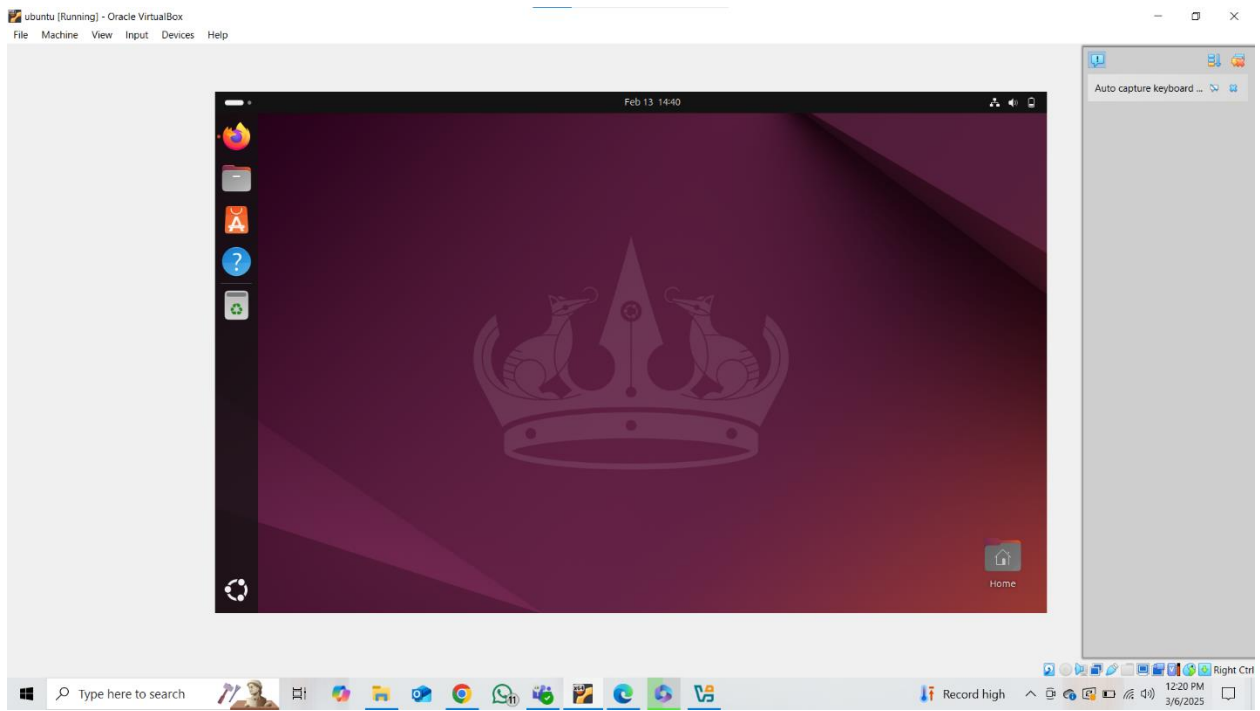
