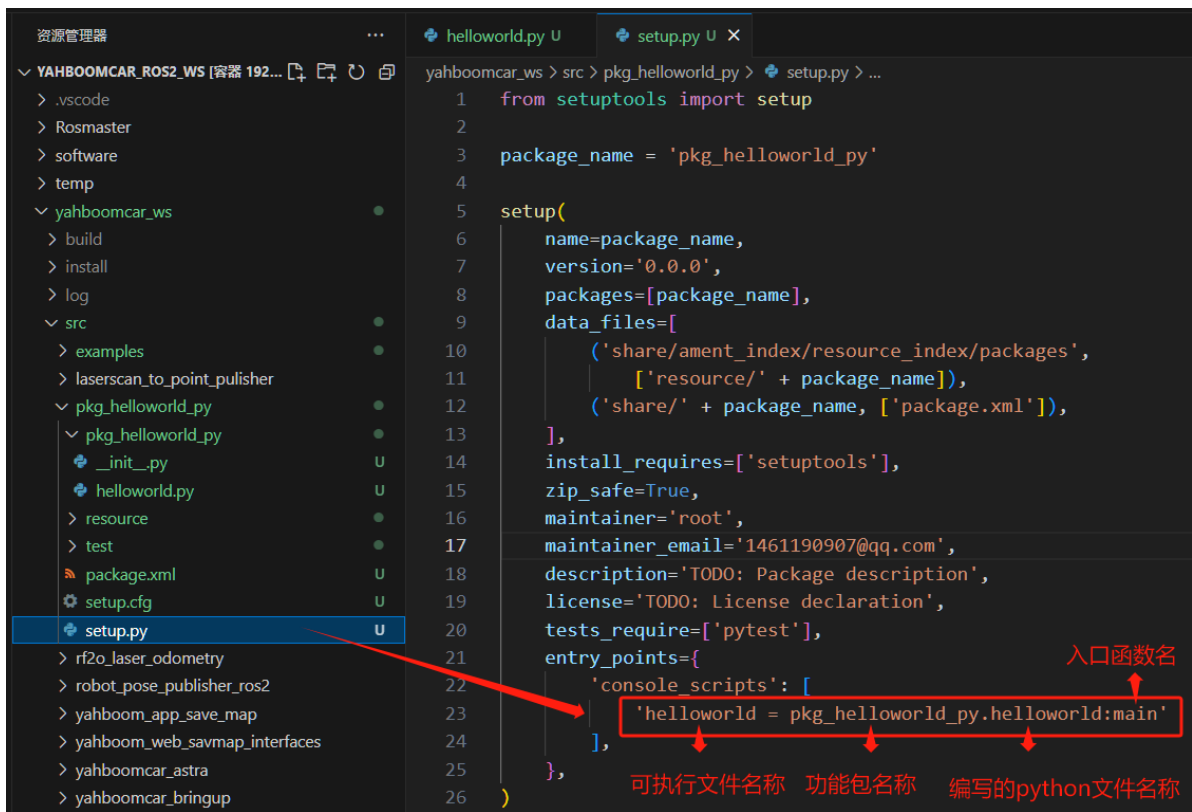
```
import rclpy                                # ROS2 Python接口库
from rclpy.node import Node                 # ROS2 节点类
import time

"""
创建一个HelloWorld节点，初始化时输出“hello world”日志
"""

class HelloWorldNode(Node):
    def __init__(self, name):
        super().__init__(name)              # ROS2节点父类初始化
        while rclpy.ok():                   # ROS2系统是否正常运行
            self.get_logger().info("Hello world") # ROS2日志输出
            time.sleep(0.5)                  # 休眠控制循环时间

def main(args=None):
    rclpy.init(args=args)                   # ROS2节点主入口main函数
    node = HelloWorldNode("helloworld")      # ROS2 Python接口初始化
    rclpy.spin(node)                        # 创建ROS2节点对象并进行初始化
    node.destroy_node()                     # 循环等待ROS2退出
    rclpy.shutdown()                        # 销毁节点对象
                                           # 关闭ROS2 Python接口
```

完成代码的编写后需要设置功能包的编译选项，让系统知道Python程序的入口，打开功能包的setup.py文件，加入如下入口点的配置：



3.3、编译功能包

- 编译功能包

```
colcon build --packages-select pkg_helloworld_py
```

- 在工作空间下刷新环境变量

```
source install/setup.bash
```

3.4、运行节点

```
ros2 run pkg_helloworld_py helloworld
```

运行成功后，可以在终端中看到循环打印“Hello World”字符串的效果：

```
root@ubuntu:~/yahboomcar_ros2_ws/yahboomcar_ws# ros2 run pkg_helloworld_py helloworld
[INFO] [1698050077.167925417] [helloworld]: Hello World
[INFO] [1698050077.669687813] [helloworld]: Hello World
[INFO] [1698050078.170830157] [helloworld]: Hello World
[INFO] [1698050078.672503205] [helloworld]: Hello World
[INFO] [1698050079.174207487] [helloworld]: Hello World
[INFO] [1698050079.675991674] [helloworld]: Hello World
[INFO] [1698050080.177530190] [helloworld]: Hello World
[INFO] [1698050080.679924285] [helloworld]: Hello World
[INFO] [1698050081.182654103] [helloworld]: Hello World
[INFO] [1698050081.685459571] [helloworld]: Hello World
[INFO] [1698050082.188045608] [helloworld]: Hello World
```

