



## YTU RACING ROS COMMANDS REPORT

**Name-Surname : Adem Berke Nargöl**

**Department : Autonomous**

**Date : 29.09.2023**

**'catkin\_make'** : Bu komut, projeyi derlememizi ve çalıştırmamızı sağlar.

**'catkin\_create\_pkg'** : Bu komut, ilk parametresinin ismi ile bir paket oluşturur ve sonraki parametreleri oluşturulan paketin bağımlılığı olarak ayarlar.

**'source devel/setup.bash'** : Bu komut, ROS projelerinin çalıştırılabilir duruma getirilmesi için gerekli olan çevresel değişkenleri ve komutları etkinleştirmek için kullanılır.

**'roscore'** : Bu komut, ROS (Robot Operating System) sisteminin merkezi yönetim bileşenidir ve bir ROS uygulamasını başlatmak için gereken temel hizmetleri sağlar.

**'roslaunch'** : Bu komut, bir ROS paketinde bulunan özel bir yürütülebilir dosyayı çalıştırmak için kullanılır. 1.parametre, çalıştırmak istediğiniz ROS paketinin adını temsil eder ve 2.parametre, paket içinde çalıştırılacak olan özel yürütülebilir dosyanın adını temsil eder.

**'rospy.init\_node('script.py', anonymous=True)'** : Bu komut, bir düğümü (node) başlatılması için kullanılır. **'anonymous=True'** parametresi, ROS düğümünün (node) adının benzersiz olup olmayacağını belirler.

**'rospy.Publisher'** : Bu komut, bir yayıncı(publisher) oluşturmamızı sağlar. Parametreleri genel olarak yayın türü, yayın konusu (topic) ve maximum kapasitedir.

**'rospy.Rate'** : Bu komut, bir ROS düğümünün (node) belirli bir frekansta çalışmasını sağlar. **'rate.sleep()'** metodu çağrıldığında, düğümün belirli bir frekansla çalışmasını sağlamak için gereken süreyi otomatik olarak hesaplar.

**'rospy.loginfo()'** : Bu fonksiyon, ROS log seviyesini belirler ve bu seviyede bir log mesajı yayınlar. Bu mesajlar genellikle düğümün çalışması hakkında bilgi verir ve hata ayıklama (debugging) amacıyla kullanılır.

**'rospy.publish()'** : Belirli bir konu üzerinde yayın yapmak için bu komut kullanılır. Bu yayınlar, diğer ROS düğümleri tarafından bu konuyu dinleyen (subscribed) düğümlere gönderilebilir. Bu şekilde, ROS sistemi içinde veri alışverişi ve iletişim gerçekleştirilebilir..

**'roslaunch'** : ROS paketlerini ve düğümleri başlatmak için kullanılır. Özellikle büyük ROS projelerini yönetmek ve konfigürasyon dosyalarını kullanmak için uygundur.

**'rostopic'** : ROS konularını listeleyip, bilgi almak ve konular üzerinde işlem yapmak için kullanılır. Örneğin, **'rosservice list'**, **'rosservice info'**, **'rosservice call'** gibi alt komutları içerir.

**'roscall'** : ROS düğümlerini listeleyip, bilgi almak ve düğümler üzerinde işlem yapmak için kullanılır. Örneğin, **'roscall list'**, **'roscall info'**, **'roscall kill'** gibi alt komutları içerir.

**'roscall'** : ROS veri kayıtlarını yönetmek ve yeniden oynatmak için kullanılır. ROS mesajlarını kaydetmek ve kaydedilen verileri analiz etmek için uygundur.

**'rviz'** : ROS Visualization (RViz) aracını başlatır. RViz, robotların ve çevrelerin 3D görselleşmesini yapmak ve ROS verilerini görsel olarak görüntülemek için kullanılır.

**'roswtf'** : ROS sistemini tanımlamak, sorunları tespit etmek ve teşhis etmek için kullanılır. ROS sistemini kontrol etmek ve sorunları gidermek için önemlidir.