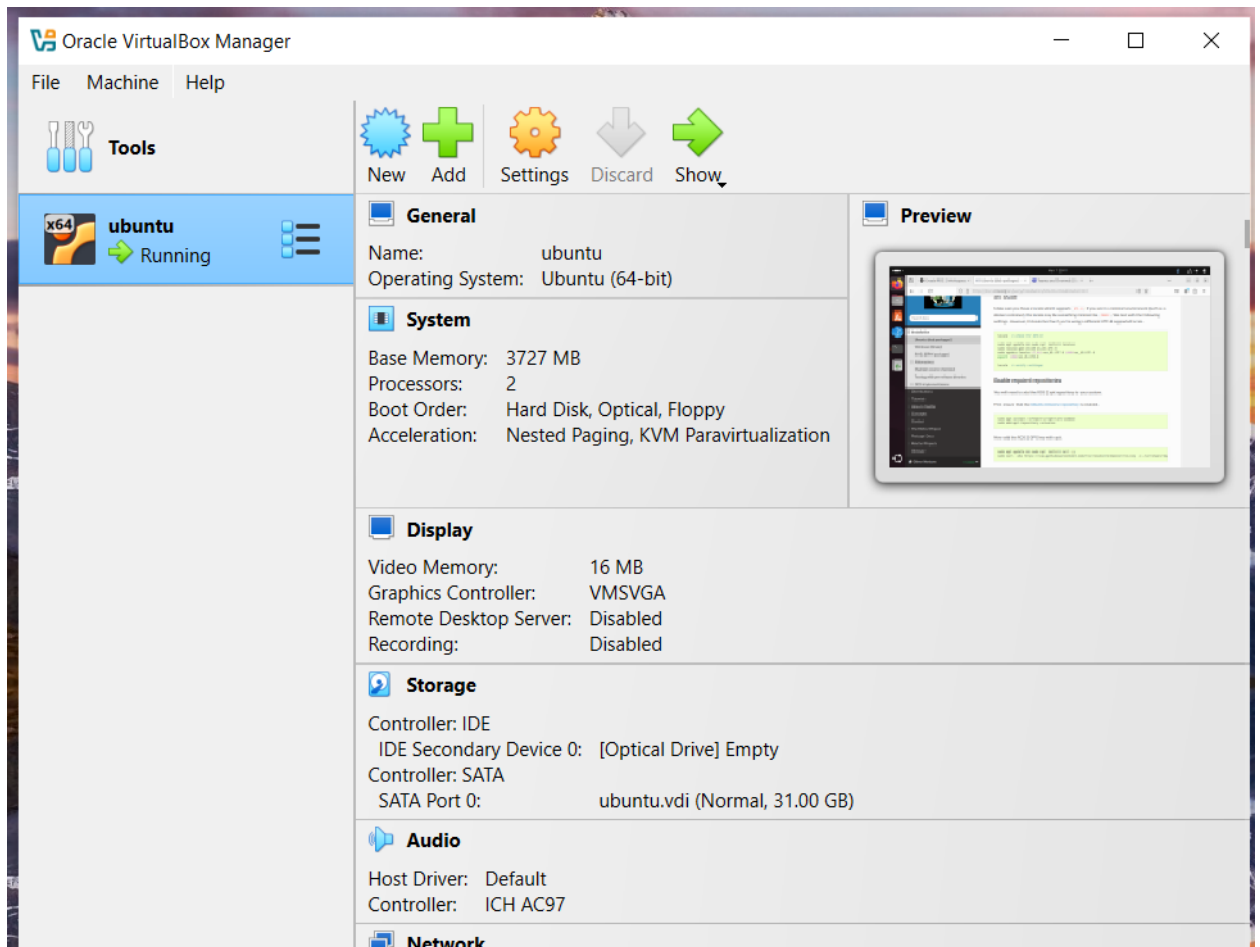


