

1.创建包

首先在 ~/ros2_ws/src 目录创建一个名为pynode 的包：

```
cd ~/ros2_ws/src
```

```
ros2 pkg create --build-type ament_python pynode
```

2.编写发布者节点代码

新建一个py文件，命名为py_pub.py

```
import rclpy
from rclpy.node import Node
from std_msgs.msg import String

class Talker(Node):
    def __init__(self):
        super().__init__('talker')
        self.publisher_ = self.create_publisher(String, 'topic', 10)
        self.timer = self.create_timer(1.0, self.timer_callback)
        self.i = 0

    def timer_callback(self):
        msg = String()
        msg.data = f'Hello, world! {self.i}'
        self.publisher_.publish(msg)
        self.get_logger().info(f'Publishing: "{msg.data}"')
        self.i += 1

def main(args=None):
    rclpy.init(args=args)
    talker = Talker()
    rclpy.spin(talker)
    talker.destroy_node()
    rclpy.shutdown()

if __name__ == '__main__':
    main()
```

2.1代码分析

2.2修改package.xml

进入ros2_ws/src/pynode目录并打开package.xml，按照之前教程要求填写description， maintainer和license。如果你并不想开源你的代码，可以忽略此步。

```
<description>TODO: Package description</description>
<maintainer email="pi@todo.todo">pi</maintainer>
<license>TODO: License declaration</license>
```

在编译工具依赖ament_python后

```
<export>
  <build_type>ament_python</build_type>
</export>
```

添加下列依赖项：

```
<exec_depend>rclpy</exec_depend>
<exec_depend>std_msgs</exec_depend>
```

改写完毕后注意记得保存文档!

2.3添加入口点

接着打开 [setup.py](#) 文件，注意保持如下内容和 package一致

```
maintainer_email='pi@todo.todo',
description='TODO: Package description',
license='TODO: License declaration',
```

.把 entry_points 字段改为如下内容并保存，这样程序就知道要运行哪个函数了，即在运行 ros2 run py_pub talker 命令时，指向 py_pub.py_pub:main：

```
entry_points={
    'console_scripts': [
        'talker=py_pub.py_pub:main'
    ],
},
```

3.编写订阅者节点代码

3.1新建文件

可以新建一个包，依照第二部分创建的方法操作（本节内容不进行展示）或者在上一节py_pub包的基础上操作，里新建一个py文件,命名为py_sub.py,并写入以下内容：

```

import rclpy
from rclpy.node import Node
from std_msgs.msg import String

class Listener(Node):
    def __init__(self):
        super().__init__('listener')
        self.subscription = self.create_subscription(
            String,
            'topic', # 话题名称
            self.listener_callback,
            10) # 队列长度
        self.subscription # 防止未使用变量警告

    def listener_callback(self, msg):
        self.get_logger().info(f'Received: "{msg.data}"')

def main(args=None):
    rclpy.init(args=args)
    listener = Listener()
    rclpy.spin(listener) # 保持节点运行
    listener.destroy_node()
    rclpy.shutdown()

if __name__ == '__main__':
    main()

```

3.2修改入口点

因为订阅者和发布者所需的依赖相同，所以这里依然不需要要修改 package.xml，只需打开 [setup.py](#) 文件，把 entry_points 字段改为如下内容并保存：

```

entry_points={
    'console_scripts': [
        'talker=py_pub.py_pub:main',
        'listener=py_pub.py_sub:main'
    ],
},

```

4实验观察

4.1同时打开两个终端， 其中一个终端输入

```
cd ~/ros2_ws  
source install/setup.bash  
ros2 run py_sub talker
```

4.2 另外一个终端输入

```
cd ~/ros2_ws  
source install/setup.bash  
ros2 run py_sub listener
```