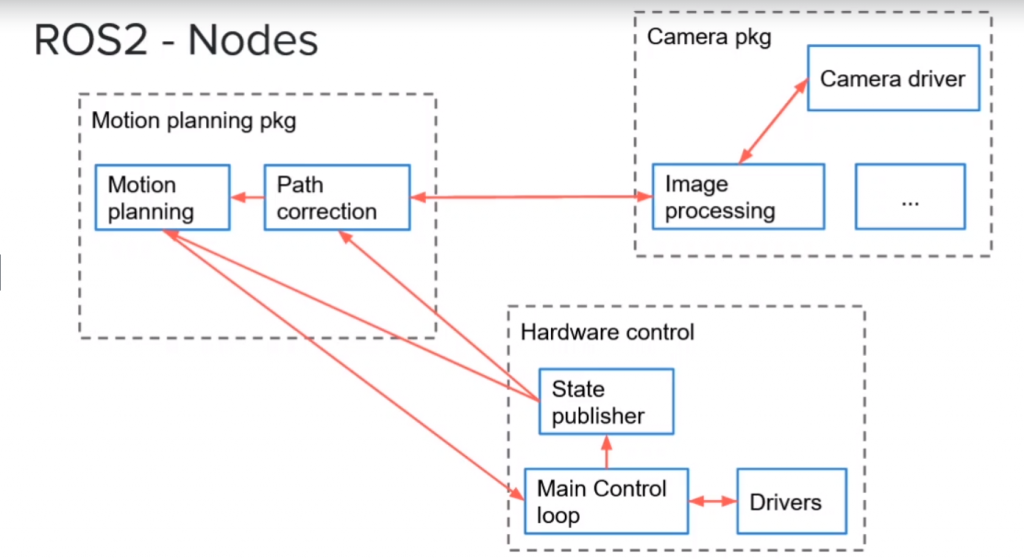
ROS2 – 8 เขียน Node ภาษา Python เบื้องต้น



เขียน Node ภาษา Python เบื้องต้น

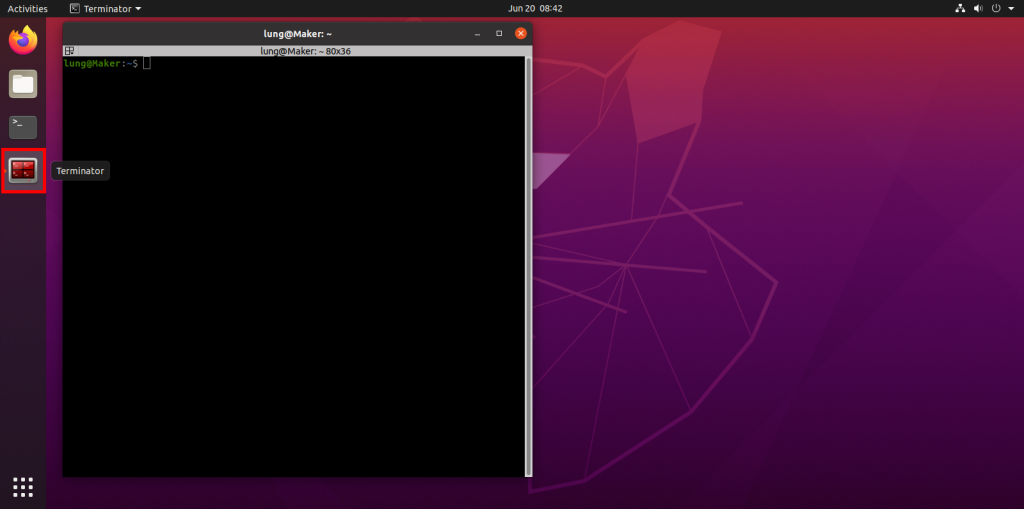
Node :เป็นกระบวนการที่ใช้ในการดำเนินการโปรแกรมต่าง ซึ่ง ROS ได้รับการออกแบบให้เป็นระบบแบบแยกส่วนในระดับที่ละเอียด ระบบควบคุมหุ่นยนต์มักจะประกอบด้วยหลายโหนด ตัวอย่างเช่นหนึ่งโหนดควบคุม laser range-finder, หนึ่งโหนดควบคุมมอเตอร์ล้อ, หนึ่งโหนดทำการแปลงค่าต่างๆ, หนึ่งโหนดทำการวางแผนเส้นทาง, หนึ่งโหนดให้มุมมองกราฟิกของระบบและอื่น ๆ เป็นต้น



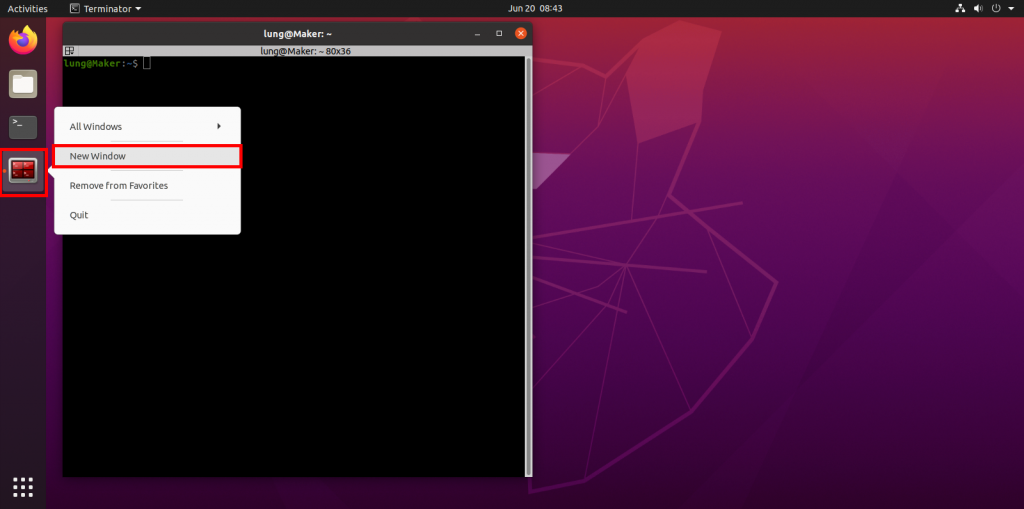
**1 : สร้าง Package ภาษา Python**  
สร้าง Package ภาษา Python ตามขั้นตอนลิงค์ด้านล่าง

* [สร้าง Package ภาษา Python](http://www.lungmaker.com/ros2-6-%e0%b8%aa%e0%b8%a3%e0%b9%89%e0%b8%b2%e0%b8%87-package-%e0%b8%a0%e0%b8%b2%e0%b8%a9%e0%b8%b2-python/)

**2 : เขียน Node ภาษา Python เบื้องต้น**  
เริ่มการทำงานโดย เปิด Terminator หน้าต่างที่ 1



คลิกขวาที่ Terminator -> New Windows



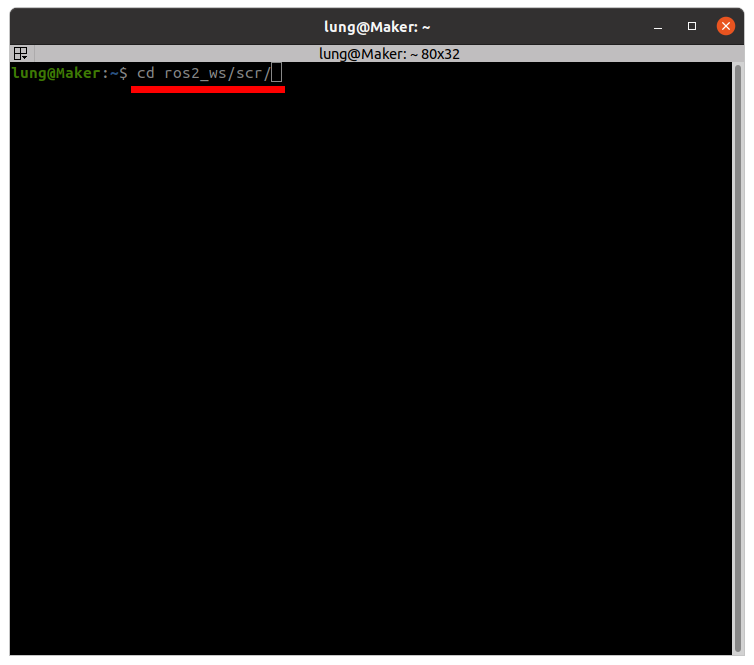
จะได้หน้าต่างที่ 2



เริ่มเขียนคำสั่ง ที่หน้าต่างที่ 1

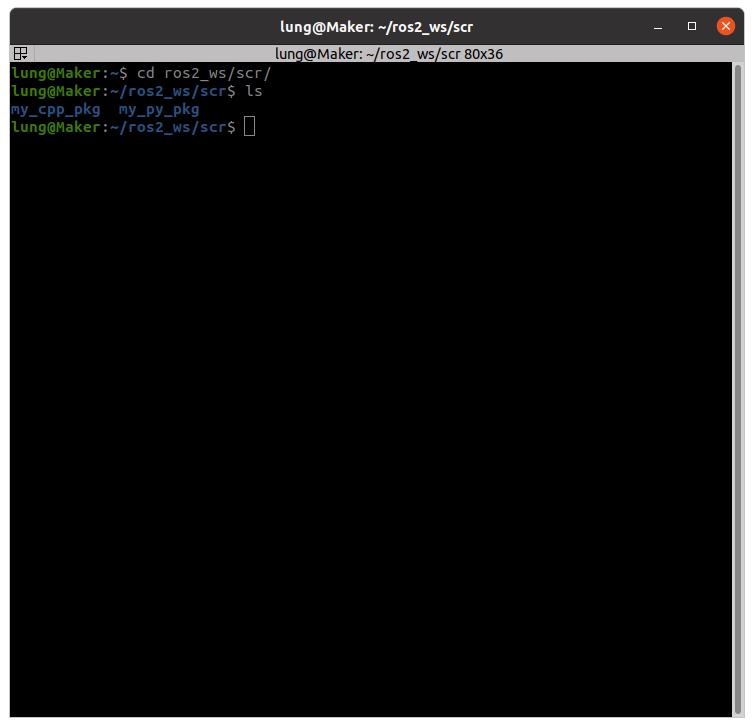
เริ่มสร้าง Node โดยเข้าไปที่โฟลเดอร์ของ Workspace ของ ROS2 โดยเราจะสร้าง Node ไว้ใน ros2\_ws/scr/