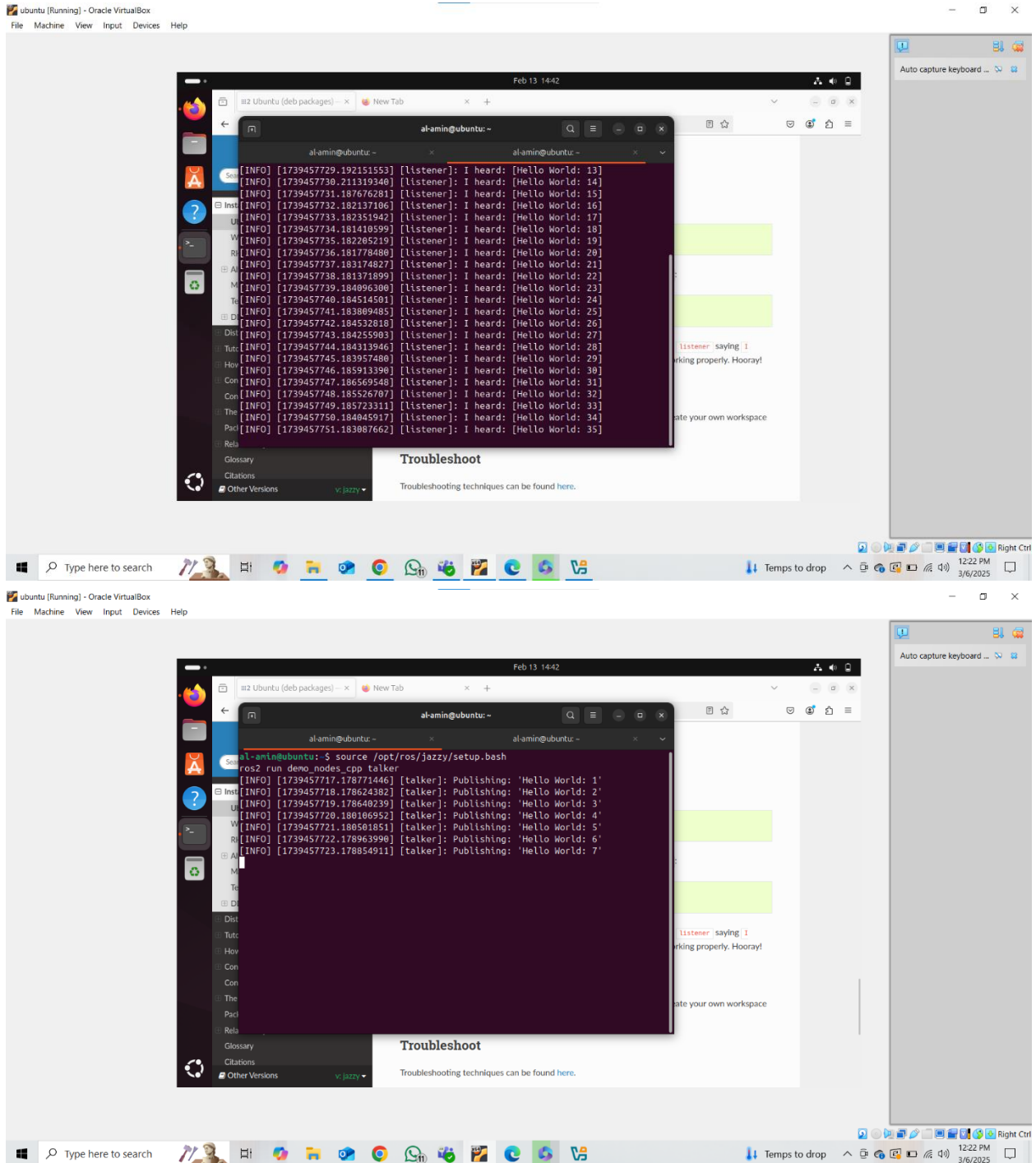
## **Lab work-1**

During this assignment, I will follow a systematic approach to understand and implement a basic ROS2 communication setup.

1. **Environment Setup:** We installed Ubuntu (in VirtualBox or WSL) and configured ROS2 Jazzy.

