

## BİRDEN FAZLA CATKIN WORKSPACE'İ OLUŞTURMA

Farklı projeler üzerinde çalışılacağı zaman bu projeleri farklı workspace'ler içinde tutarak ayırmak mantıklı olacaktır. Halihazırda bir workspace varken yeni bir workspace oluşturma süreci aslında herhangi bir workspace oluşturmakla aynıdır ancak ek olarak birkaç işlem yapmak gerekir.

Yeni bir workspace oluşturmadan önce halihazırda bulunan workspace'in içindeki yapıları incelemek mantıklı olacaktır.

```
ros@ros-vm:~$ cd catkin_ws/
ros@ros-vm:~/catkin_ws$ ls
build  devel  src
ros@ros-vm:~/catkin_ws$ cd src/
ros@ros-vm:~/catkin_ws/src$ ls
CMakeLists.txt  my_robot_tutorials
ros@ros-vm:~/catkin_ws/src$ cd ..
ros@ros-vm:~/catkin_ws$ catkin_make
Base path: /home/ros/catkin_ws
Source space: /home/ros/catkin_ws/src
Build space: /home/ros/catkin_ws/build
Devel space: /home/ros/catkin_ws/devel
Install space: /home/ros/catkin_ws/install
####
#### Running command: "make cmake_check_build_system" in "/home/ros/catkin_ws/build"
####
####
#### Running command: "make -j4 -l4" in "/home/ros/catkin_ws/build"
####
ros@ros-vm:~/catkin_ws$ ls
build  devel  src
```

“cd catkin\_ws/” yaparak workspace'in içine erişilir. “ls” komutuyla ise workspace'in içindeki diğer dosyaları görülebilir. Workspace'in içindeki src dosyasının altında ise önceden oluşturulan paketler yer almaktadır. “catkin\_make” komutu ise cmake komudunu aktive ederek build ve devel dosyalarını yeniden oluşturur.

Terminale catkin\_ws içindeyken “cd devel/” yazıldığında ise devel dosyasının içerisine gidilmiş olur. “ls” komutu aracılığıyla devel dosyasının altındaki diğer dosyalara bakıldığında ise o dosyalardan birisinin “setup.bash” olduğu görülür. Workspace atmosferini çalışılabilecek hale getirmek için devel dosyasının içinde “source setup.bash” yazılmalı. Yani başka bir deyişle setup.bash dosyası source'lanmalı.

Bu dosyanın içeriğine bakmak için ise terminale “cat ~/.bashrc” yazılabilir. Linux işletim sisteminde “cat” komutu concatenate komudunun kısaltmasıdır ve terminal aracılığıyla dosyanın kendisini doğrudan açmadan bir dosyanın içeriğini göstermeye yarar.

```
fi
source /opt/ros/melodic/setup.bash
source ~/catkin_ws/devel/setup.bash
ros@ros-vm:~/catkin_ws/devel$
```

```
<-- For Kinetic: /opt/ros/kinetic/setup.bash
For Melodic: /opt/ros/melodic/setup.bash
For Noetic: /opt/ros/noetic/setup.bash
```

İlk satır önceden bilgisayarda kurulu olan ROS versiyonunun beyanıdır. Diğer satır ise workspace ile ilgilidir. Başka bir

deyişle önce ROS'un kaynağı daha sonra ise üzerinde çalışılacak olan workspace'in kaynağı belirtilir. Buradan anlaşıldığı üzere aynı anda çalışan bir catkin workspace'i olabilir.

Şimdi ise ne olacağını görmek için yeni bir workspace oluşturalım. Bu workspace'in adı örnek olarak “another\_catkin\_ws” seçilebilir. Bunun dışında istenilen başka isimler de kullanılabilir. Sırasıyla terminale yazılacaklar:

- mkdir another\_catkin\_ws (örnek olarak seçilebilecek workspace adı) **workspace'i oluşturmak için**
- cd another\_catkin\_ws/ **bu workspace'in içine girmek için**
- mkdir src **bu workspace'in içindeki src dosyasını oluşturmak için**
- catkin\_make **cmake komudunu çalıştırmak için – build ve devel dosyaları oluşur**

Bu yeni oluşturulan devel dosyasının içeriğine “ls” komutuyla baktığımızda göreceğimiz dosyalardan birisi setup.bash dosyasıdır. Bu yeni oluşturulan workspace'i sorunsuz kullanabilmek için birkaç düzenleme yapılması gerekiyor. Bunlar sırasıyla:

- Vim editörü kullanılarak bashrc içine gidilmeli. Bunun için terminale “vim ~/.bashrc” yazılmalı. (home directory'sinde)

- Bu dosyanın içinde yukarıdaki görselde olduğu gibi en alt iki satırda yazanlar gözükecektir. Bunlara ek olarak “source ~/another\_catkin\_ws/devel/setup.bash” eklenmeli. **eğer workspace adı farklıysa o workspace adı yazılmalı**
- Bu dosyanın içinde en alt satırda yazan workspace source’lanacaktır. Bunun daha sağlıklı bir şekilde ilerlemesi için source’lanmasını istemediğimiz workspace’in source satırı yorum olarak ayarlanabilir. Bunun için o satırın başına “#” koymak yeterli olacaktır. O satırın renginin değişmesinden de bu işlem takip edilebilir. Kısacası kullanılması istenen workspace’in source’lanması yeterli olacaktır.

Bu adımlar uygulandıktan sonra yeni workspace’imiz kullanıma hazır:)

## EK OLARAK

- Bashrc dosyasını düzenlemenin vim editöründen başka farklı yolları da var. Eğer vim editörü yüklü değilse “nano ~/.bashrc” komudu terminale yazılarak da düzenleme yapılabilir.
- Herhangi bir metin düzenleyici açılmadan da gereken satır bashrc içine eklenebilir. Bunu yapmak için terminale “**echo "source ~/workspace\_adı/devel/setup.bash" >> ~/.bashrc**” yazılmalıdır.
- Eğer ki bashrc içinde bu source’lama olayı yapılmazsa her yeni terminal açıldığında workspace içine gidilmeli ve “source devel/setup.bash” yazılmalıdır. Aslında bashrc içine gidip source’lama yapmak her seferinde uğraşmak yerine kısa yolla işi çözer.