cd ros2\_ws/scr/



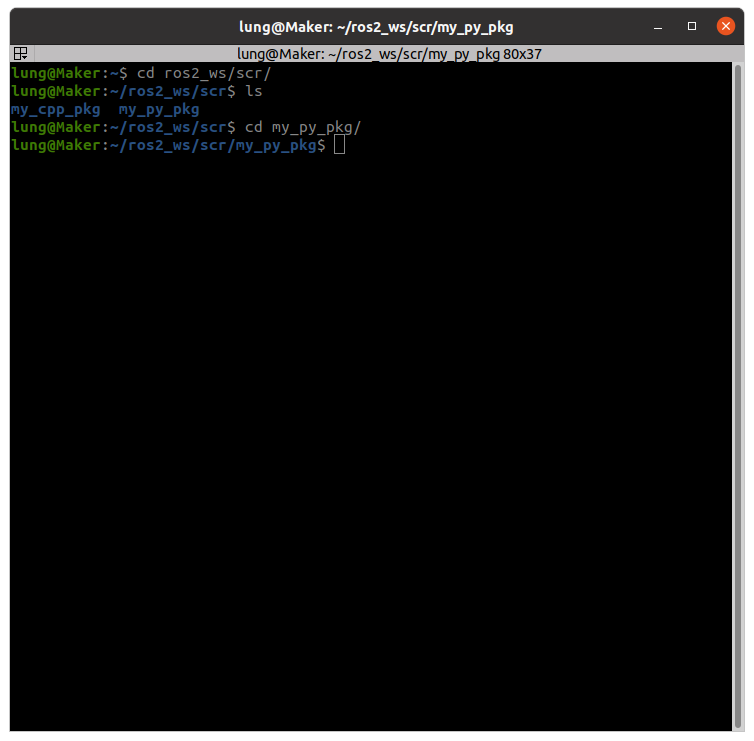
ดูรายชื่อไฟล์และ โฟลเดอร์ จะพบโฟลเดอร์ my\_py\_pkg และ my\_cpp\_pkg ซึ่งเป็น Package ที่ได้สร้างไว้ก่อนหน้านี้

ls



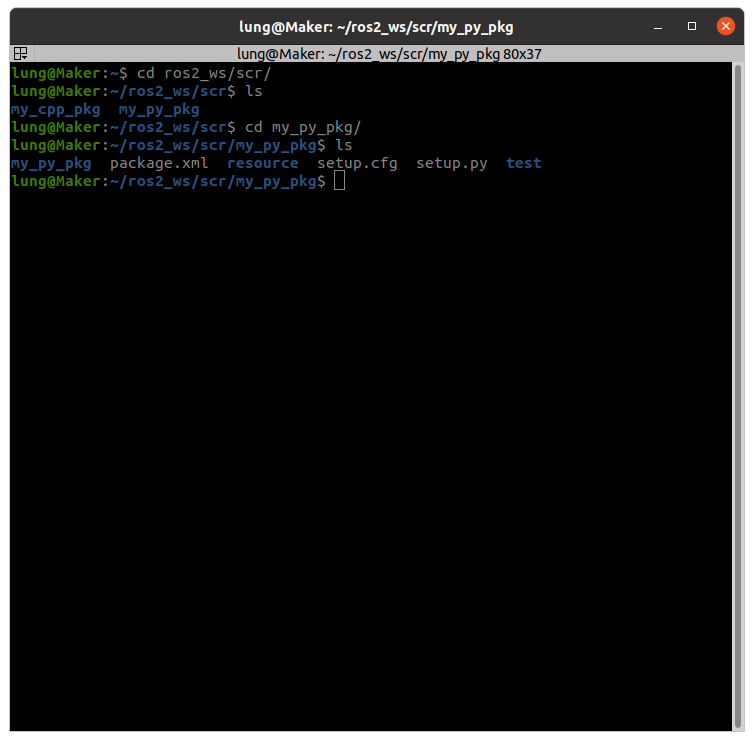
เข้าไปใน โฟลเดอร์ my\_py\_pkg

cd my\_py\_pkg/



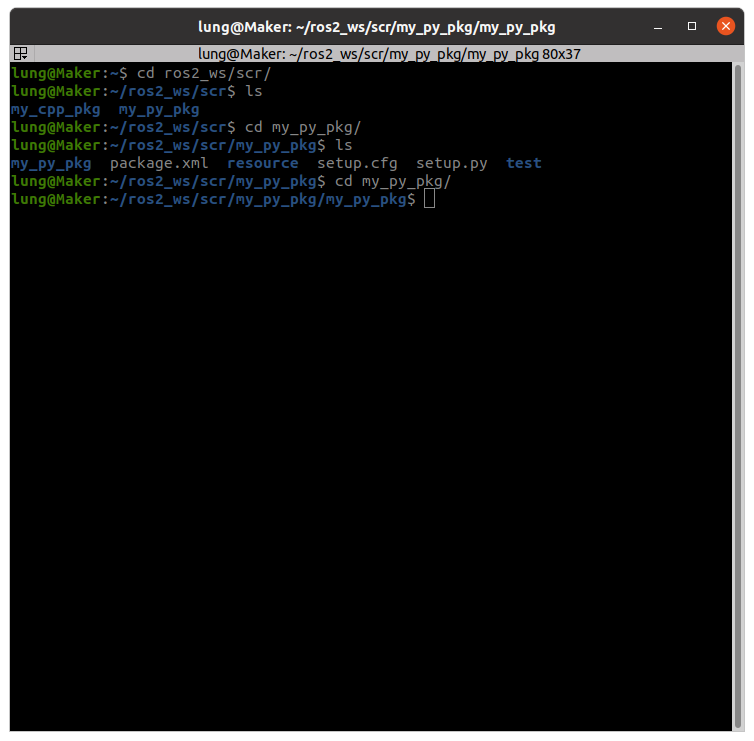
ดูรายชื่อไฟล์และ โฟลเดอร์ จะพบโฟลเดอร์ my\_py\_pkg

ls



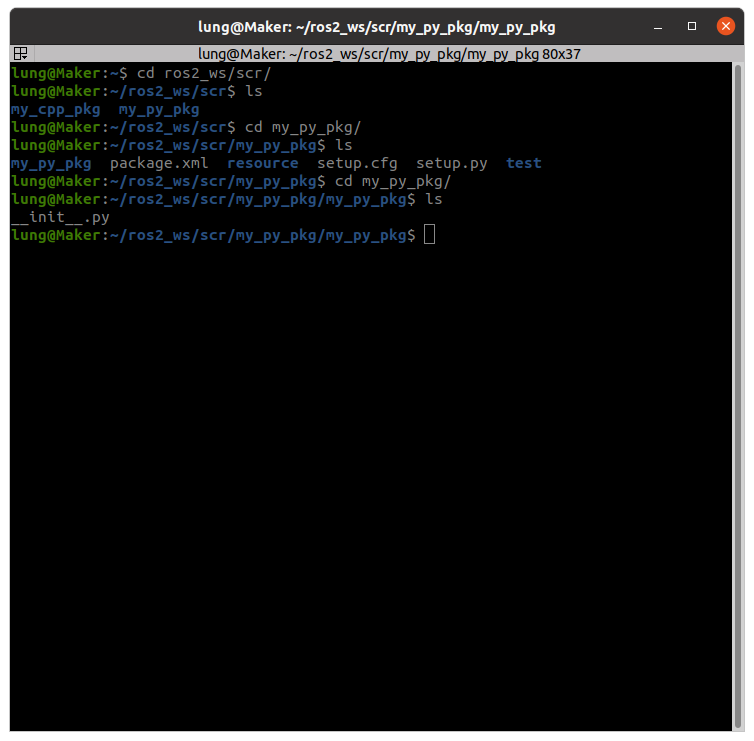
เข้าไปใน โฟลเดอร์ my\_py\_pkg อีกครั้ง

cd my\_py\_pkg/



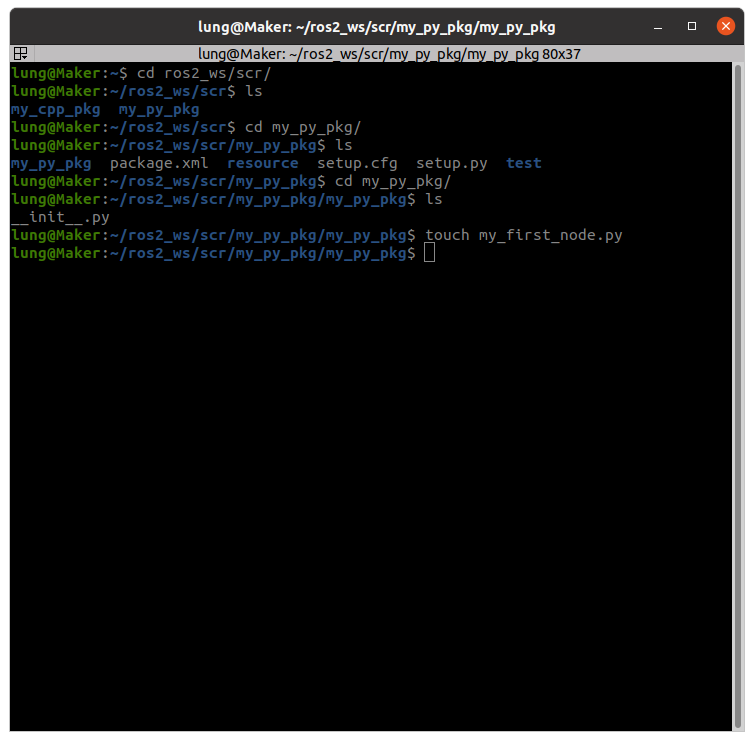
ดูรายชื่อไฟล์และ โฟลเดอร์ จะพบไฟล์ **init.py**