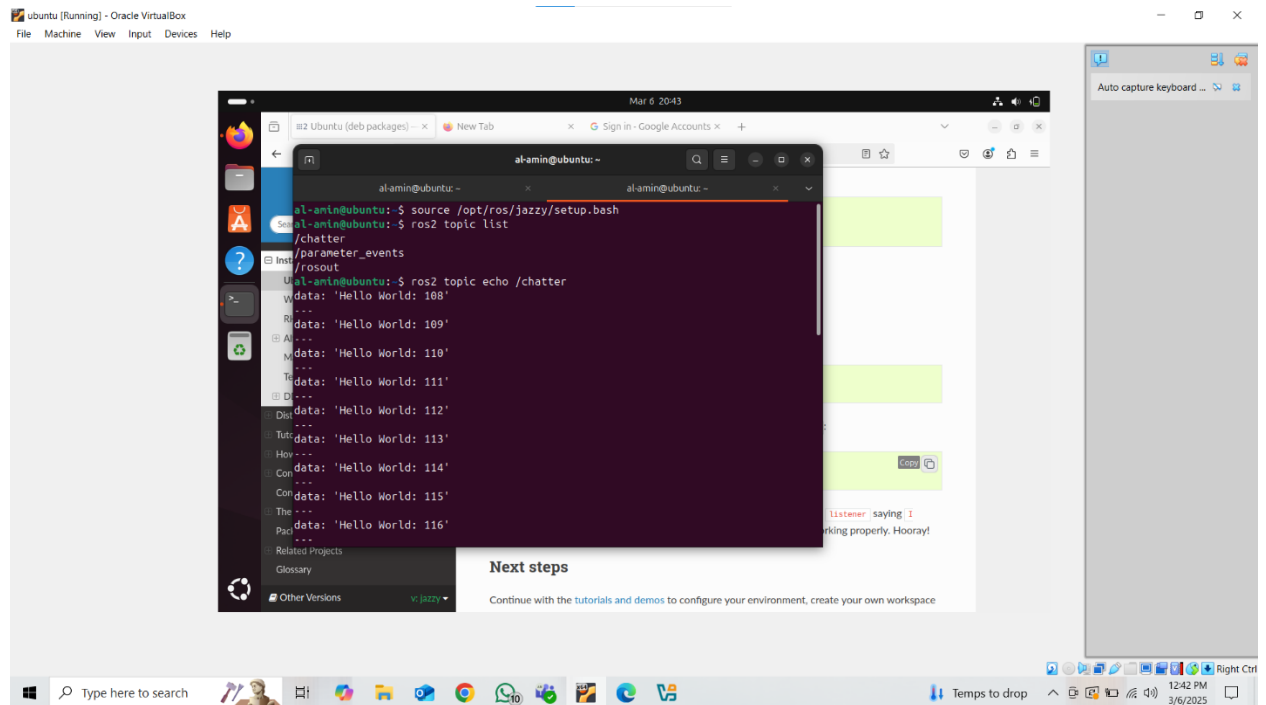


## 2. Running Default ROS2 Nodes: i successfully executed the built-in Talker-Listener example, verifying that ROS2 was functioning correctly.



## 3. Inspecting Topics and Messages: Using ROS2 commands (ros2 topic list, ros2 topic echo), i explored how nodes

communicate through topics.



- ```
al-amin@ubuntu:~$ ros2 topic info /chatter
Type: std_msgs/msg/String
Publisher count: 1
Subscription count: 1
al-amin@ubuntu:~$
```
4. ted Projects
- ```
al-amin@ubuntu:~$ ros2 topic type /chatter
std_msgs/msg/String
```
5. al-amin@ubuntu:~\$
- ```
std_msgs/msg/String
al-amin@ubuntu:~$ ros2 node list
/listener
/talker
al-amin@ubuntu:~$
```
6. ted Projects

## 7. Modifying the Example:

- Created a new publisher (custom\_talker) that sends messages to custom\_topic.
- Added two subscriber nodes (custom\_listener and extra\_listener) that receive messages from custom\_topic.

