

Raken Putra Athallah

1103204186

Tk-44-02

Technical Report UTS Robotik

Write ROS2 Subscriber Node With Python

1. ROS2

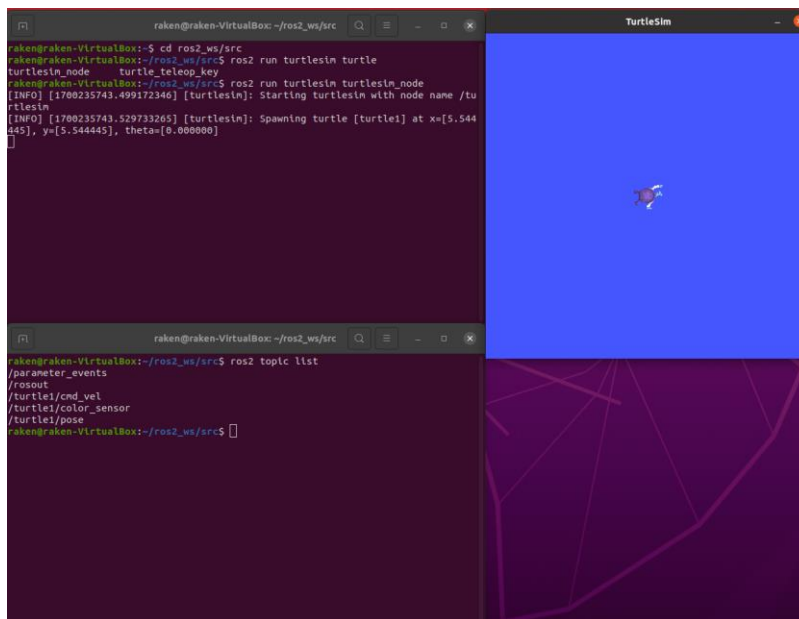
ROS 2, atau Robot Operating System 2, adalah sebuah framework open-source yang dirancang untuk memudahkan pengembangan, pengujian, dan operasi robotika. Ini merupakan evolusi dari ROS (Robot Operating System) yang pertama, dengan tujuan meningkatkan beberapa aspek kinerja dan fleksibilitas.

2. Subscriber Node

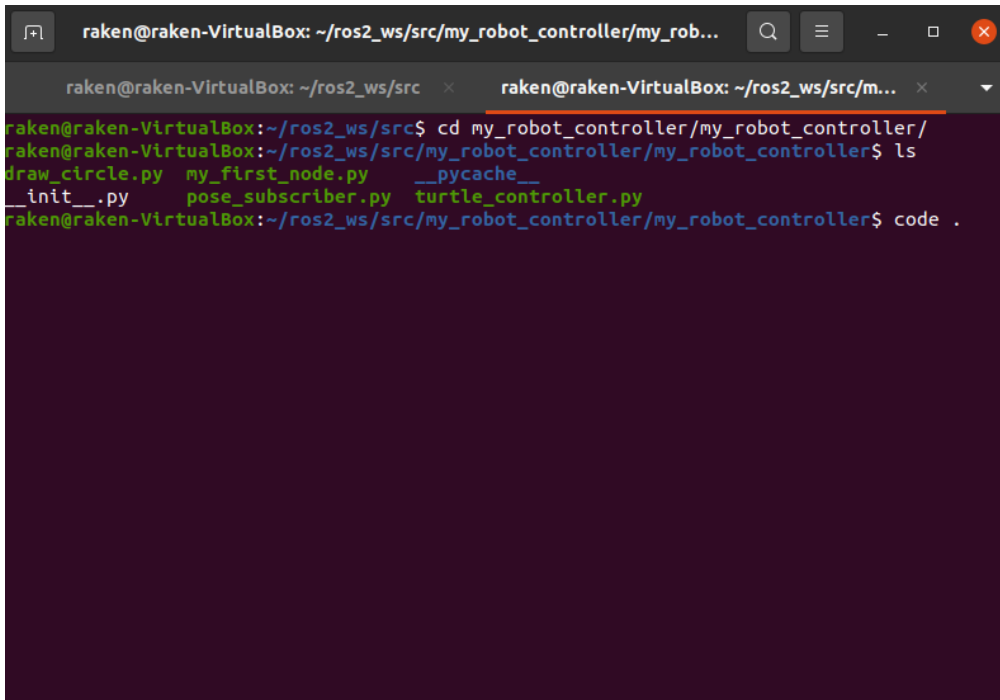
ROS 2 Subscriber Node adalah suatu entitas perangkat lunak yang berjalan di dalam lingkungan Robot Operating System 2 (ROS 2) dan bertindak sebagai penerima atau pelanggan (subscriber) dari data yang dikirim oleh Publisher Node. Dalam arsitektur publikasi-berlangganan (publish-subscribe) yang digunakan oleh ROS 2, Subscriber Node menerima data dari topik (topic) yang diterbitkan oleh Publisher Node.

