Register number	20BRS1231	**Name**	Harish Kumar. K
Course Title	Simulation and Modeling	**Date of experiment**	20/01/2023
Course code	CSE3102	**Date of submission**	25/01/2023

Exercise in the lab

publisher.py

Code

Publisher.py

```
#!/usr/bin/env python3
import rospy
from std_msgs.msg import String
if __name__ == '__main__':
	rospy.init_node('publisher_node')
	pub = rospy.Publisher("/robot_publisher_node",String, queue_size=10)
	rate = rospy.Rate(2)
	while not rospy.is_shutdown():
		msg = String()
		msg.data = "Hi i am Harish and i am publishing"
		pub.publish(msg)
		rate.sleep()
	rospy.loginfo("Publisher Node stopped")
```

Subscribe.py

1) code

```
import rospy
from std_msgs.msg import String
def callback_receive_radio_data(msg):
	rospy.loginfo("Message received:")
	rospy.loginfo(msg)


if __name__ == '__main__':
	rospy.init_node('subscriber')
	sub = rospy.Subscriber("/robot_publisher_node", String, callback_receive_radio_data)
	rospy.spin()
```

Output

1. Create one publisher and two subscribers

Code

Publisher.py

```
#!/usr/bin/env python3
import rospy
from std_msgs.msg import String
if __name__ == '__main__':
        rospy.init_node('publisher_node 1')
        pub = rospy.Publisher("/robot_publisher_node",String, queue_size=10)
        rate = rospy.Rate(2)
        while not rospy.is_shutdown():
                msg = String()
                msg.data = "Hi i am publishing"
                pub.publish(msg)
                rate.sleep()
        rospy.loginfo("Publisher Node stopped")
```

Subscriber-1.py

```
#!/usr/bin/env python3
import rospy
from std_msgs.msg import String
def callback_receive_radio_data(msg):
        rospy.loginfo("Message received:")
        rospy.loginfo(msg)
if __name__ == '__main__':
        rospy.init_node('subscriber-1')
        sub = rospy.Subscriber("/robot_publisher_node", String,
callback_receive_radio_data)
        rospy.spin()
```

Subscriber-2.py

```
#!/usr/bin/env python3
import rospy
from std_msgs.msg import String
def callback_receive_radio_data(msg):
        rospy.loginfo("Message received:")
        rospy.loginfo(msg)
if __name__ == '__main__':
        rospy.init_node('subscriber-2')
```

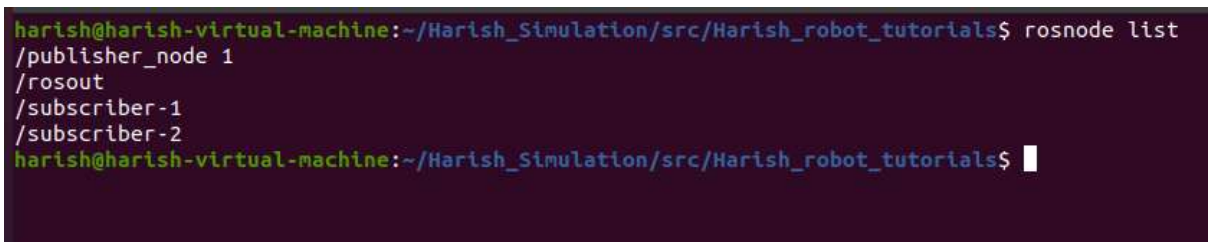
```
    sub = rospy.Subscriber("/robot_publisher_node", String,
callback_receive_radio_data)
    rospy.spin()
```

Screenshots of the terminal window

Subscriber - 1

Subscriber – 2

Rosnode list



2. **Create two publishers and two subscribers**

Code

Publisher – 1

```
#!/usr/bin/env python3
import rospy
from std_msgs.msg import String
if __name__ == '__main__':
        rospy.init_node('publisher_node 1')
        pub = rospy.Publisher("/robot_publisher_node",String, queue_size=10)
        rate = rospy.Rate(2)
        while not rospy.is_shutdown():
                msg = String()
                msg.data = "Hi i am publisher 1"
                pub.publish(msg)
                rate.sleep()
        rospy.loginfo("Publisher Node stopped")
```

Publisher – 2

```
#!/usr/bin/env python3
import rospy
from std_msgs.msg import String
if __name__ == '__main__':
        rospy.init_node('publisher_node 2')
        pub = rospy.Publisher("/robot_publisher_node",String, queue_size=10)
        rate = rospy.Rate(2)
        while not rospy.is_shutdown():
                msg = String()
                msg.data = "Hi i am publisher 2"
                pub.publish(msg)
                rate.sleep()
        rospy.loginfo("Publisher Node stopped")
```

Subscriber – 1

```
#!/usr/bin/env python3
import rospy
from std_msgs.msg import String
def callback_receive_radio_data(msg):
        rospy.loginfo("Message received:")
        rospy.loginfo(msg)


if __name__ == '__main__':
        rospy.init_node('subscriber')
        sub = rospy.Subscriber("/robot_publisher_node", String,
callback_receive_radio_data)
        rospy.spin()
```