- Changed the publishing interval to 2 seconds to control message frequency.

```

al-amin@ubuntu: ~/ros2_ws/src      al-amin@ubuntu: ~/ros2_ws/src/custom_no...
GNU nano 7.2      custom_nodes/custom_talker.py *
import rclpy
from rclpy.node import Node
from std_msgs.msg import String

class CustomTalker(Node):
    def __init__(self):
        super().__init__('custom_talker')
        self.publisher_ = self.create_publisher(String, 'custom_topic', 10)
        self.timer = self.create_timer(1.0, self.publish_message) # 1-second
        self.counter = 0

    def publish_message(self):
        msg = String()
        msg.data = f"Hello from CustomTalker: {self.counter}"
        self.publisher_.publish(msg)
        self.get_logger().info(f'Publishing: {msg.data}')
        self.counter += 1

def main(args=None):
    rclpy.init(args=args)

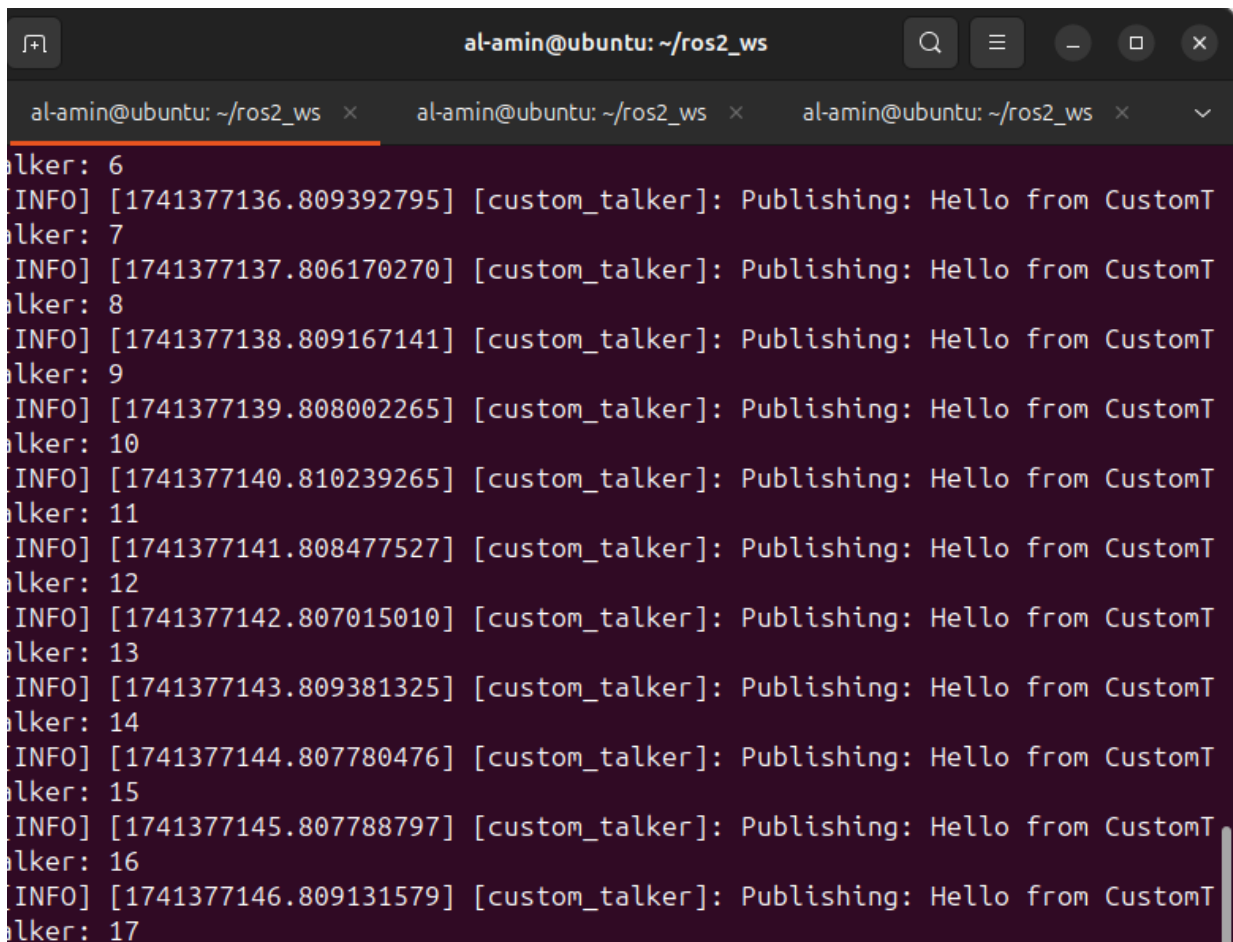
File Name to Write: custom_nodes/custom_talker.py
^G Help      M-D DOS Format      M-A Append      M-B Backup File
^C Cancel    M-M Mac Format      M-P Prepend     ^T Browse

al-amin@ubuntu:~/ros2_ws/src/custom_nodes$ chmod +x custom_nodes/custom_talker.py
al-amin@ubuntu:~/ros2_ws/src/custom_nodes$ chmod +x custom_nodes/custom_listener.py
al-amin@ubuntu:~/ros2_ws/src/custom_nodes$ chmod +x custom_nodes/extra_listener.py
al-amin@ubuntu:~/ros2_ws/src/custom_nodes$

