Exercise - 2

21_AIE_213

Robotics operating System– SEM-IV

Professor – Dr. Navneeth

Submitted By: Vikhyat Bansal [CB.EN.U4AIE21076]

Q2. Code of number_counter_reset

```
import rclpy
from rclpy.node import Node
from example_interfaces.srv import SetBool

class ResetCounterClient(Node):

    def __init__(self):
        super().__init__("reset_counter_client")
        self.reset_counter_client_ = self.create_client(SetBool,
"reset_count")
        while self.reset_counter_client_.wait_for_service(0.25) ==
False:
            self.get_logger().warn("Waiting for reset_counter service")
        self.get_logger().info("Reset Counter service is available")
        self.reset_counter_request_ = SetBool.Request()
        self.reset_counter_request_.data = True # Change this to False
for letting the counter continue the count.
        self.reset_counter_future_ =
self.reset_counter_client_.call_async(self.reset_counter_request_)

self.reset_counter_future_.add_done_callback(self.callback_reset_counte
r)

    def callback_reset_counter(self, future):
        response = future.result()
        self.get_logger().info(f"Response: {response.message}")

def main(args=None):
    rclpy.init(args=args)
    #Code goes here
    node = ResetCounterClient()
    rclpy.spin(node)
    rclpy.shutdown()

if __name__ == "__main__":
    main()
```