

robot_ws\src\assign2_650610841\assign2_650610841\master_650610841.py

```
1 import sys
2
3 import rclpy
4 from rclpy.node import Node
5 from myword_650610841.srv import SumFourInt
6 from myword_650610841.msg import Num
7
8
9 class MinimalClientAsync(Node):
10
11     def __init__(self):
12         super().__init__('master_650610841')
13         self.cli = self.create_client(SumFourInt, 'sum_four_ints')
14         while not self.cli.wait_for_service(timeout_sec=1.0):
15             self.get_logger().info('service not available, waiting again...')
16         self.req = SumFourInt.Request()
17
18         self.publisher_ = self.create_publisher(Num, 'gossip_650610841', 10)
19
20     def send_request(self, a, b, c, d):
21         self.req.a = a
22         self.req.b = b
23         self.req.c = c
24         self.req.d = d
25         return self.cli.call_async(self.req)
26
27 def main():
28     rclpy.init()
29
30     minimal_client = MinimalClientAsync()
31     future = minimal_client.send_request(int(sys.argv[1]), int(sys.argv[2]), int(sys.argv[3]),
32 int(sys.argv[4]))
33     rclpy.spin_until_future_complete(minimal_client, future)
34     response = future.result()
35     minimal_client.get_logger().info(
36         'Ahh! Result is %d + %d + %d + %d = %d' %
37         (int(sys.argv[1]), int(sys.argv[2]), int(sys.argv[3]), int(sys.argv[4]), response.i))
38     msg = Num()
39     msg.n = response.i
40     minimal_client.publisher_.publish(msg)
41     minimal_client.get_logger().info(
42         'Shouting: result is %d' % response.i)
43
44     minimal_client.destroy_node()
45     rclpy.shutdown()
46
47 if __name__ == '__main__':
```

```
48 |         main()
```