```

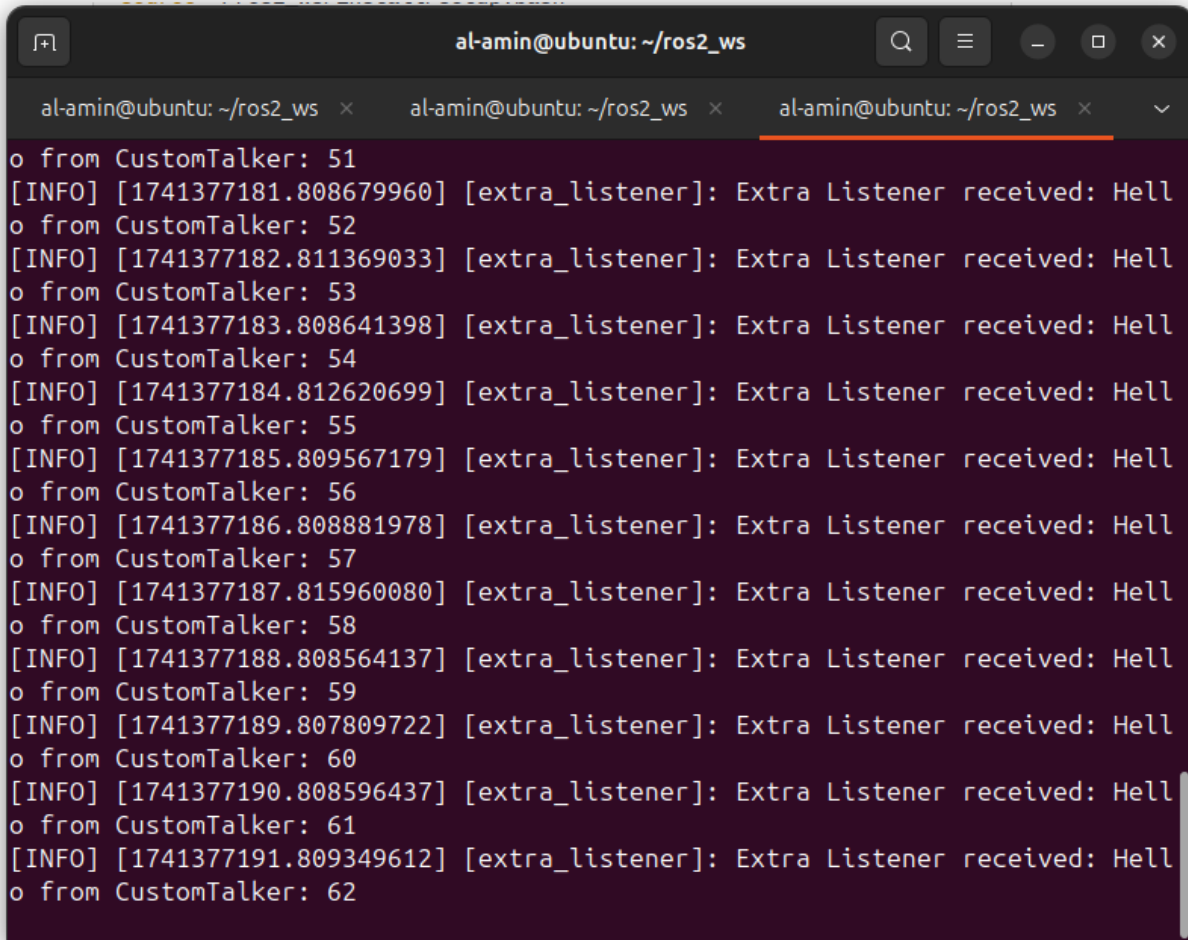
```
al-amin@ubuntu:~/ros2_ws/src/custom_nodes$ cd ~/ros2_ws
colcon build
source install/setup.bash
Starting >>> custom_nodes
Finished <<< custom_nodes [3.79s]

Summary: 1 package finished [4.33s]
al-amin@ubuntu:~/ros2_ws$
```