ls



สร้าง Node ภาษา Python ชื่อ my\_first\_node.py

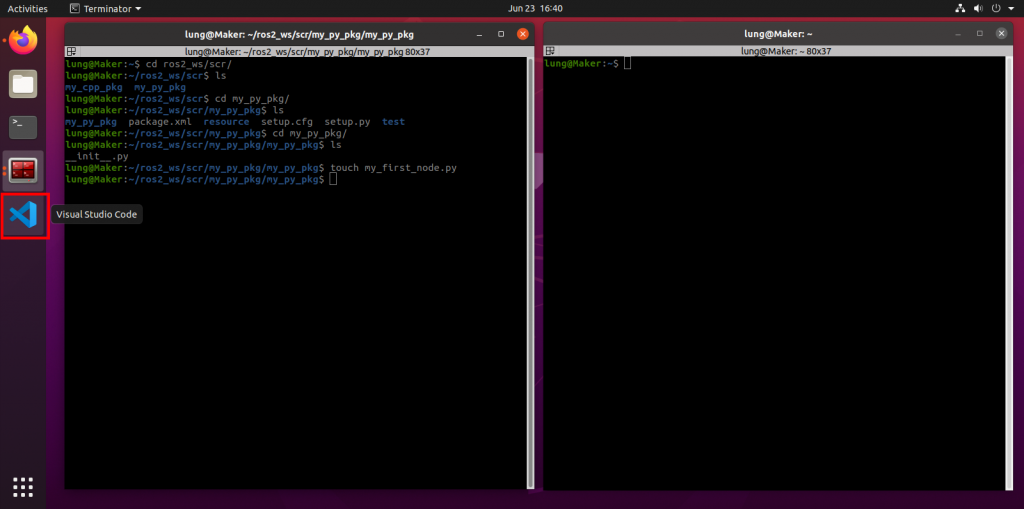
touch my\_first\_node.py



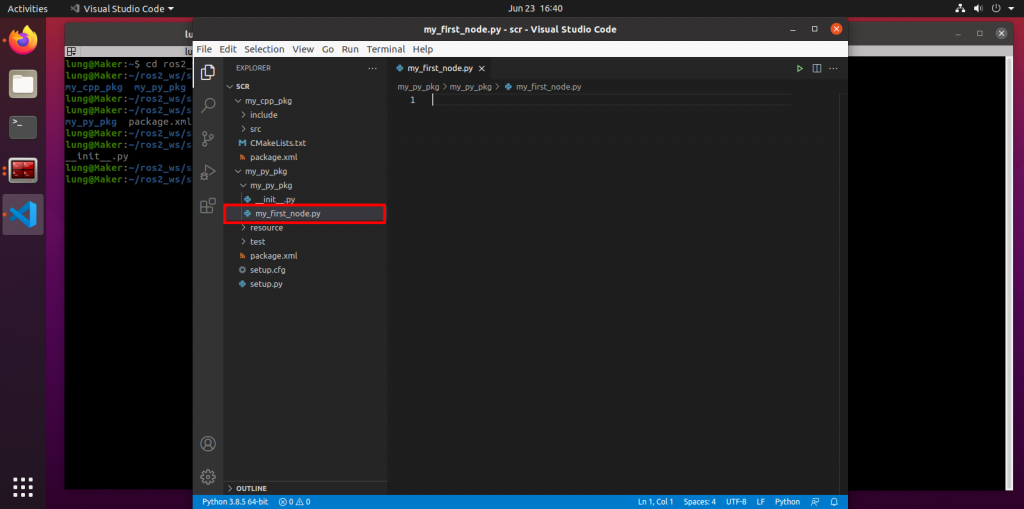
**3 : ติดตั้ง Visual Studio Code**  
ติดตั้ง Visual Studio Code บน Ubuntu 20.04 และ ทดสอบ โปรแกรมแรก ภาษา Python ตามขั้นตอนลิงค์ด้านล่าง

* [ติดตั้ง Visual Studio Code บน Ubuntu 20.04](http://www.lungmaker.com/ros2-5-%e0%b8%95%e0%b8%b4%e0%b8%94%e0%b8%95%e0%b8%b1%e0%b9%89%e0%b8%87-visual-studio-code-%e0%b8%9a%e0%b8%99-ubuntu-20-04/)

**4 : เขียนโค้ดด้วย Visual Studio Code**  
เปิด Visual Studio Code



เปิดโฟลเดอร์ scr แล้วคลิกเปิดไฟล์ my\_first\_node.py เริ่มเขียนโค้ด



เขียนโค้ดดังนี้

#!/usr/bin/env python3

import rclpy

from rclpy.node import Node

def main(args=None):

rclpy.init(args=args)

node = Node("py\_test")

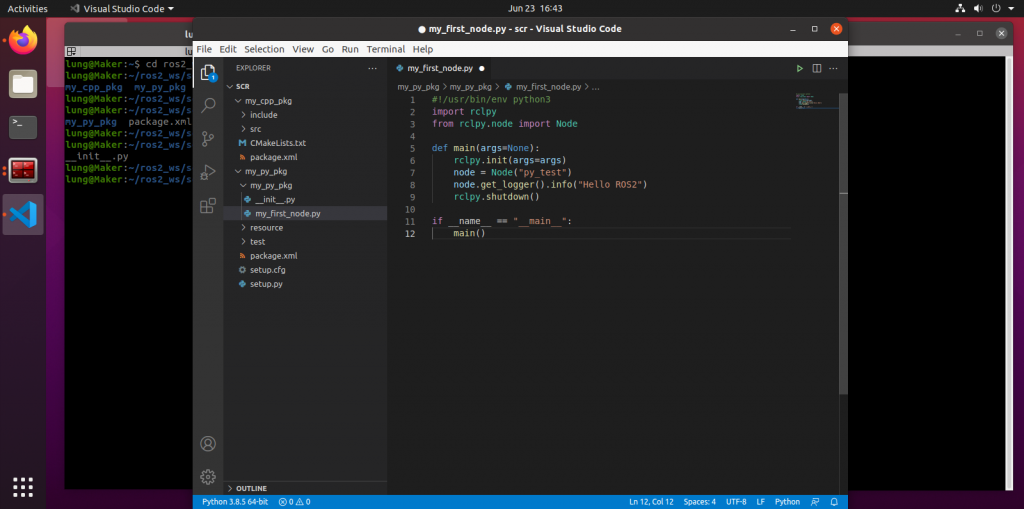
node.get\_logger().info("Hello ROS2")

rclpy.shutdown()

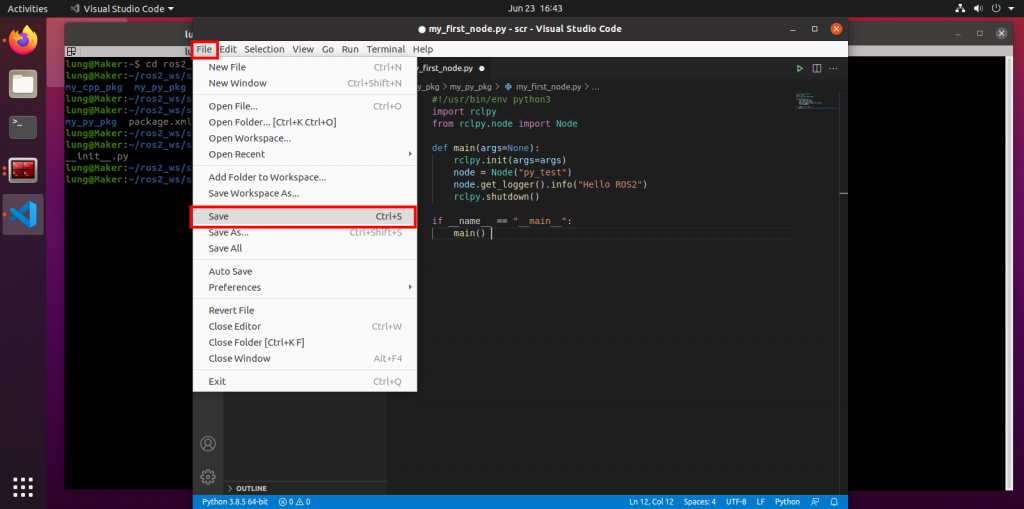
if \_\_name\_\_ == "\_\_main\_\_":

main()