3. Langkah-Langkah Yang Dilakukan Untuk Membuat Subscriber Node

- Memilih sebuah topik untuk dijadikan membuat subscriber node pada topic tersebut. Pada tahap ini saya memanggil nodes turtlesim dan menggunakan topic bernama **'/turtle1/pose'** untuk dibuatkan subscriber node-nya

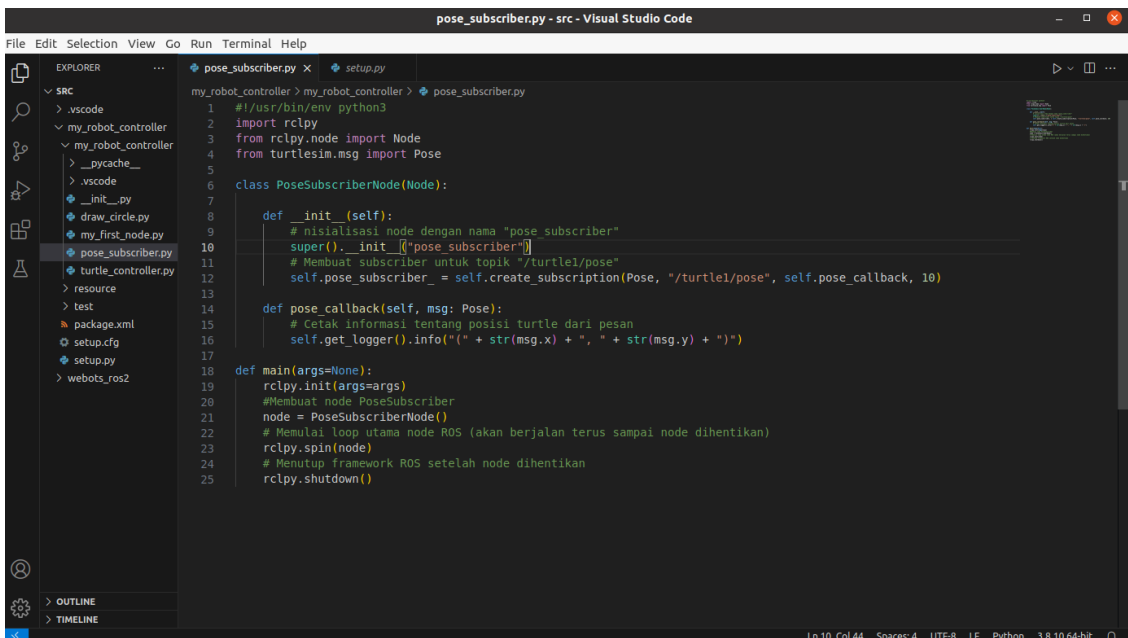


- Membuat file python bernama **‘Pose_subscriber’** didalam direktori yang akan digunakan. Dan melakukan koding dengan menggunakan visual studio code didalam file yang telah dibuat tersebut.



```
raken@raken-VirtualBox: ~/ros2_ws/src/my_robot_controller/my_robot_controller$ cd my_robot_controller/my_robot_controller/
raken@raken-VirtualBox: ~/ros2_ws/src/my_robot_controller/my_robot_controller$ ls
draw_circle.py  my_first_node.py  __pycache__
__init__.py     pose_subscriber.py  turtle_controller.py
raken@raken-VirtualBox: ~/ros2_ws/src/my_robot_controller/my_robot_controller$ code .
```

