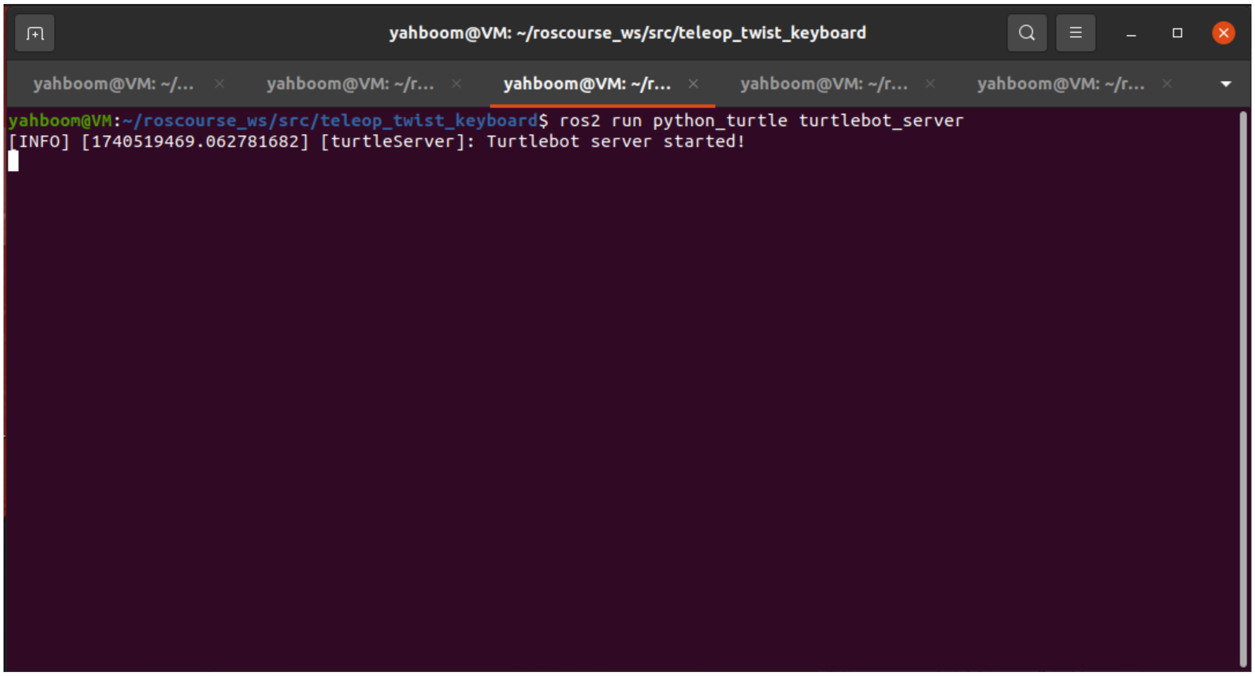
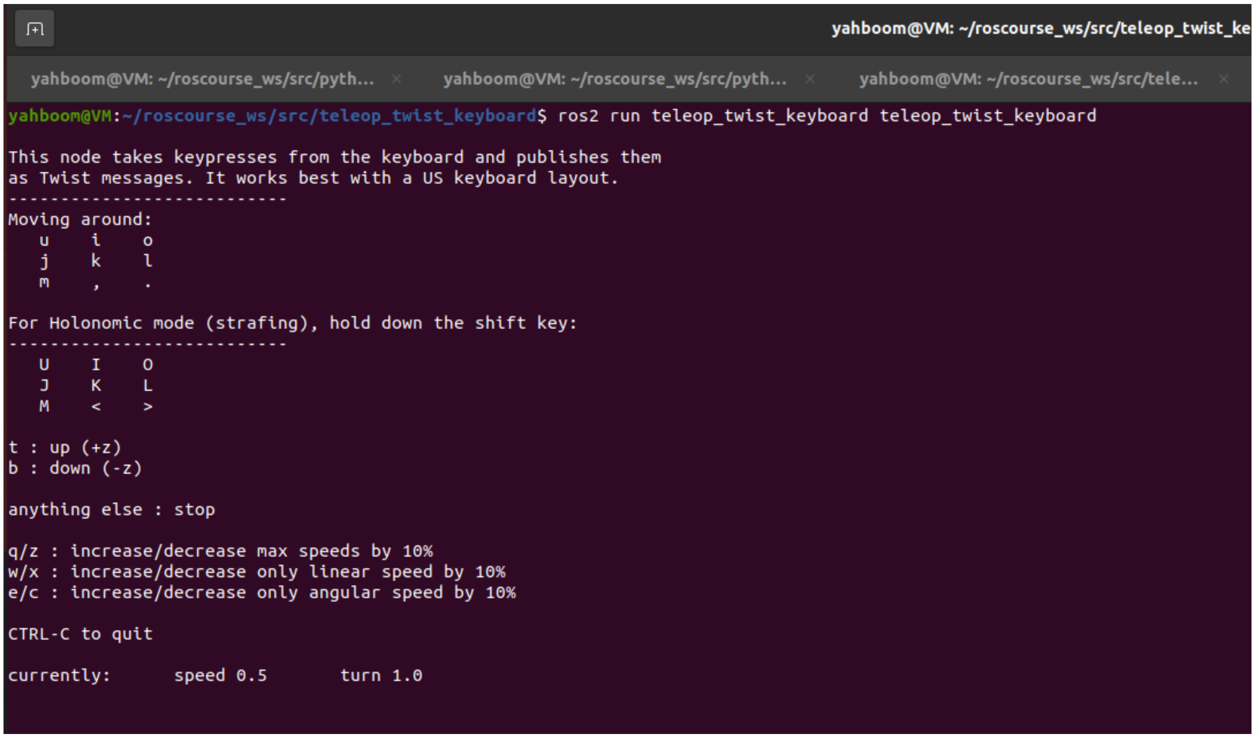
Task 1:



```
yahboom@VM:~/roscourse_ws/src/teleop_twist_keyboard$ ros2 run python_turtle turtlebot_server
[INFO] [1740519469.062781682] [turtleServer]: Turtlebot server started!
```