A terminal window titled "al-amin@ubuntu: ~/ros2\_ws" with three tabs. The active tab shows the output of a custom\_talker node. The output consists of 12 lines, each starting with a timestamp in brackets, followed by "[custom\_talker]: Publishing: Hello from CustomT". The timestamps range from 1741377136.809392795 to 1741377146.809131579. The terminal window has a dark background and a light-colored border.

```
al-amin@ubuntu: ~/ros2_ws
al-amin@ubuntu: ~/ros2_ws x al-amin@ubuntu: ~/ros2_ws x al-amin@ubuntu: ~/ros2_ws x
talker: 6
[INFO] [1741377136.809392795] [custom_talker]: Publishing: Hello from CustomT
talker: 7
[INFO] [1741377137.806170270] [custom_talker]: Publishing: Hello from CustomT
talker: 8
[INFO] [1741377138.809167141] [custom_talker]: Publishing: Hello from CustomT
talker: 9
[INFO] [1741377139.808002265] [custom_talker]: Publishing: Hello from CustomT
talker: 10
[INFO] [1741377140.810239265] [custom_talker]: Publishing: Hello from CustomT
talker: 11
[INFO] [1741377141.808477527] [custom_talker]: Publishing: Hello from CustomT
talker: 12
[INFO] [1741377142.807015010] [custom_talker]: Publishing: Hello from CustomT
talker: 13
[INFO] [1741377143.809381325] [custom_talker]: Publishing: Hello from CustomT
talker: 14
[INFO] [1741377144.807780476] [custom_talker]: Publishing: Hello from CustomT
talker: 15
[INFO] [1741377145.807788797] [custom_talker]: Publishing: Hello from CustomT
talker: 16
[INFO] [1741377146.809131579] [custom_talker]: Publishing: Hello from CustomT
talker: 17
```

```
al-amin@ubuntu: ~/ros2_ws
al-amin@ubuntu: ~/ros2_ws x al-amin@ubuntu: ~/ros2_ws x al-amin@ubuntu: ~/ros2_ws x
customTalker: 21
[INFO] [1741377151.806894725] [custom_listener]: Listener heard: Hello from C
customTalker: 22
[INFO] [1741377152.808624027] [custom_listener]: Listener heard: Hello from C
customTalker: 23
[INFO] [1741377153.808881715] [custom_listener]: Listener heard: Hello from C
customTalker: 24
[INFO] [1741377154.810480802] [custom_listener]: Listener heard: Hello from C
customTalker: 25
[INFO] [1741377155.809245095] [custom_listener]: Listener heard: Hello from C
customTalker: 26
[INFO] [1741377156.811866081] [custom_listener]: Listener heard: Hello from C
customTalker: 27
[INFO] [1741377157.808398287] [custom_listener]: Listener heard: Hello from C
customTalker: 28
[INFO] [1741377158.808642445] [custom_listener]: Listener heard: Hello from C
customTalker: 29
[INFO] [1741377159.807919579] [custom_listener]: Listener heard: Hello from C
customTalker: 30
[INFO] [1741377160.810183961] [custom_listener]: Listener heard: Hello from C
customTalker: 31
[INFO] [1741377161.807133175] [custom_listener]: Listener heard: Hello from C
customTalker: 32
```

