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\_counter)

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()