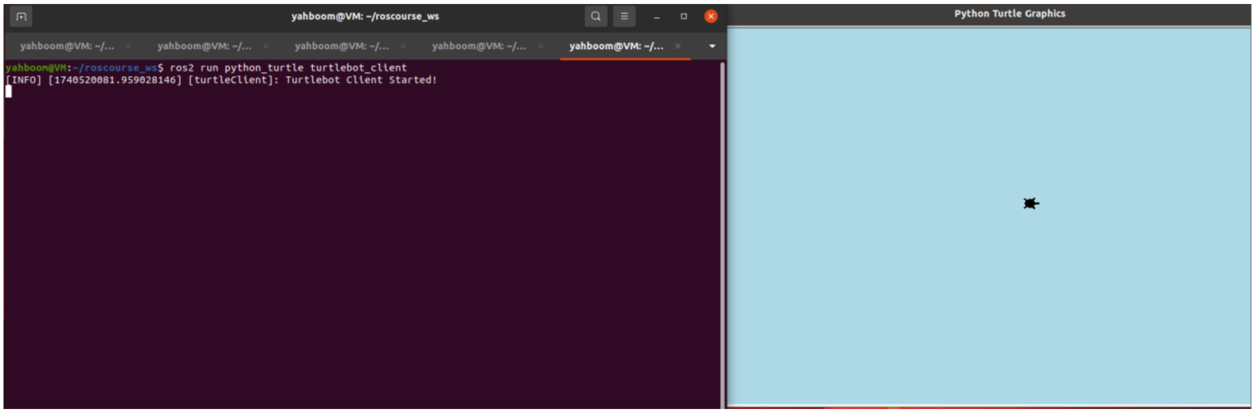
Figure 1: Starting the server



yahboom@VM: ~/roscourse_ws/src/teleop_twist_ke
yahboom@VM: ~/roscourse_ws/src/pyth...
yahboom@VM: ~/roscourse_ws/src/pyth...
yahboom@VM: ~/roscourse_ws/src/tele...

```
yahboom@VM:~/roscourse_ws/src/teleop_twist_keyboard$ ros2 run teleop_twist_keyboard teleop_twist_keyboard

This node takes keypresses from the keyboard and publishes them
as Twist messages. It works best with a US keyboard layout.
---------------------------
Moving around:
   u    i    o
   j    k    l
   m    ,    .

For Holonomic mode (strafing), hold down the shift key:
---------------------------