<http://www.lungmaker.com/code/my_first_node.py>

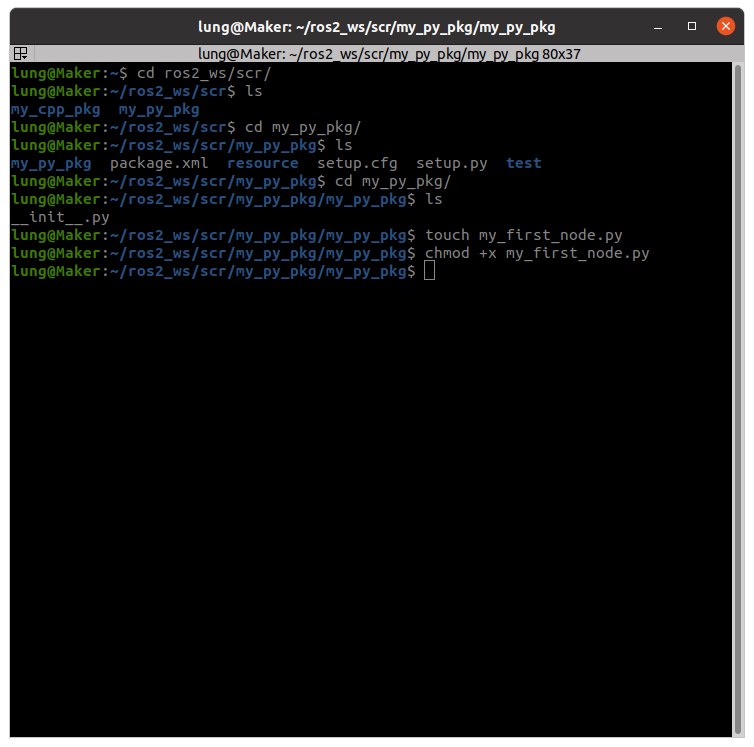


File -> Save



กลับไปที่ Terminator เขียนคำสั่ง

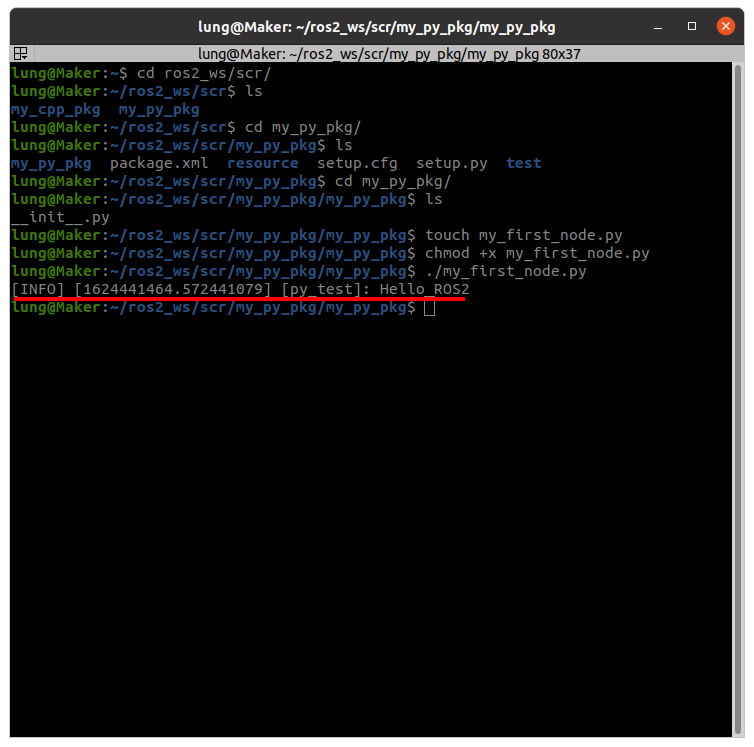
chmod +x my\_first\_node.py



ทดสอบการทำงาน ด้วยคำสั่ง

./my\_first\_node.py

แสดงผลลัพธ์ การทำงาน



แทรกโค้ดด้านล่าง ไปที่ บรรทัดที่ 9 แล้ว Save

rclpy.spin(node)

เขียนโค้ดดังนี้

#!/usr/bin/env python3

import rclpy

from rclpy.node import Node

def main(args=None):

rclpy.init(args=args)

node = Node("py\_test")

node.get\_logger().info("Hello ROS2")

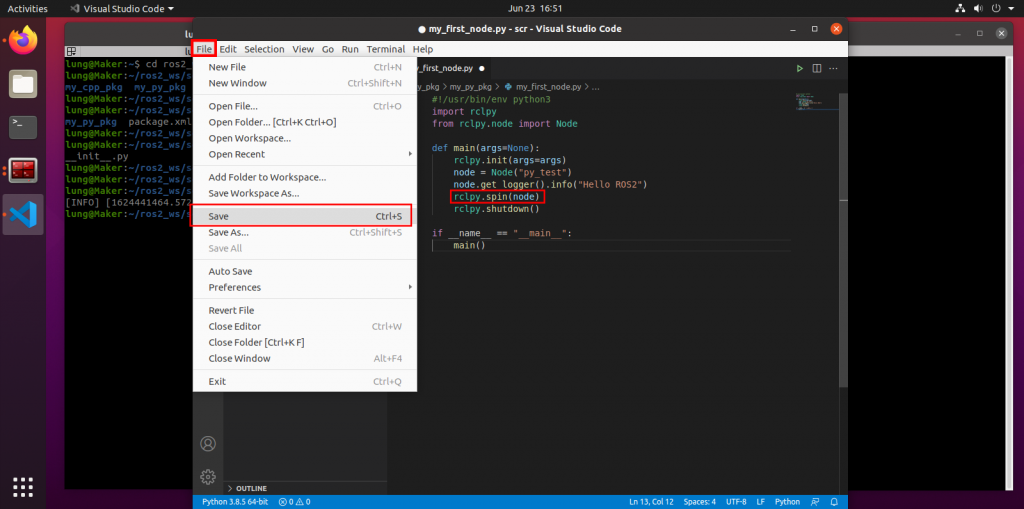
rclpy.spin(node)

rclpy.shutdown()

if \_\_name\_\_ == "\_\_main\_\_":

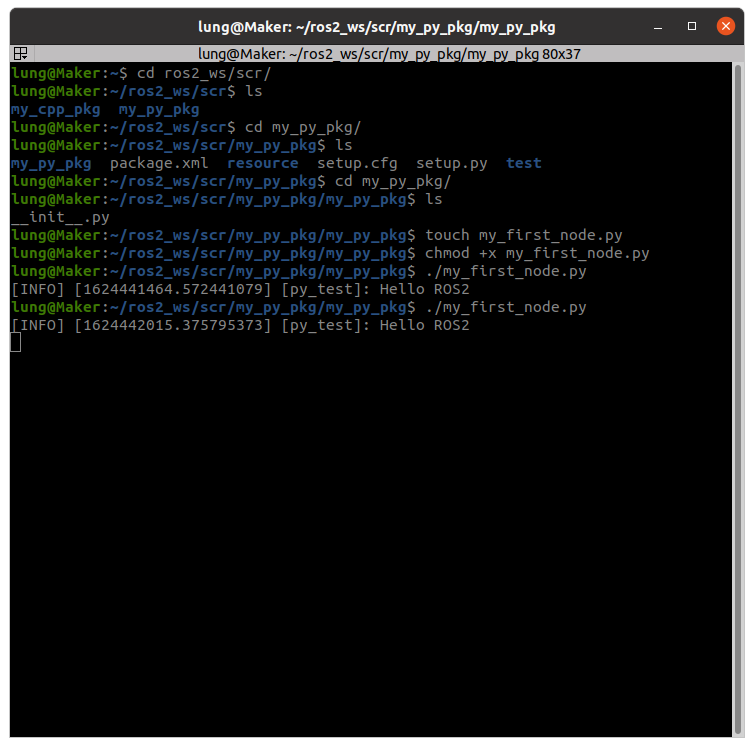
main()

