Task 5:

* To create workspace:

$ mkdir -p ~/catkin\_ws/src

$ cd ~/catkin\_ws/src

$ catkin\_init\_workspace

$ cd ~/catkin\_ws/

$ catkin\_make

$ source devel/setup.bash

$ echo $ROS\_PACKAGE\_PATH

A screenshot of a cell phone

Description automatically generated

* To create a new package:

$ cd catkin\_ws/src

$ catkin\_create\_pkg <packge\_name> rospy roscpp std\_msgs

$ cd <package\_name>

$ mkdir scripts

$ cd scripts

A screenshot of a cell phone

Description automatically generated

script file contain talker and listener python file.

A screenshot of a computer

Description automatically generated

* To create publisher and subscriber nodes by running the python files using

$ roscore

$ chmod +x talker.py

$ python talker.py

$ chmod +x listener.py

$ python listener.py

A screenshot of a cell phone

Description automatically generated

Publisher code:

import rospy

from std\_msgs.msg import String

def talker():

pub = rospy.Publisher('chatter', String, queue\_size=10)

rospy.init\_node('talker', anonymous=True)

rate = rospy.Rate(10) # 10hz

while not rospy.is\_shutdown():

hello\_str = "hello world %s" % rospy.get\_time()

rospy.loginfo(hello\_str)

pub.publish(hello\_str)

rate.sleep()

if \_\_name\_\_ == '\_\_main\_\_':

try:

talker()

except rospy.ROSInterruptException:

pass

listener code:

import rospy

from std\_msgs.msg import String

def callback(data):

rospy.loginfo(rospy.get\_caller\_id() + 'I heard %s', data.data)

def listener():

# In ROS, nodes are uniquely named. If two nodes with the same

# name are launched, the previous one is kicked off. The

# anonymous=True flag means that rospy will choose a unique

# name for our 'listener' node so that multiple listeners can

# run simultaneously.

rospy.init\_node('listener', anonymous=True)

rospy.Subscriber('chatter', String, callback)

# spin() simply keeps python from exiting until this node is stopped

rospy.spin()

if \_\_name\_\_ == '\_\_main\_\_':

listener()