   U    I    O
   J    K    L
   M    <    >

t : up (+z)
b : down (-z)

anything else : stop

q/z : increase/decrease max speeds by 10%
w/x : increase/decrease only linear speed by 10%
e/c : increase/decrease only angular speed by 10%

CTRL-C to quit

currently:      speed 0.5       turn 1.0
```

Figure 2: Enabling the movement keys



```
yahboom@VM:~/roscourse_ws$ ros2 run python_turtle turtlebot_client
[INFO] [1740520081.959028146] [turtleClient]: Turtlebot Client Started!
```

Figure 3: Enabling the turtle and showing the output of the turtle

Task 2:

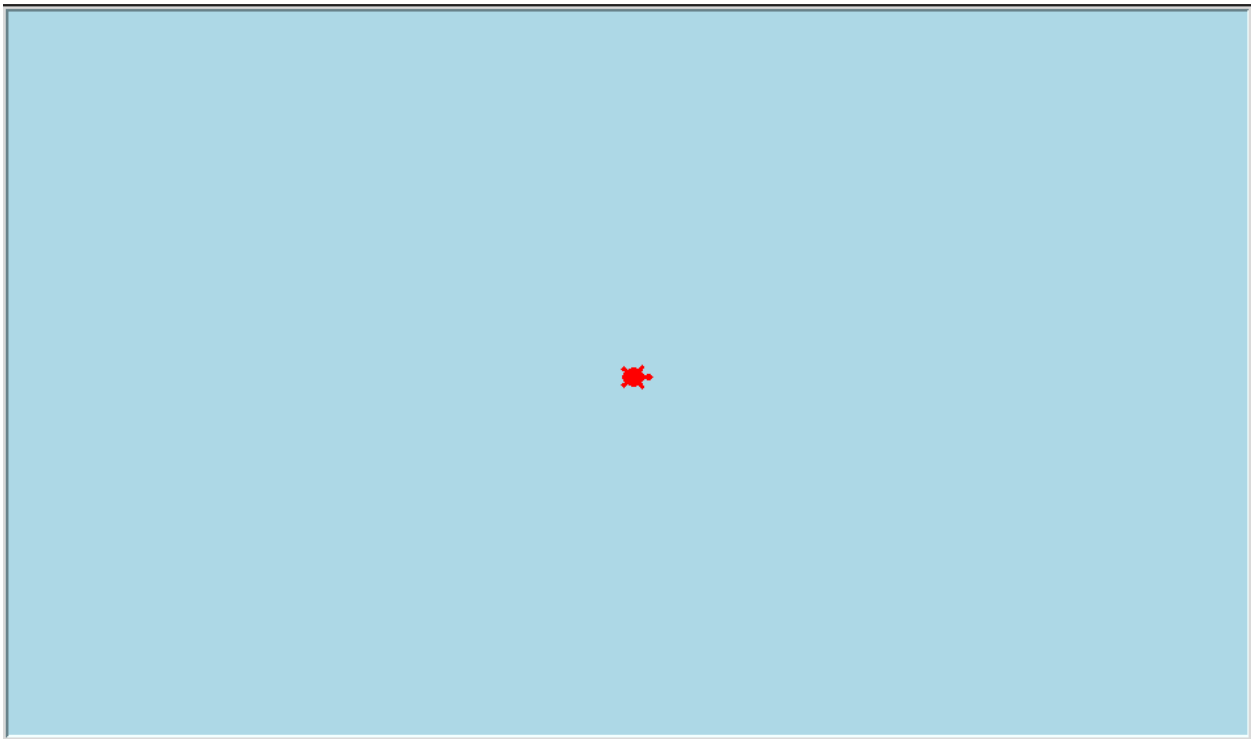
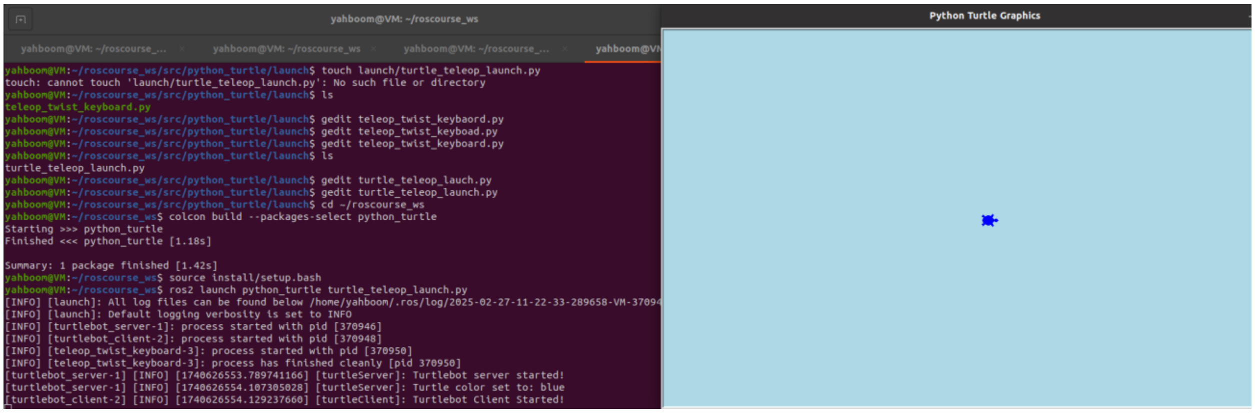