Subscriber – 2

```
#!/usr/bin/env python3
import rospy
from std_msgs.msg import String
def callback_receive_radio_data(msg):
        rospy.loginfo("Message received and i am the second subscriber:")
        rospy.loginfo(msg)


if __name__ == '__main__':
        rospy.init_node('subscriber-2')
        sub = rospy.Subscriber("/robot_publisher_node", String,
callback_receive_radio_data)
        rospy.spin()
```

Screenshots of terminal window

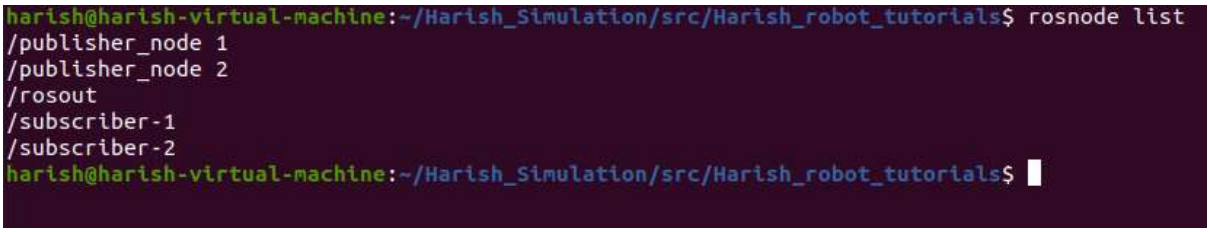
Publisher – 1

Publisher – 2

Subscriber – 1

Subscriber – 2

Rosnode list



3. Try with different data type

I. Integer 32(INT32)

Publisher.py

```
#!/usr/bin/env python3
import rospy
from std_msgs.msg import Int32
if __name__ == '__main__':
        rospy.init_node('publisher_node 1')
        pub = rospy.Publisher("/robot_publisher_node",Int32, queue_size=10)
        rate = rospy.Rate(2)
        while not rospy.is_shutdown():
                msg = Int32()
                msg.data = 11
                pub.publish(msg)
                rate.sleep()
        rospy.loginfo("Publisher Node stopped")
```

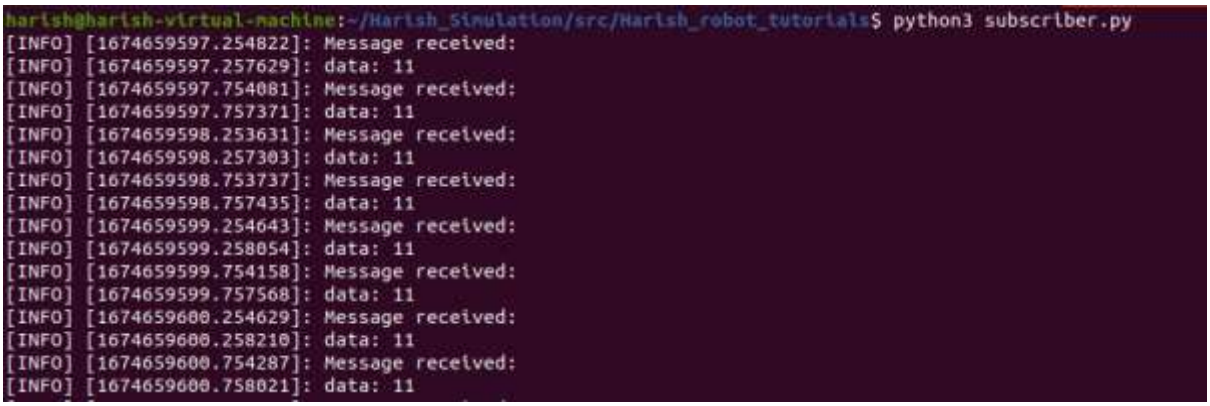
Subscriber.py

```
#!/usr/bin/env python3
import rospy
from std_msgs.msg import Int32
def callback_receive_radio_data(msg):
        rospy.loginfo("Message received:")
        rospy.loginfo(msg)

if __name__ == '__main__':

        rospy.init_node('subscriber')
        sub = rospy.Subscriber("/robot_publisher_node", Int32,
callback_receive_radio_data)
        rospy.spin()
```

Output



II. Time

Seconds and the nano seconds are published.