File -> Save

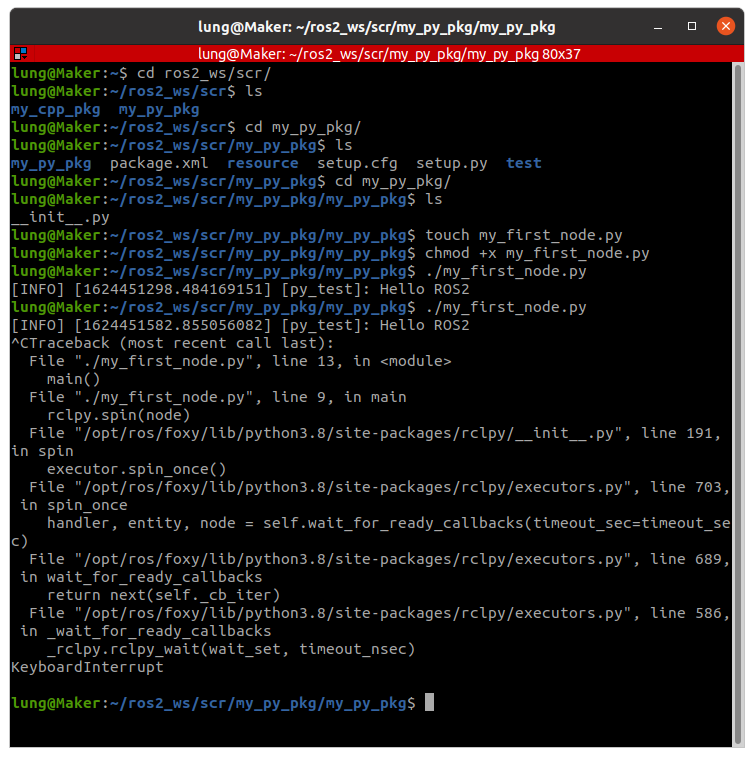


ทดสอบการทำงานอีกครั้ง

./my\_first\_node.py

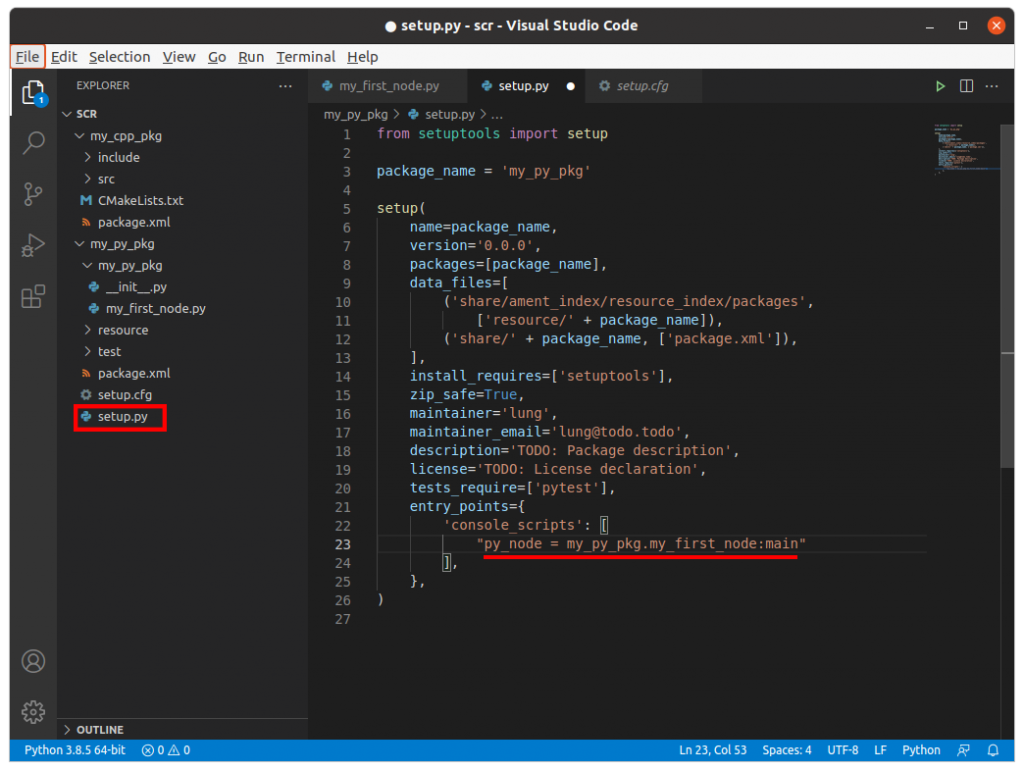


กด Ctrl + c เพื่อหยุดการทำงาน

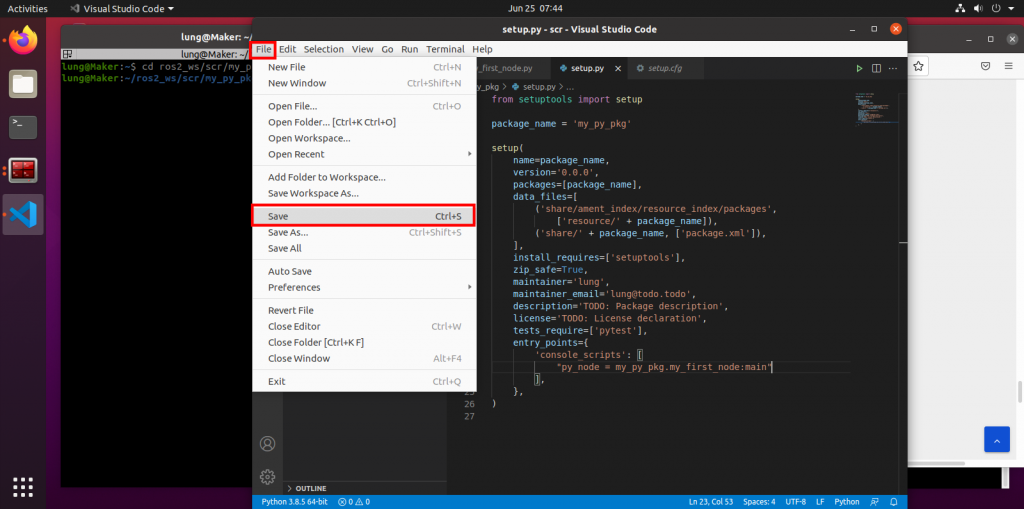


**5 : เรียกใช้งาน py\_node**  
เปิดไฟล์ setup.py เพิ่มโค้ดบรรทัดที่ 23 ล่าง ‘console scripts’ ด้วยโค้ดด้านล่าง

"py\_node = my\_py\_pkg.my\_first\_node:main"



File -> Save



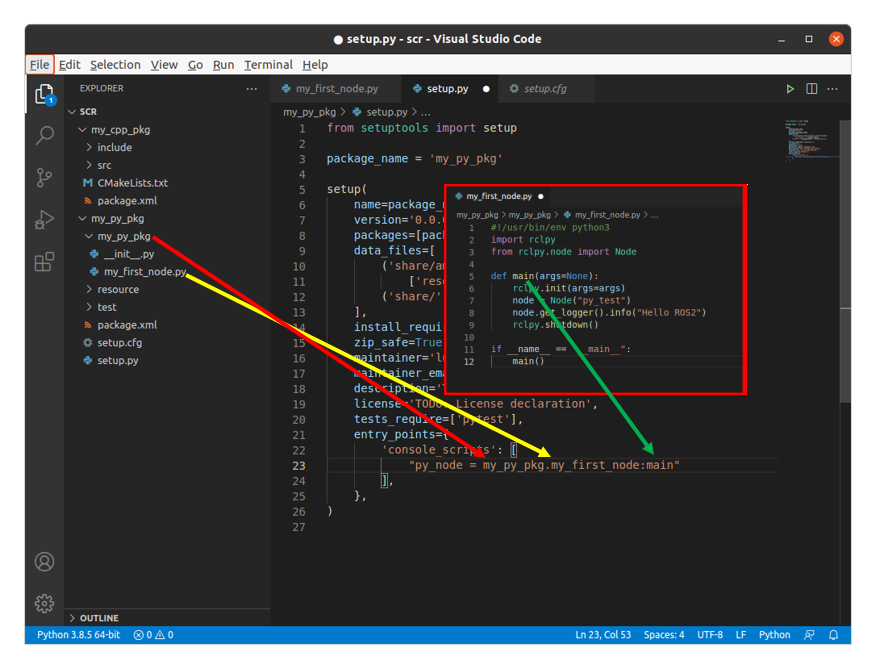
โดย โค้ดที่เพิ่มมี 4 ส่วนดังนี้

1. py\_node คือชื่อที่จะเรียกใช้ในการทำงาน

2. my\_py\_pkg คือ ชื่อ Packag

3. my\_first\_node คือ ชื่อไฟล์ ของโค้ด Python

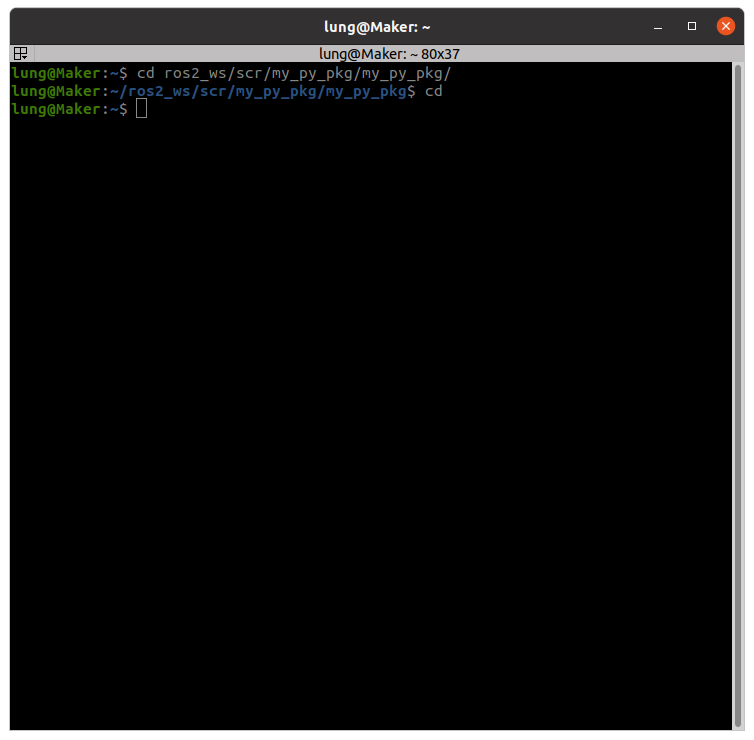
4. main คือ ชื่อฟังก์ชัน จากไฟล์ข้อที่ 3