- Membuat koding pose subscriber berdasarkan topic node yang telah dipilih

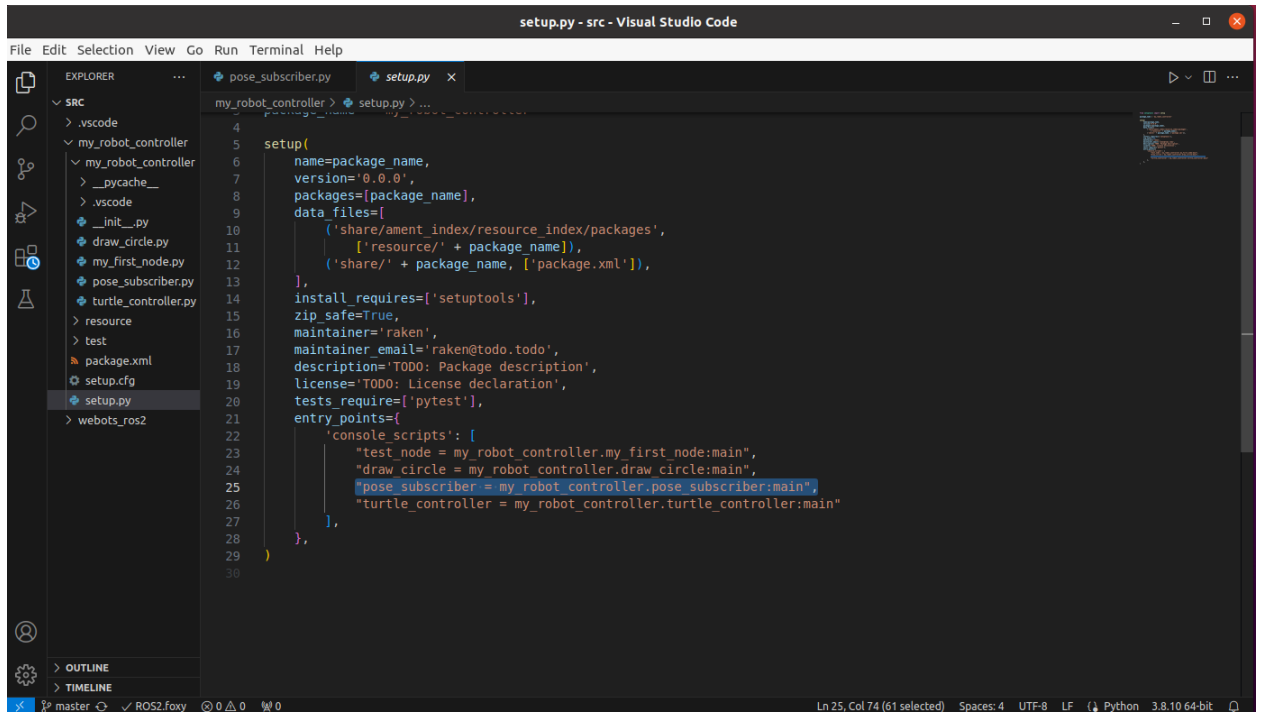


```
pose_subscriber.py - src - Visual Studio Code
File Edit Selection View Go Run Terminal Help

EXPLORER
SRC
  .vscode
  my_robot_controller
    my_robot_controller
      __pycache__
      __init__.py
      draw_circle.py
      my_first_node.py
      pose_subscriber.py
      turtle_controller.py
    resource
    test
  package.xml
  setup.cfg
  setup.py
  webots_ros2

pose_subscriber.py
1  #!/usr/bin/env python3
2  import rclpy
3  from rclpy.node import Node
4  from turtlesim.msg import Pose
5
6  class PoseSubscriberNode(Node):
7
8      def __init__(self):
9          # inisialisasi node dengan nama "pose subscriber"
10         super().__init__("pose_subscriber")
11         # Membuat subscriber untuk topik "/turtle1/pose"
12         self.pose_subscriber = self.create_subscription(Pose, "/turtle1/pose", self.pose_callback, 10)
13
14     def pose_callback(self, msg: Pose):
15         # Cetak informasi tentang posisi turtle dari pesan
16         self.get_logger().info("(" + str(msg.x) + ", " + str(msg.y) + ")")
17
18     def main(args=None):
19         rclpy.init(args=args)
20         #Membuat node PoseSubscriber
21         node = PoseSubscriberNode()
22         # Memulai loop utama node ROS (akan berjalan terus sampai node dihentikan)
23         rclpy.spin(node)
24         # Menutup framework ROS setelah node dihentikan
25         rclpy.shutdown()
```

- Menginstall node yang telah dibuat didalam file bernama **'setup.py'**. Dan melakukan colconbuild – symlink -install