Figure 4: Turtle showing itself to be a different color after proper code is ran

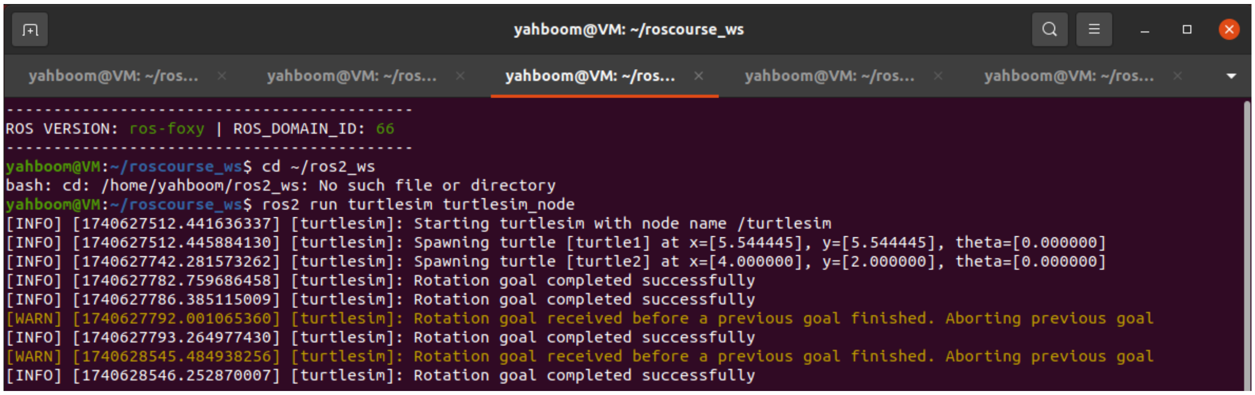
Task 3:


