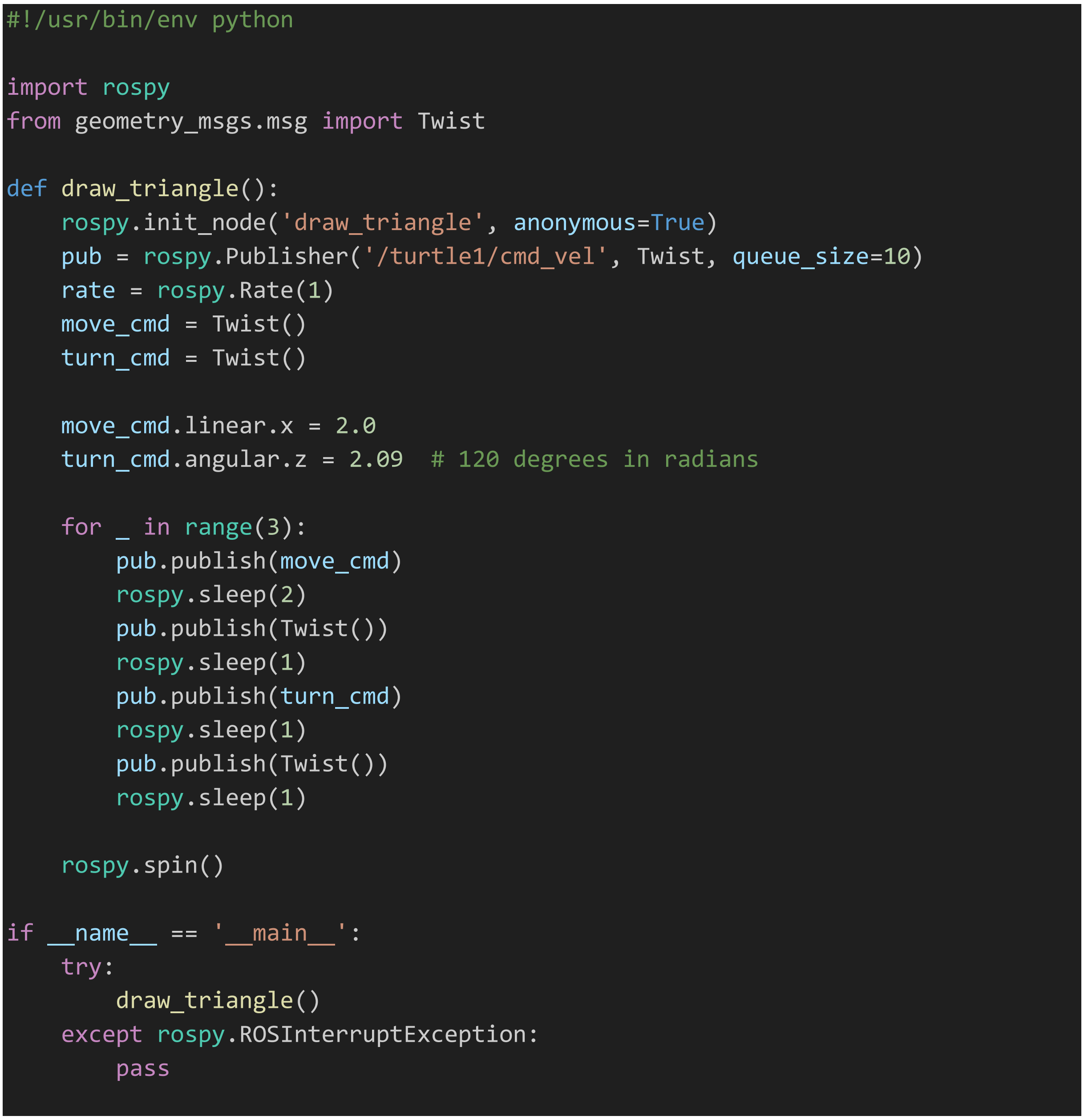
ROS2 TurtleSim Shapes

Square:

```
#!/usr/bin/env python

import rospy
from geometry_msgs.msg import Twist

def draw_square():
    rospy.init_node('draw_square', anonymous=True)
    pub = rospy.Publisher('/turtle1/cmd_vel', Twist, queue_size=10)
    rate = rospy.Rate(1)
    move_cmd = Twist()
    turn_cmd = Twist()

    move_cmd.linear.x = 2.0
    turn_cmd.angular.z = 1.57

    for _ in range(4):
        pub.publish(move_cmd)
        rospy.sleep(2)
        pub.publish(Twist())
        rospy.sleep(1)
        pub.publish(turn_cmd)
        rospy.sleep(1)
        pub.publish(Twist())
        rospy.sleep(1)

    rospy.spin()

if __name__ == '__main__':
    try:
        draw_square()
    except rospy.ROSInterruptException:
        pass
```

Triangle:



```python
#!/usr/bin/env python

import rospy
from geometry_msgs.msg import Twist

def draw_triangle():
    rospy.init_node('draw_triangle', anonymous=True)
    pub = rospy.Publisher('/turtle1/cmd_vel', Twist, queue_size=10)
    rate = rospy.Rate(1)
    move_cmd = Twist()
    turn_cmd = Twist()

    move_cmd.linear.x = 2.0
    turn_cmd.angular.z = 2.09  # 120 degrees in radians

    for _ in range(3):
        pub.publish(move_cmd)
        rospy.sleep(2)
        pub.publish(Twist())
        rospy.sleep(1)
        pub.publish(turn_cmd)
        rospy.sleep(1)
        pub.publish(Twist())
        rospy.sleep(1)

    rospy.spin()

if __name__ == '__main__':
    try:
        draw_triangle()
    except rospy.ROSInterruptException:
        pass
```

5 Pointed Star:

```
#!/usr/bin/env python

import rospy
from geometry_msgs.msg import Twist

def draw_star():
    rospy.init_node('draw_star', anonymous=True)
    pub = rospy.Publisher('/turtle1/cmd_vel', Twist, queue_size=10)
    rate = rospy.Rate(1)
    move_cmd = Twist()
    turn_cmd = Twist()

    move_cmd.linear.x = 2.0
    turn_cmd.angular.z = 2.51327  # 144 degrees in radians

    for _ in range(5):
        pub.publish(move_cmd)
        rospy.sleep(2)
        pub.publish(Twist())
        rospy.sleep(1)
        pub.publish(turn_cmd)
        rospy.sleep(1)
        pub.publish(Twist())
        rospy.sleep(1)

    rospy.spin()

if __name__ == '__main__':
    try:
        draw_star()
    except rospy.ROSInterruptException:
        pass
```