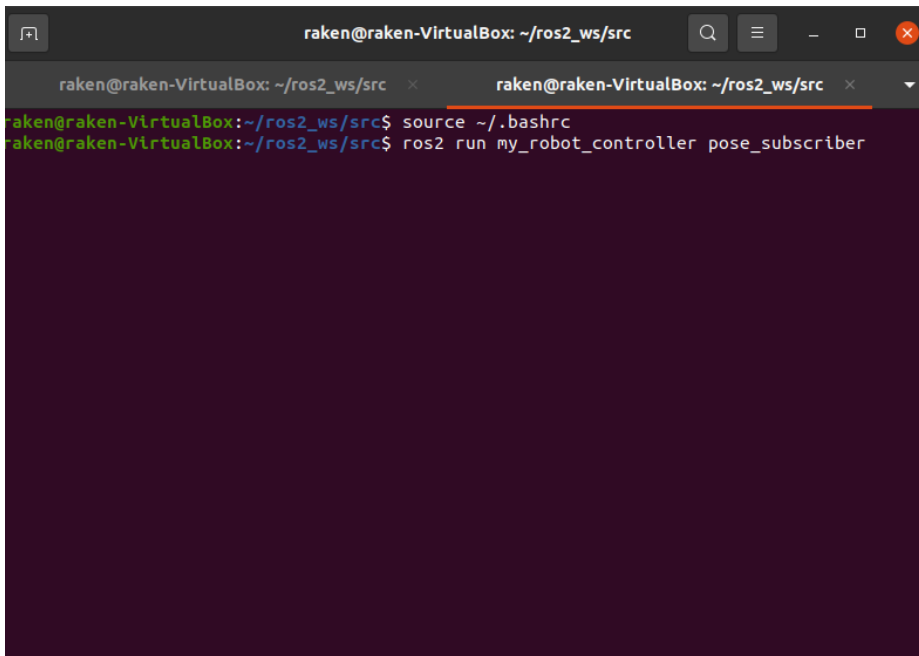


The screenshot shows the Visual Studio Code interface with the 'setup.py' file open. The Explorer panel on the left shows the project structure, including the 'my_robot_controller' package. The main editor displays the following Python code:

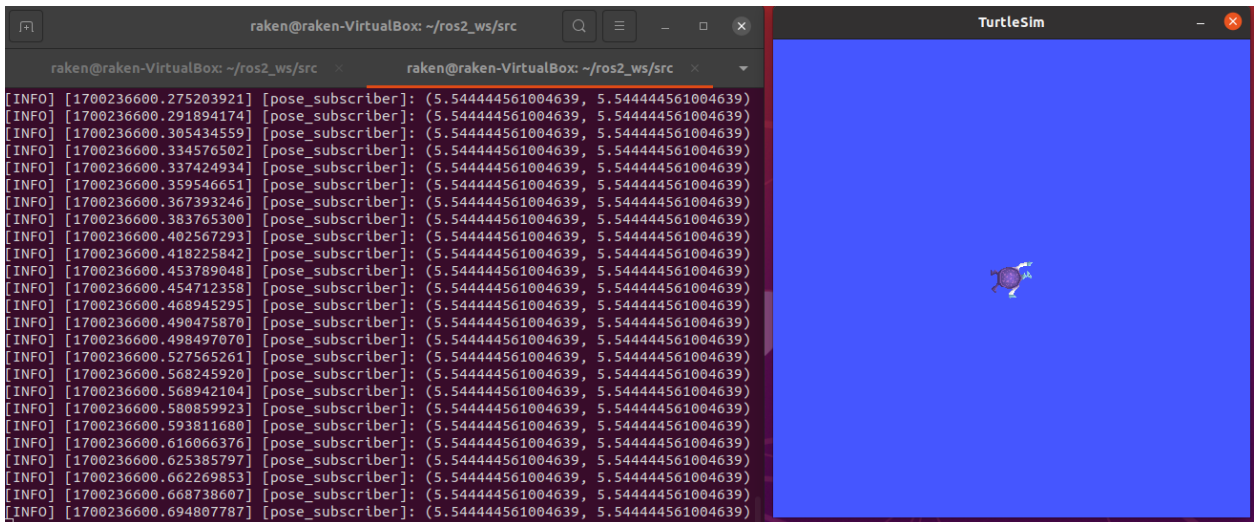
```
4
5
6     name=package_name,
7     version='0.0.0',
8     packages=[package_name],
9     data_files=[
10         ('share/ament_index/resource_index/packages',
11          ['resource/' + package_name]),
12         ('share/' + package_name, ['package.xml']),
13     ],
14     install_requires=['setuptools'],
15     zip_safe=True,
16     maintainer='raken',
17     maintainer_email='raken@todo.todo',
18     description='TODO: Package description',
19     license='TODO: License declaration',
20     tests_require=['pytest'],
21     entry_points={
22         'console_scripts': [
23             'test_node = my_robot_controller.my_first_node:main',
24             'draw_circle = my_robot_controller.draw_circle:main',
25             'pose_subscriber = my_robot_controller.pose_subscriber:main',
26             'turtle_controller = my_robot_controller.turtle_controller:main'
27         ],
28     },
29 )
30
```

The status bar at the bottom indicates the file is at line 25, column 74 (61 selected), with 4 spaces, UTF-8 encoding, LF line endings, and Python 3.8.10 64-bit.