yahboom@VM: ~/roscourse_ws
yahboom@VM: ~/roscourse_...
yahboom@VM: ~/roscourse_ws
yahboom@VM: ~/roscourse_...
yahboom@VI
Python Turtle Graphics

```
yahboom@VM:~/roscourse_ws/src/python_turtle/launch$ touch launch/turtle_teleop_launch.py
touch: cannot touch 'launch/turtle_teleop_launch.py': No such file or directory
yahboom@VM:~/roscourse_ws/src/python_turtle/launch$ ls
teleop_twist_keyboard.py
yahboom@VM:~/roscourse_ws/src/python_turtle/launch$ gedit teleop_twist_keybaord.py
yahboom@VM:~/roscourse_ws/src/python_turtle/launch$ gedit teleop_twist_keyboad.py
yahboom@VM:~/roscourse_ws/src/python_turtle/launch$ gedit teleop_twist_keyboard.py
yahboom@VM:~/roscourse_ws/src/python_turtle/launch$ ls
turtle_teleop_launch.py
yahboom@VM:~/roscourse_ws/src/python_turtle/launch$ gedit turtle_teleop_lauch.py
yahboom@VM:~/roscourse_ws/src/python_turtle/launch$ gedit turtle_teleop_launch.py
yahboom@VM:~/roscourse_ws/src/python_turtle/launch$ cd ~/roscourse_ws
yahboom@VM:~/roscourse_ws$ colcon build --packages-select python_turtle
Starting >>> python_turtle
Finished <<< python_turtle [1.18s]

Summary: 1 package finished [1.42s]
yahboom@VM:~/roscourse_ws$ source install/setup.bash
yahboom@VM:~/roscourse_ws$ ros2 launch python_turtle turtle_teleop_launch.py
[INFO] [launch]: All log files can be found below /home/yahboom/.ros/log/2025-02-27-11-22-33-289658-VM-37094
[INFO] [launch]: Default logging verbosity is set to INFO
[INFO] [turtlebot_server-1]: process started with pid [370946]
[INFO] [turtlebot_client-2]: process started with pid [370948]
[INFO] [teleop_twist_keyboard-3]: process started with pid [370950]
[INFO] [teleop_twist_keyboard-3]: process has finished cleanly [pid 370950]
[turtlebot_server-1] [INFO] [1740626553.789741166] [turtleServer]: Turtlebot server started!
[turtlebot_server-1] [INFO] [1740626554.107305028] [turtleServer]: Turtle color set to: blue
[turtlebot_client-2] [INFO] [1740626554.129237660] [turtleClient]: Turtlebot Client Started!
```

