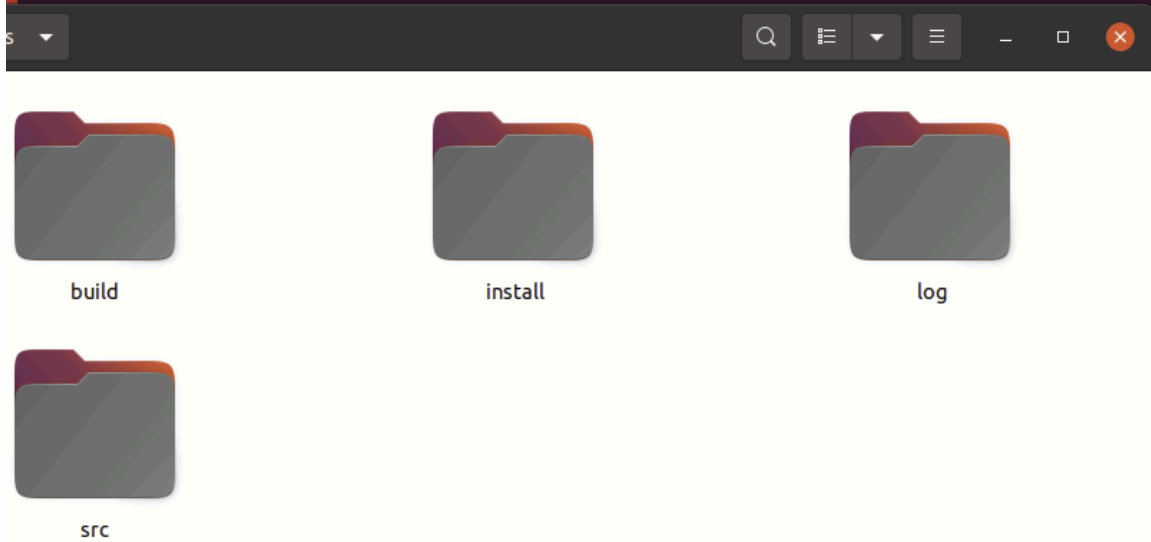


Task 2 Step 8

```
yahboom@VM:~/roscourse_ws$ colcon build --symlink-install
Starting >>> webcam
Finished <<< webcam [2.27s]

Summary: 1 package finished [2.88s]
yahboom@VM:~/roscourse_ws$
```



Task 3 Step 22

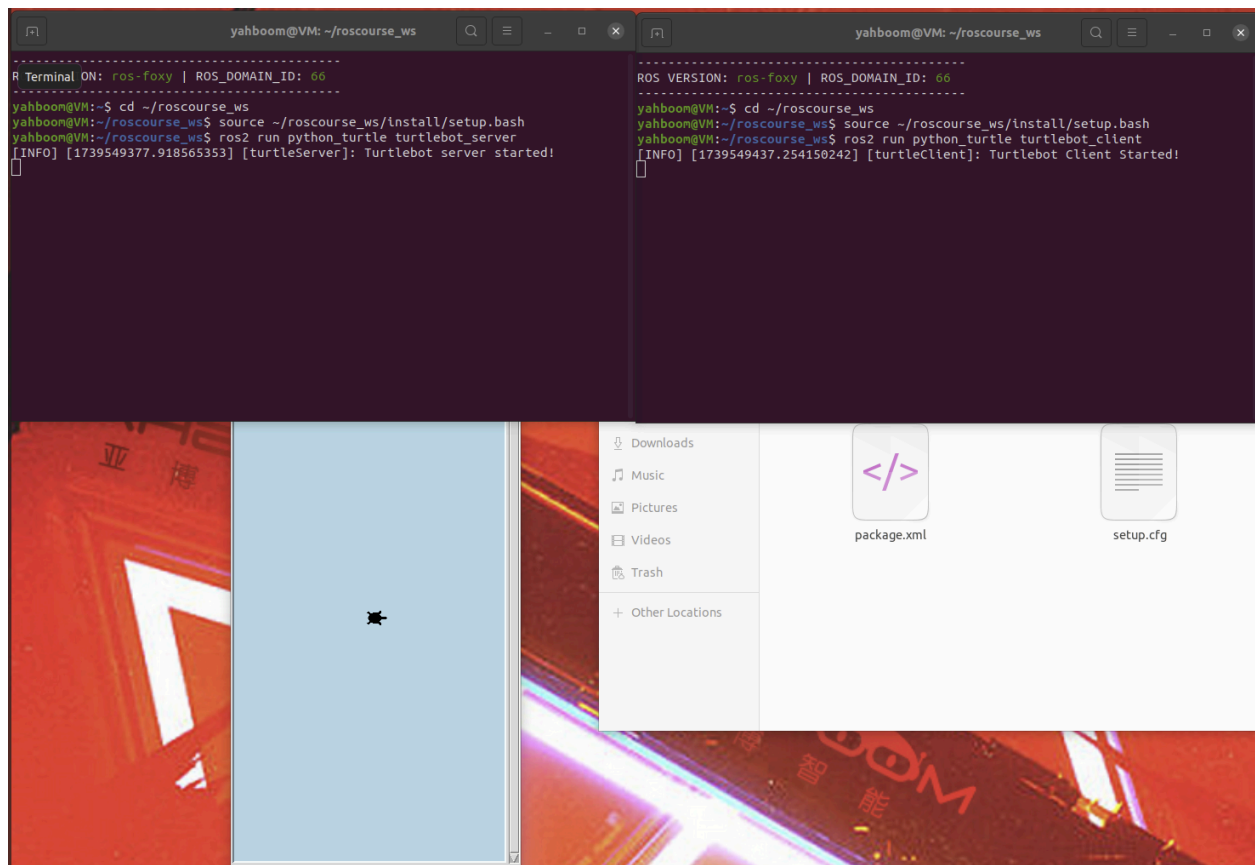
```
yahboom@VM:~/roscourse_ws$ ros2 interface show turtle_interfaces/msg/TurtleMsg
string name
geometry_msgs/Pose turtle_pose
string color
yahboom@VM:~/roscourse_ws$ ros2 interface show turtle_interfaces/srv/SetPose
geometry_msgs/PoseStamped turtle_pose
---
int8 ret
yahboom@VM:~/roscourse_ws$ ros2 interface show turtle_interfaces/srv/SetColor
string color
---
int8 ret
yahboom@VM:~/roscourse_ws$
```

Required Question:

1. Uses Python's turtle module to visualize the turtle's movement.
2. Initializes a TurtleMsg object to store the turtle's state.
3. Publishes velocity commands (Twist messages) to the turtleDrive topic.
4. The server node listens to this topic and moves the turtle accordingly.
5. Subscribes to turtleState to get real-time updates about the turtle's position and orientation.
6. Triggers turtle_callback whenever new data arrives.

1. Creates a TurtleMsg object to store the turtle's current position and orientation.
2. Publishes the turtle's updated state to turtleState.
3. Calculates the new position (x, y) based on velocity and time.
4. Updates the turtle's new orientation.

Task 4 step 7



Task 4 step 16