- Menjalankan node pose subscriber yang telah dibuat



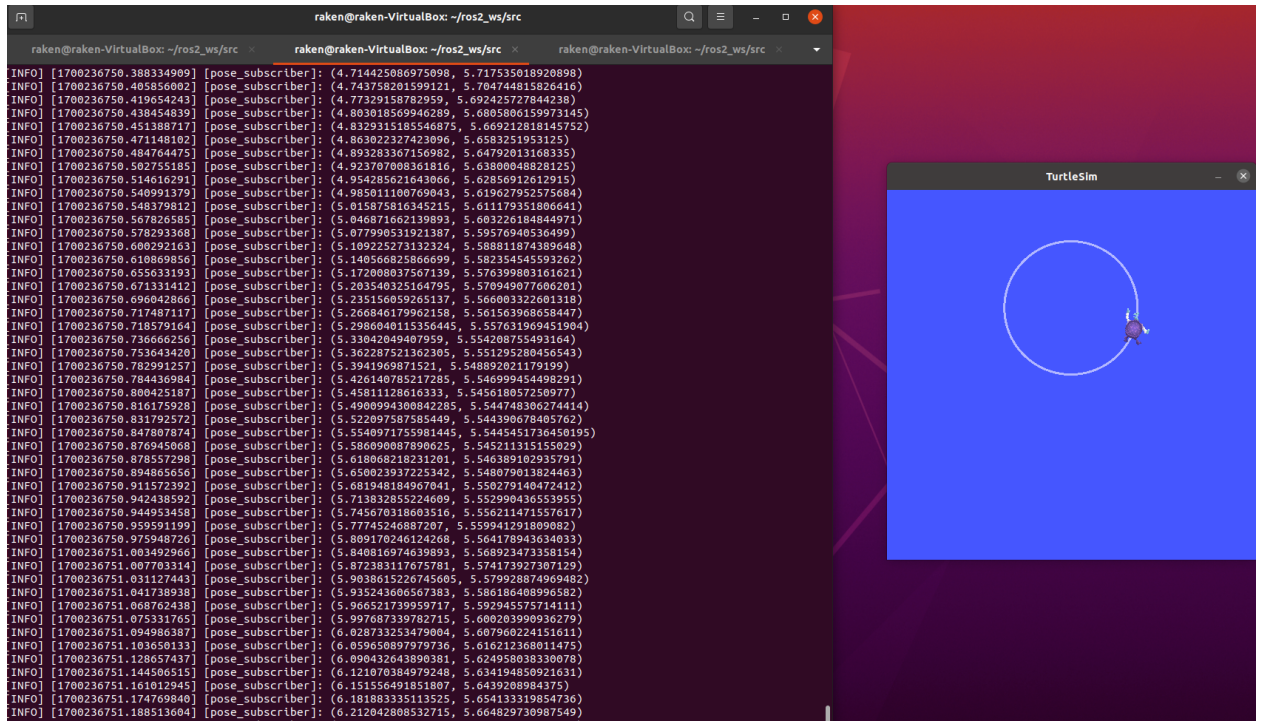
```
raken@raken-VirtualBox: ~/ros2_ws/src
raken@raken-VirtualBox: ~/ros2_ws/src
raken@raken-VirtualBox:~/ros2_ws/src$ source ~/.bashrc
raken@raken-VirtualBox:~/ros2_ws/src$ ros2 run my_robot_controller pose_subscriber
```



```
raken@raken-VirtualBox: ~/ros2_ws/src
raken@raken-VirtualBox: ~/ros2_ws/src
[INFO] [1700236600.275203921] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.291894174] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.305434559] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.334576502] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.337424934] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.359546651] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.367393246] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.383765300] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.402567293] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.418225842] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.453789048] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.454712358] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.468945295] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.490475870] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.498497070] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.527565261] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.568245920] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.568942104] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.580859923] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.593811680] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.61606376] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.625385797] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.662269853] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.668738607] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
[INFO] [1700236600.694807787] [pose_subscriber]: (5.544444561004639, 5.544444561004639)
```

Dari gambar diatas node pose_subscriber menunjukka nilai konstan, dikarenakan posisi kura-kura saat ini tidak bergerak

- Menggerakkan Turtlesim untuk melihat apakah subscriber node tersebut mendapatkan nilai dari posisi kura-kura jika bergerak.



Dari gambar tersebut dapat terlihat nilai yang didapatkan node pose_subscriber berubah mengikuti dari posisi si kura-kura dari node turtlesim