เริ่มทำงานต่อจากขั้นที่ 4

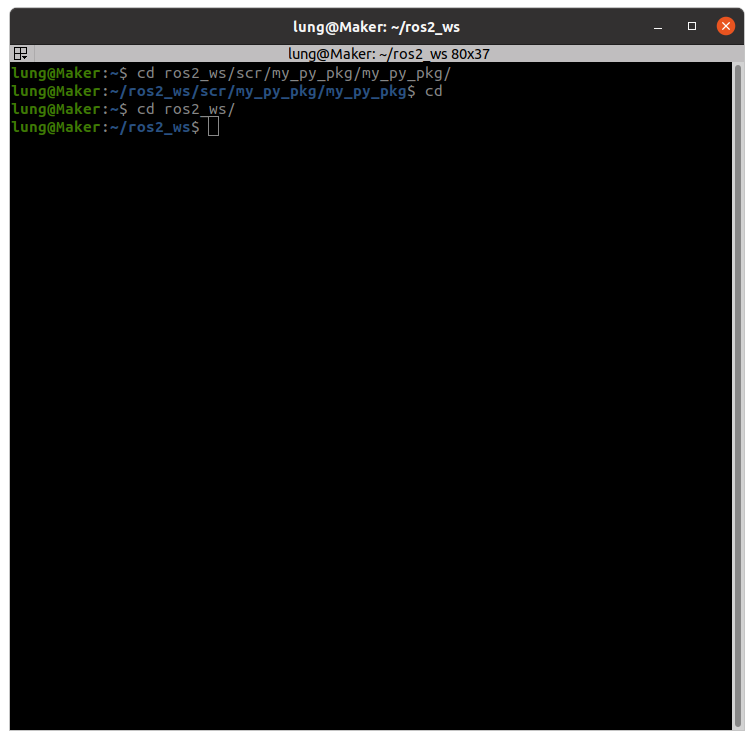
กลับไปที่ Home

cd



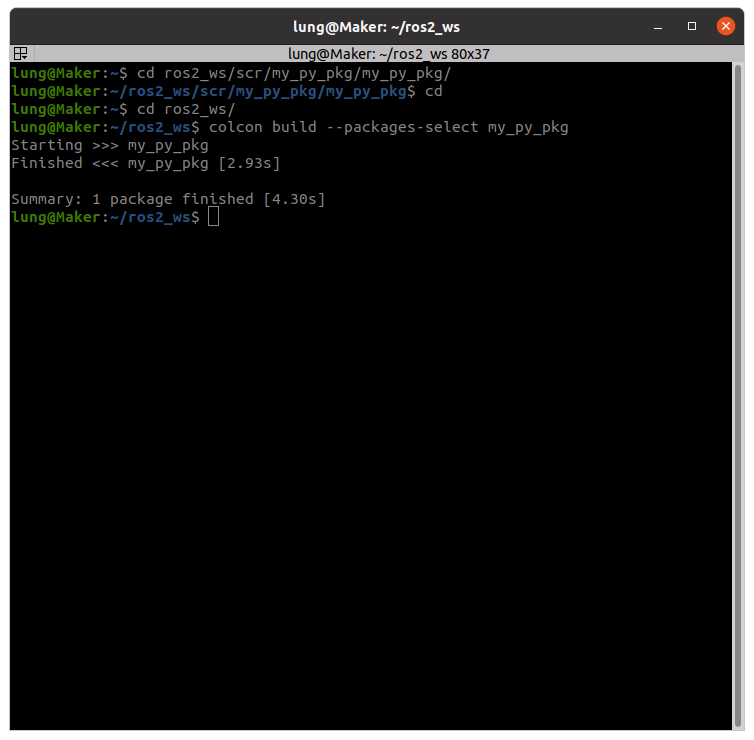
เข้าไปใน โฟลเดอร์ ros2\_ws

cd ros2\_ws/

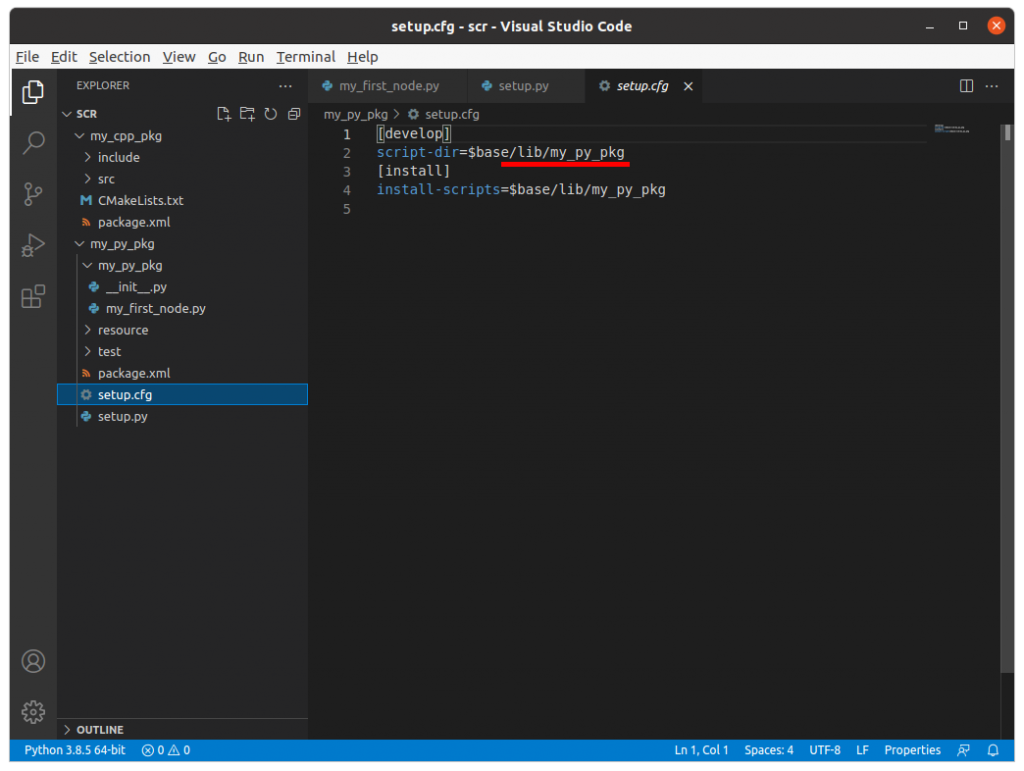


Build แพ็คเกจ my\_py\_pkg อีกครั้ง

colcon build --packages-select my\_py\_pkg

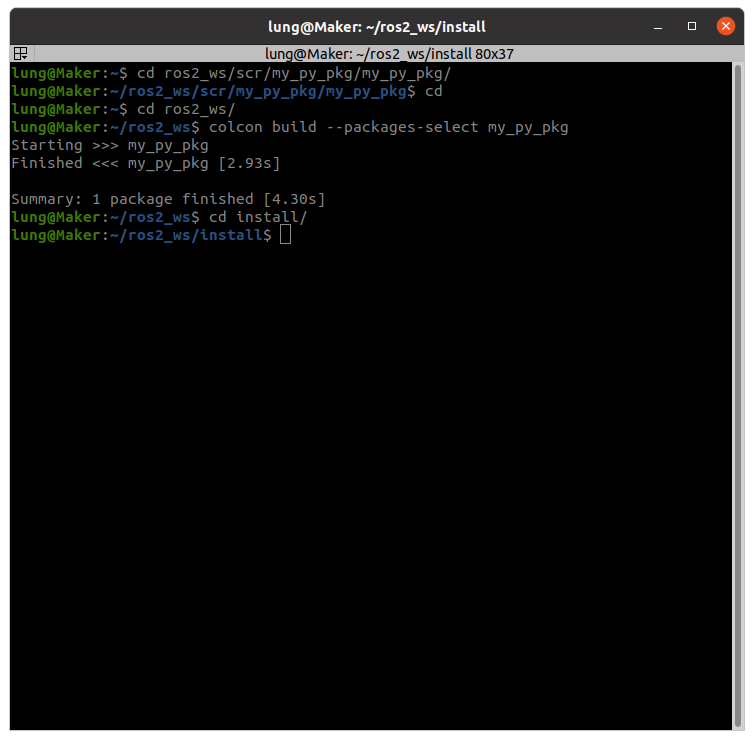


**เรียกใช้งาน py\_node ได้อย่างไร ?**  
เปิดไฟล์ setup.cfg จะแสดงที่อยู่ของ py\_node อยู่ที่ /lib/my\_py\_pkg



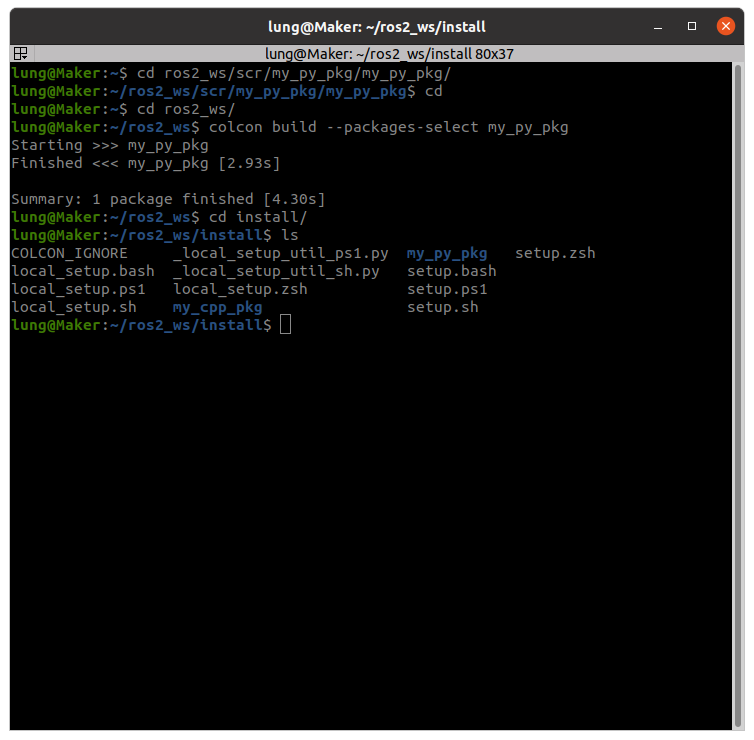
เข้าไปที่โฟลเดอร์ install

cd install/