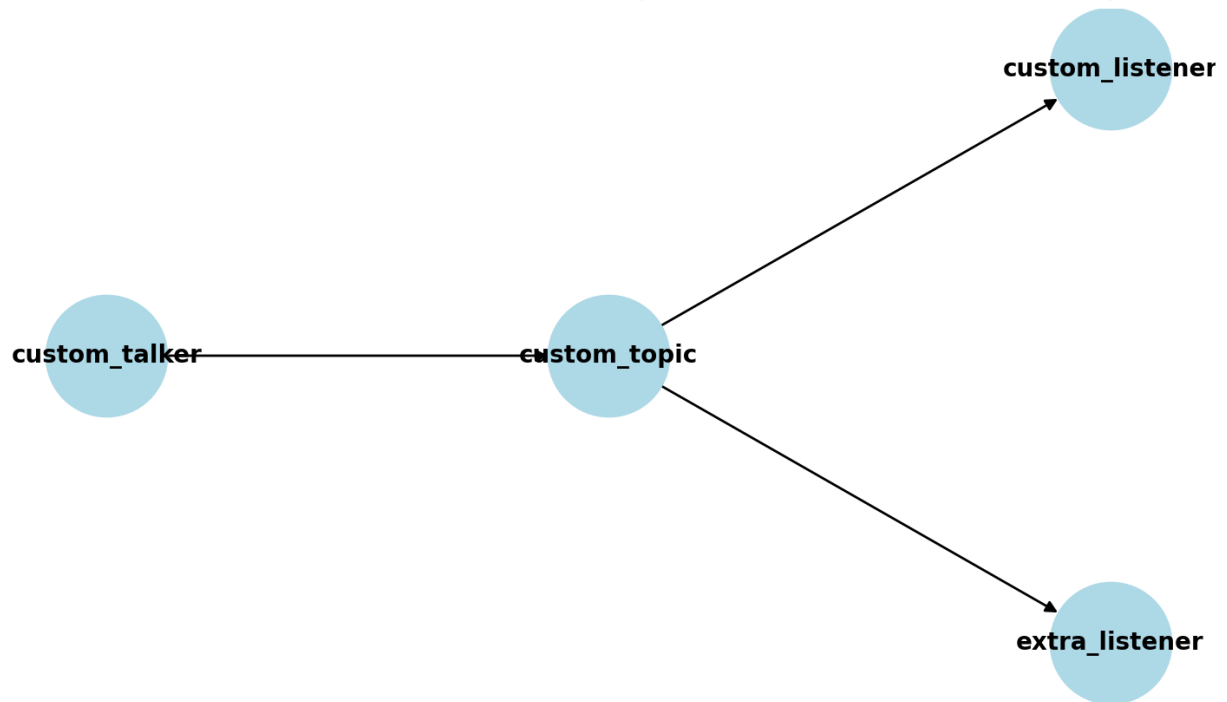
A terminal window titled 'al-amin@ubuntu: ~/ros2\_ws' with three tabs. The active tab shows a series of log messages. Each message consists of a line starting with 'o from CustomTalker: ' followed by a number (51 to 62), and a second line starting with '[INFO] [timestamp] [extra\_listener]: Extra Listener received: Hell'. The timestamps are in the format [1741377181.808679960] to [1741377191.809349612].

```
al-amin@ubuntu: ~/ros2_ws x al-amin@ubuntu: ~/ros2_ws x al-amin@ubuntu: ~/ros2_ws x
o from CustomTalker: 51
[INFO] [1741377181.808679960] [extra_listener]: Extra Listener received: Hell
o from CustomTalker: 52
[INFO] [1741377182.811369033] [extra_listener]: Extra Listener received: Hell
o from CustomTalker: 53
[INFO] [1741377183.808641398] [extra_listener]: Extra Listener received: Hell
o from CustomTalker: 54
[INFO] [1741377184.812620699] [extra_listener]: Extra Listener received: Hell
o from CustomTalker: 55
[INFO] [1741377185.809567179] [extra_listener]: Extra Listener received: Hell
o from CustomTalker: 56
[INFO] [1741377186.808881978] [extra_listener]: Extra Listener received: Hell
o from CustomTalker: 57
[INFO] [1741377187.815960080] [extra_listener]: Extra Listener received: Hell
o from CustomTalker: 58
[INFO] [1741377188.808564137] [extra_listener]: Extra Listener received: Hell
o from CustomTalker: 59
[INFO] [1741377189.807809722] [extra_listener]: Extra Listener received: Hell
o from CustomTalker: 60
[INFO] [1741377190.808596437] [extra_listener]: Extra Listener received: Hell
o from CustomTalker: 61
[INFO] [1741377191.809349612] [extra_listener]: Extra Listener received: Hell
o from CustomTalker: 62
```

**8. Visualizing Data Flow:** i created a diagram to clearly represent how information flows between nodes via topics.



## ROS2 Communication Flow (Custom Talker-Listeners)



### Final Thought:

This project provided me a foundational understanding of ROS2 communication, demonstrating how nodes exchange information through topics in a modular and scalable system. The successful implementation of a multi-subscriber model highlights the flexibility of ROS2 in real-world robotic applications. (used little help from chat-gpt)....