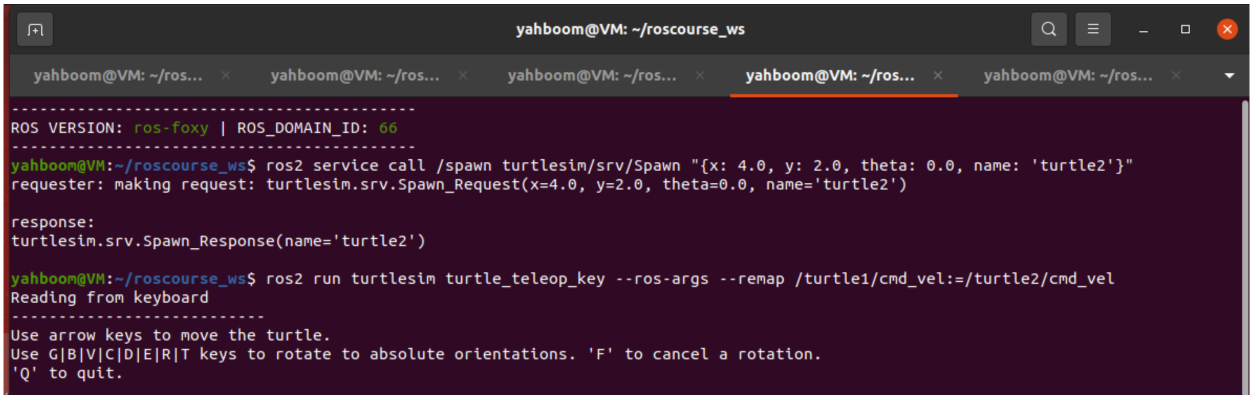
Figure 5: Terminal output for step 8

Task 4:



```
-------------------------------------------
ROS VERSION: ros-foxy | ROS_DOMAIN_ID: 66
-------------------------------------------
yahboom@VM:~/roscourse_ws$ cd ~/ros2_ws
bash: cd: /home/yahboom/ros2_ws: No such file or directory
yahboom@VM:~/roscourse_ws$ ros2 run turtlesim turtlesim_node
[INFO] [1740627512.441636337] [turtlesim]: Starting turtlesim with node name /turtlesim
[INFO] [1740627512.445884130] [turtlesim]: Spawning turtle [turtle1] at x=[5.544445], y=[5.544445], theta=[0.000000]
[INFO] [1740627742.281573262] [turtlesim]: Spawning turtle [turtle2] at x=[4.000000], y=[2.000000], theta=[0.000000]
[INFO] [1740627782.759686458] [turtlesim]: Rotation goal completed successfully
[INFO] [1740627786.385115009] [turtlesim]: Rotation goal completed successfully
[WARN] [1740627792.001065360] [turtlesim]: Rotation goal received before a previous goal finished. Aborting previous goal
[INFO] [1740627793.264977430] [turtlesim]: Rotation goal completed successfully
[WARN] [1740628545.484938256] [turtlesim]: Rotation goal received before a previous goal finished. Aborting previous goal
[INFO] [1740628546.252870007] [turtlesim]: Rotation goal completed successfully
```

Figure 6: Starting the turtle



```
-------------------------------------------
ROS VERSION: ros-foxy | ROS_DOMAIN_ID: 66
-------------------------------------------
yahboom@VM:~/roscourse_ws$ ros2 service call /spawn turtlesim/srv/Spawn "{x: 4.0, y: 2.0, theta: 0.0, name: 'turtle2'}"
requester: making request: turtlesim.srv.Spawn_Request(x=4.0, y=2.0, theta=0.0, name='turtle2')

response:
turtlesim.srv.Spawn_Response(name='turtle2')

yahboom@VM:~/roscourse_ws$ ros2 run turtlesim turtle_teleop_key --ros-args --remap /turtle1/cmd_vel:=/turtle2/cmd_vel
Reading from keyboard
---------------------------
Use arrow keys to move the turtle.
Use G|B|V|C|D|E|R|T keys to rotate to absolute orientations. 'F' to cancel a rotation.
'Q' to quit.
```

Figure 7: Enabling the keyboard inputs

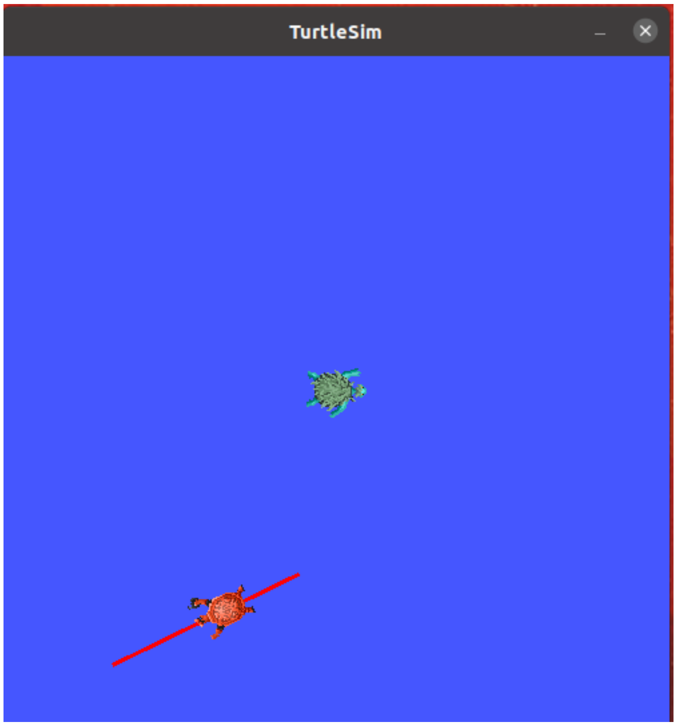


Figure 8: Two turtles with different pen colors