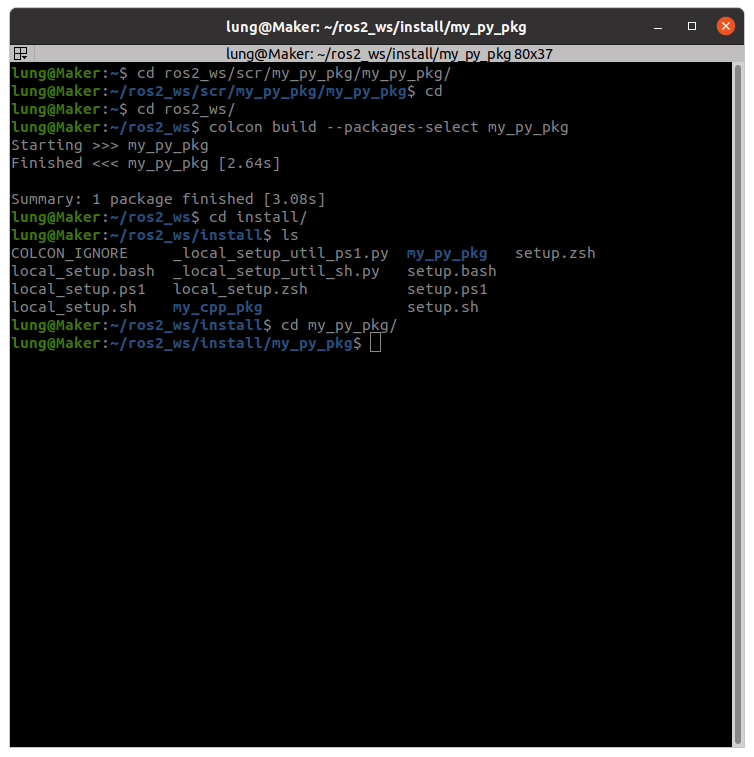
ลิสรายการ

ls



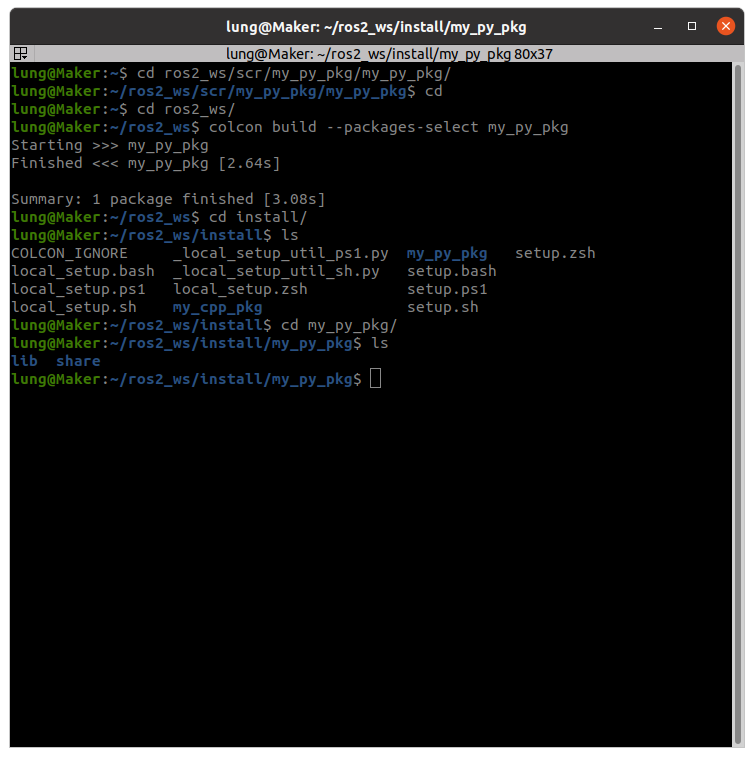
เข้าไปโฟลเดอร์ my\_py\_pkg

cd my\_py\_pkg/



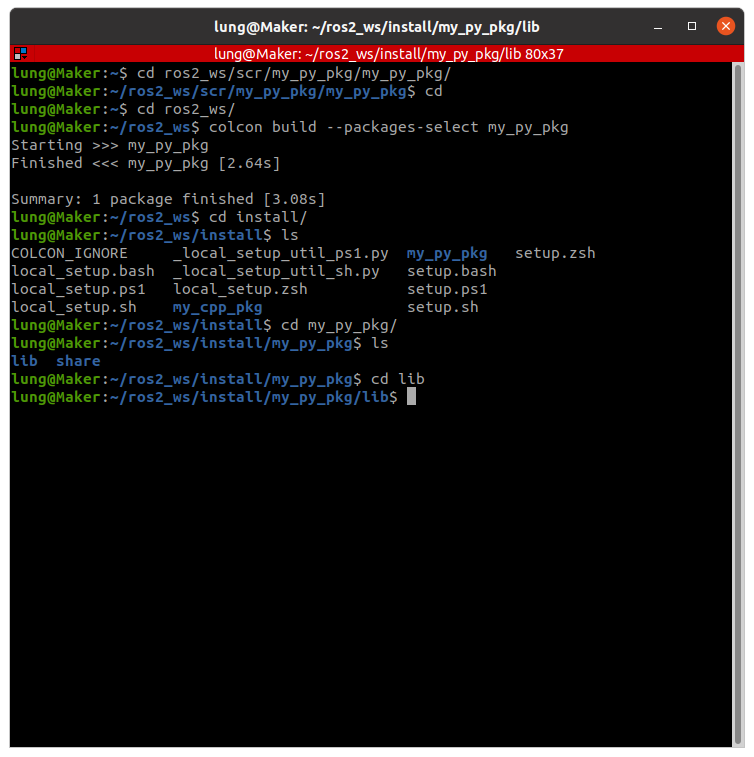
ลิสรายการ

ls



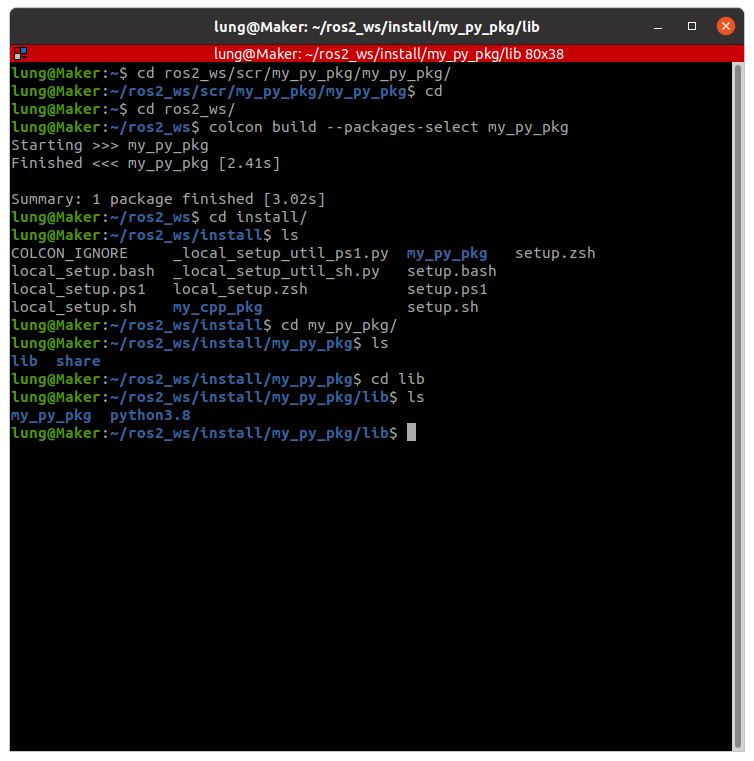
เข้าไปในโฟลเดอร์ lib

cd lib



ลิสรายการ

ls



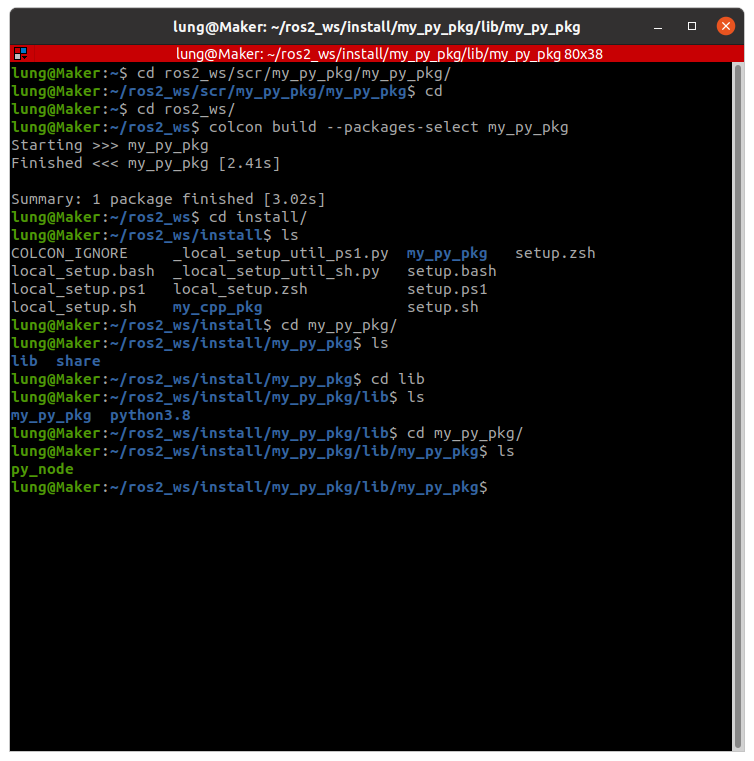
เข้าไปโฟลเดอร์ my\_py\_pkg

cd my\_py\_pkg/



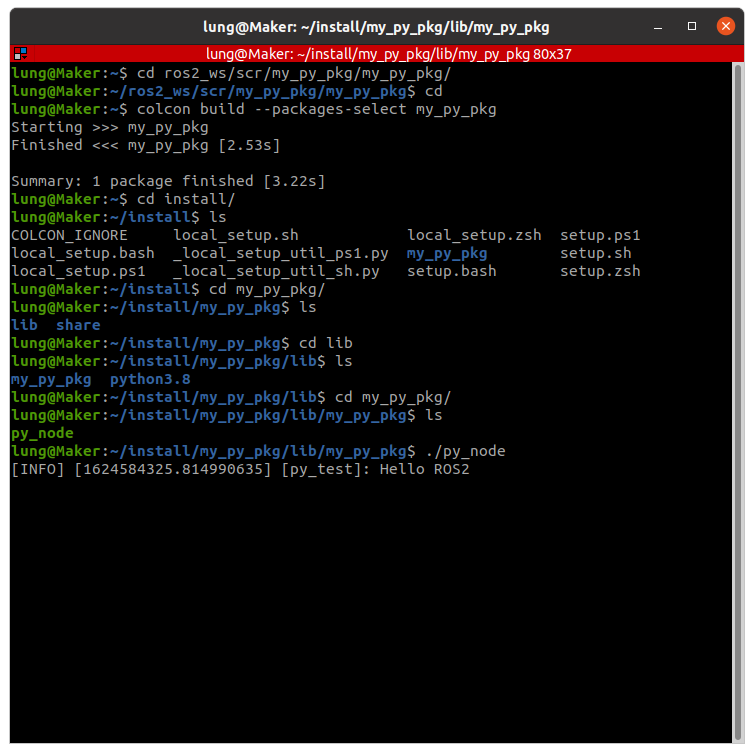
ลิสรายการ จะพบ py\_node อยู่ในโฟลเดอร์นี้

ls

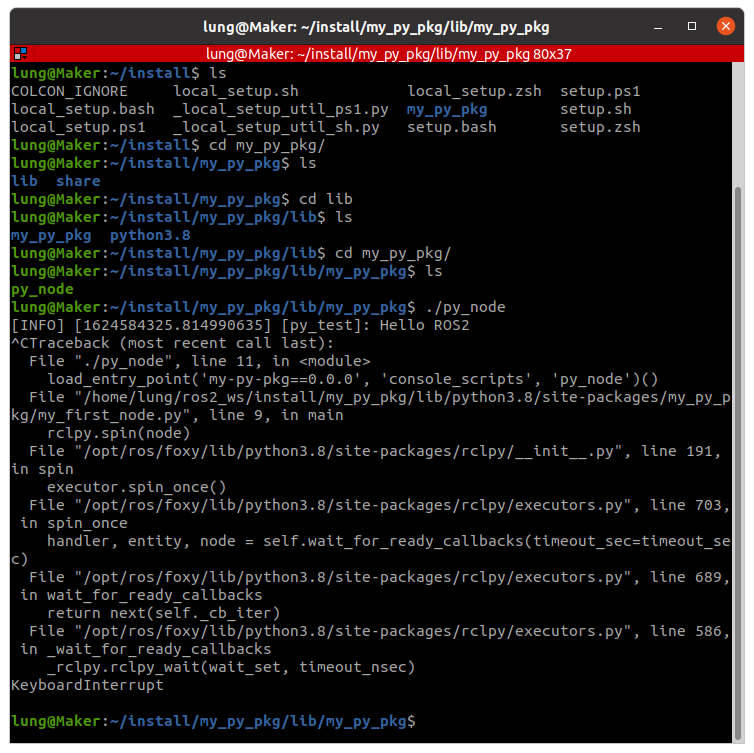


