

Omni-Wheel Robot - Interactive Challenge Notebook

This notebook will guide you through controlling an **omni-wheel robot** step by step. You'll see an **example**, then try to complete a similar challenge on your own!

Learning Objectives

- Understand how to move the robot using Python commands.
- Experiment with different speeds, directions, and durations.
- Observe and adjust movement patterns to improve control.

```
In [ ]: import sys
import os

# Add parent directory to the Python path
sys.path.insert(0, os.path.abspath('../'))

import rclpy
from controllers.omni_robot_controller import OmniWheelControlNode # Import

# Initialize ROS2 node
rclpy.init()
node = OmniWheelControlNode()
```

Example 1: Moving Forward

Here's how you can **move the robot forward** at a slow speed for 2 seconds. Try running this command and observe the behavior.

```
In [ ]: # Example: Move forward
node.move_in_direction(0, 0.5, 2)
```

Challenge 1: Moving Backward

Now it's your turn! Write a command that moves the robot **backward** at the same speed for 3 seconds. Fill in the missing values below:

```
In [ ]: # Your code here:
# node.move_in_direction(?, ?, ?)
```

Example 2: Rotating the Robot

Here's how you can make the robot **rotate counterclockwise** for 1.5 seconds.

```
In [ ]: # Example: Rotate left
node.rotate_left(30, 1.5)
```

Challenge 2: Rotating Clockwise

Write a command to **rotate the robot clockwise** at the same speed for 2 seconds. Test it and observe the result!

```
In [ ]: # Your code here:
# node.rotate_right(?, ?)
```

Example 3: Moving in a Square

Robots can follow **patterns** by combining movements. Here's an example of moving the robot **in a square pattern** (2m per side).

```
In [ ]: # Example: Move in a square
for _ in range(4):
    node.move_in_direction(0, 0.5, 2)
    node.rotate_right(90, 1)
```

Challenge 3: Moving in a Triangle

Write a command to move the robot **in a triangle pattern** instead of a square.

- Each side should be **3 meters** long.
- The robot should rotate **120 degrees** at each turn. Try to complete the loop below!

```
In [ ]: # Your code here:
# for _ in range(3):
#     node.move_in_direction(?, ?, ?)
#     node.rotate_right(?, ?)
```

Final Challenge: Experiment with Speed

Now, **experiment!**

1. Try increasing the **speed** to 1.0 and see what happens.
2. Try decreasing the **duration** to 1 second. How does the robot's behavior change?

```
In [ ]: # Your code here:  
        # node.move_in_direction(?, ?, ?)
```

Shutting Down the Node

Once you're done, **shutdown the node** properly.

```
In [ ]: node.destroy_node()  
        rclpy.shutdown()
```