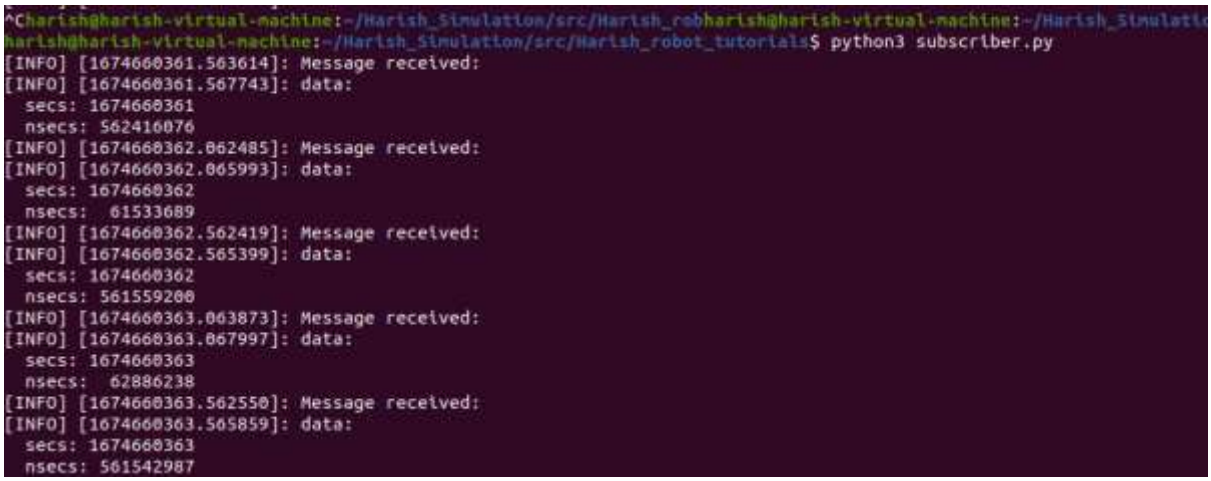
Publisher.py

```
#!/usr/bin/env python3
import rospy
from std_msgs.msg import Time
if __name__ == '__main__':
    rospy.init_node('publisher_node 1')
    pub = rospy.Publisher("/robot_publisher_node",Time, queue_size=10)
    rate = rospy.Rate(2)
    while not rospy.is_shutdown():
        msg = rospy.get_rostime()
        pub.publish(msg)
        rate.sleep()
    rospy.loginfo("Publisher Node stopped")
```

Subscriber.py

```
#!/usr/bin/env python3
import rospy
from std_msgs.msg import Time
def callback_receive_radio_data(msg):
    rospy.loginfo("Message received:")
    rospy.loginfo(msg)
if __name__ == '__main__':
    rospy.init_node('subscriber')
    sub = rospy.Subscriber("/robot_publisher_node", Time, callback_receive_radio_data)
    rospy.spin()
```

Output


^Charish@harish-virtual-machine:~/Harish_Simulation/src/Harish_robharish@harish-virtual-machine:~/Harish_Simulatio
harish@harish-virtual-machine:~/Harish_Simulation/src/Harish_robot_tutorials$ python3 subscriber.py
[INFO] [1674660361.563614]: Message received:
[INFO] [1674660361.567743]: data:
secs: 1674660361
nsecs: 562416076
[INFO] [1674660362.062485]: Message received:
[INFO] [1674660362.065993]: data:
secs: 1674660362
nsecs: 61533689
[INFO] [1674660362.562419]: Message received:
[INFO] [1674660362.565399]: data:
secs: 1674660362
nsecs: 561559200
[INFO] [1674660363.063873]: Message received:
[INFO] [1674660363.067997]: data:
secs: 1674660363
nsecs: 62886238
[INFO] [1674660363.562550]: Message received:
[INFO] [1674660363.565859]: data:
secs: 1674660363
nsecs: 561542987

4. **Check the rqt_graph**

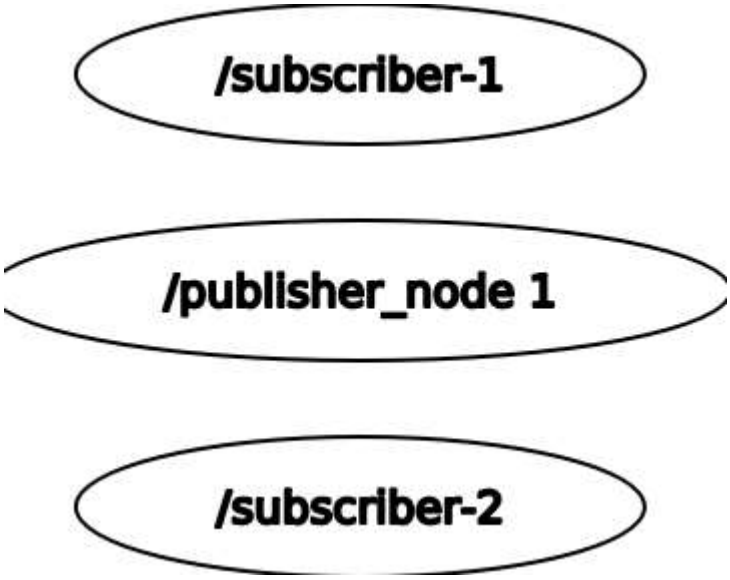
Code

```
harish@harish-virtual-machine:~/Harish_Simulation/src/Harish_robot_tutorials$ rosrun rqt_graph rqt_graph
```

How the ros graph appears in the window

For problem 1

Saved as a png file and took screenshot and pasted here



For problem 2

Saved as a png file and took screenshot and pasted here