เรียกใช้งาน py\_node จะแสดงผลลัพธ์ การทำงาน

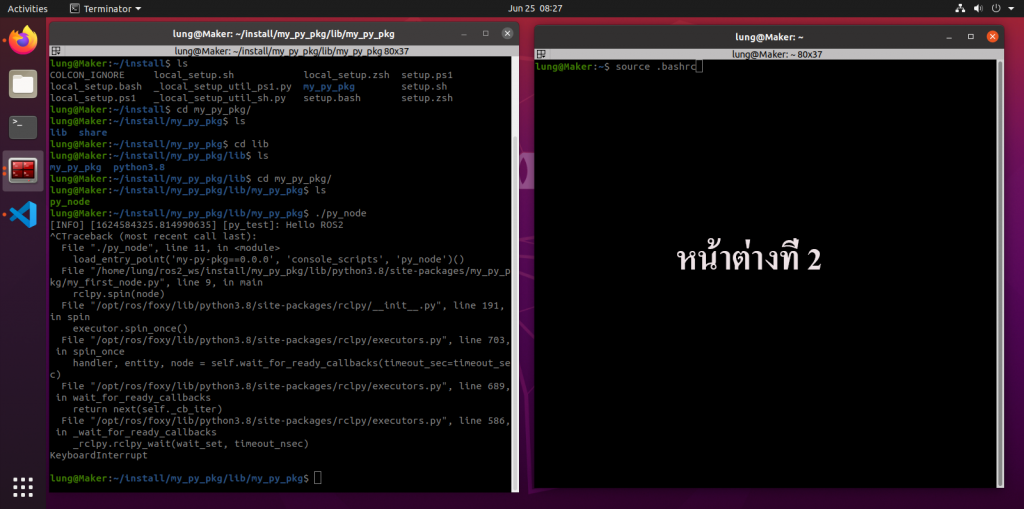
./py\_node



กด Ctrl + c เพื่อหยุดการทำงาน



**6 : เรียกใช้งาน py\_node โดยใช้ ROS2**  
ซึ่งการเรียกใช้งาน py\_node แบบขั้นตอนที่ผ่านมา จะเห็นได้ว่ามีหลายขั้นตอน ที่ยุ่งยาก โดยการเรียกใช้งานด้วย ROS2 มีขั้นตอนดังนี้  
ไปที่ Terminator หน้าต่างที่ 2



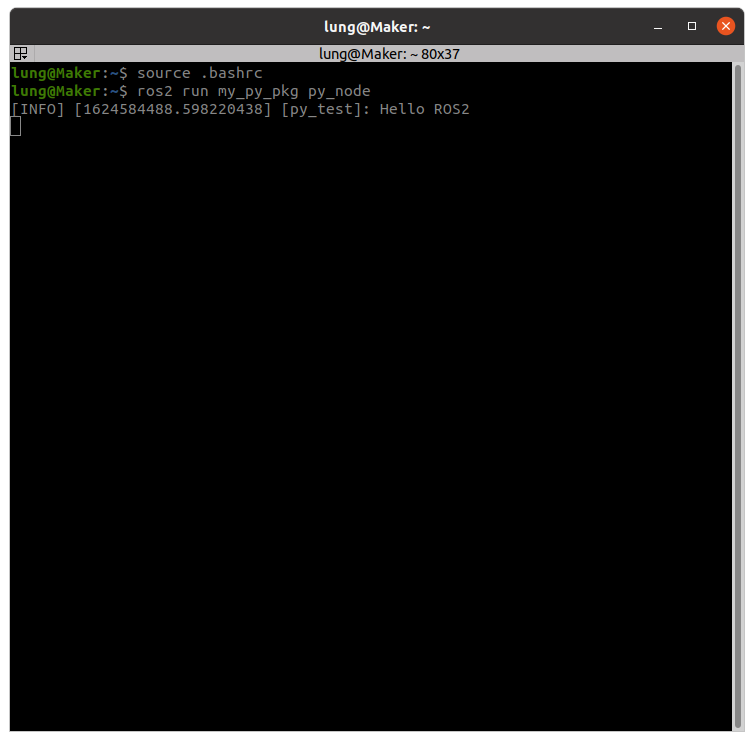
ใช้คำสั่ง

source .bashrc



เรียกใช้งานด้วยคำสั่ง ros2 run ชื่อPackag ชื่อNode จะแสดงผลลัพธ์ การทำงาน

ros2 run my\_py\_pkg py\_node



กด Ctrl + c เพื่อหยุดการทำงาน

