

Hossam-Eddin Salem

Using: Ubuntu20.04—ROS Noetic—
python3.8.2

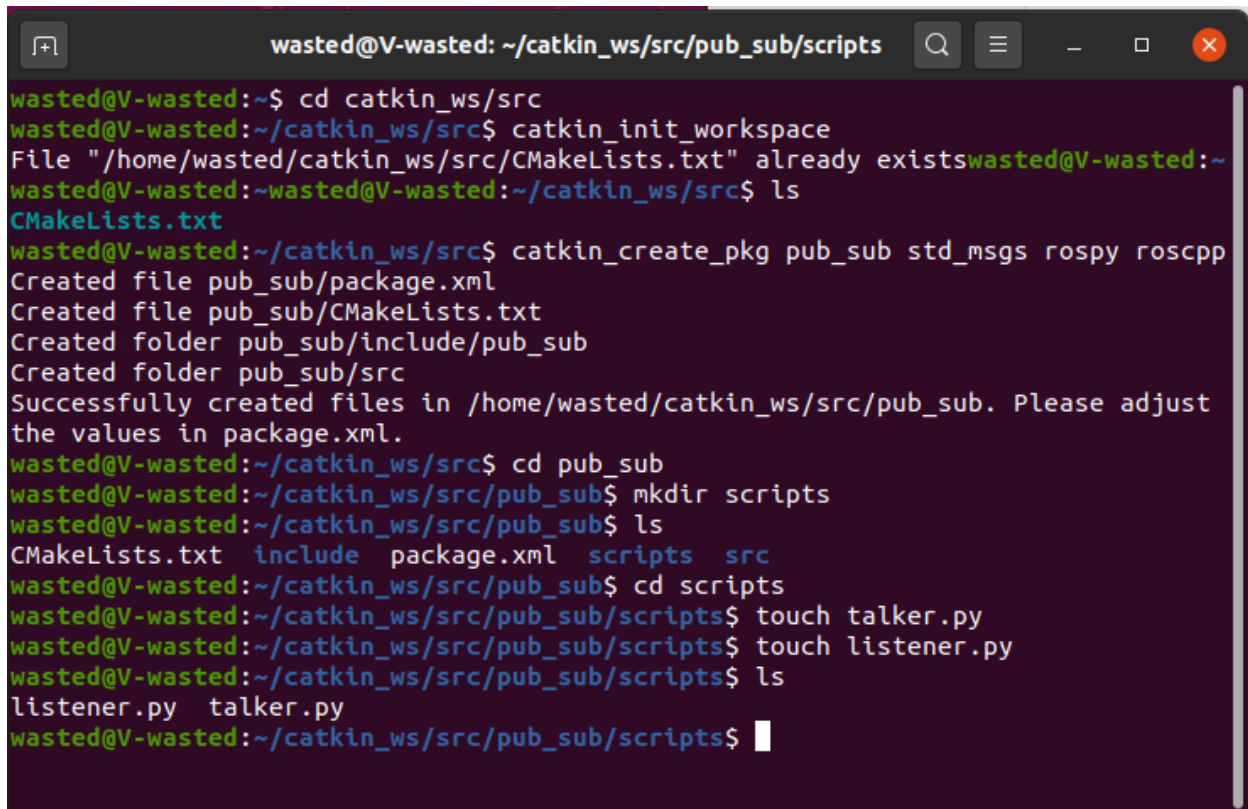
Tutorial about
2 nodes sharing a string message in a
custom package
(Explanation **below pictures**)

```
wasted@V-wasted: ~  
wasted@V-wasted:~$ sudo apt install python3.8  
[sudo] password for wasted:  
Sorry, try again.  
[sudo] password for wasted:  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
python3.8 is already the newest version (3.8.2-1ubuntu1.1).  
0 upgraded, 0 newly installed, 0 to remove and 245 not upgraded.  
wasted@V-wasted:~$ python3  
Python 3.8.2 (default, Apr 27 2020, 15:53:34)  
[GCC 9.3.0] on linux  
Type "help", "copyright", "credits" or "license" for more information.  
>>>
```

In the terminal window we start with installing python 3, enter the password (if exists), then check the installed python version as shown in the picture above:

```
wasted@V-wasted: ~/catkin_ws  
wasted@V-wasted: ~  
wasted@V-wasted:~$ source /opt/ros/noetic/setup.bash  
wasted@V-wasted:~$ mkdir -p ~/catkin_ws/src  
wasted@V-wasted:~$ cd ~/catkin_ws/  
wasted@V-wasted:~/catkin_ws$ catkin_make  
Base path: /home/wasted/catkin_ws  
Source space: /home/wasted/catkin_ws/src  
Build space: /home/wasted/catkin_ws/build  
Devel space: /home/wasted/catkin_ws/devel  
Install space: /home/wasted/catkin_ws/install  
Creating symlink "/home/wasted/catkin_ws/src/CMakeLists.txt" pointing to "/opt/  
ros/noetic/share/catkin/cmake/toplevel.cmake"  
####  
#### Running command: "cmake /home/wasted/catkin_ws/src -DCATKIN_DEVEL_PREFIX=/  
home/wasted/catkin_ws/devel -DCMAKE_INSTALL_PREFIX=/home/wasted/catkin_ws/insta  
ll -G Unix Makefiles" in "/home/wasted/catkin_ws/build"  
####  
CMake Warning (dev) in CMakeLists.txt:  
  No project() command is present.  The top-level CMakeLists.txt file must  
  contain a literal, direct call to the project() command.  Add a line of  
  code such as  
  
    project(ProjectName)
```

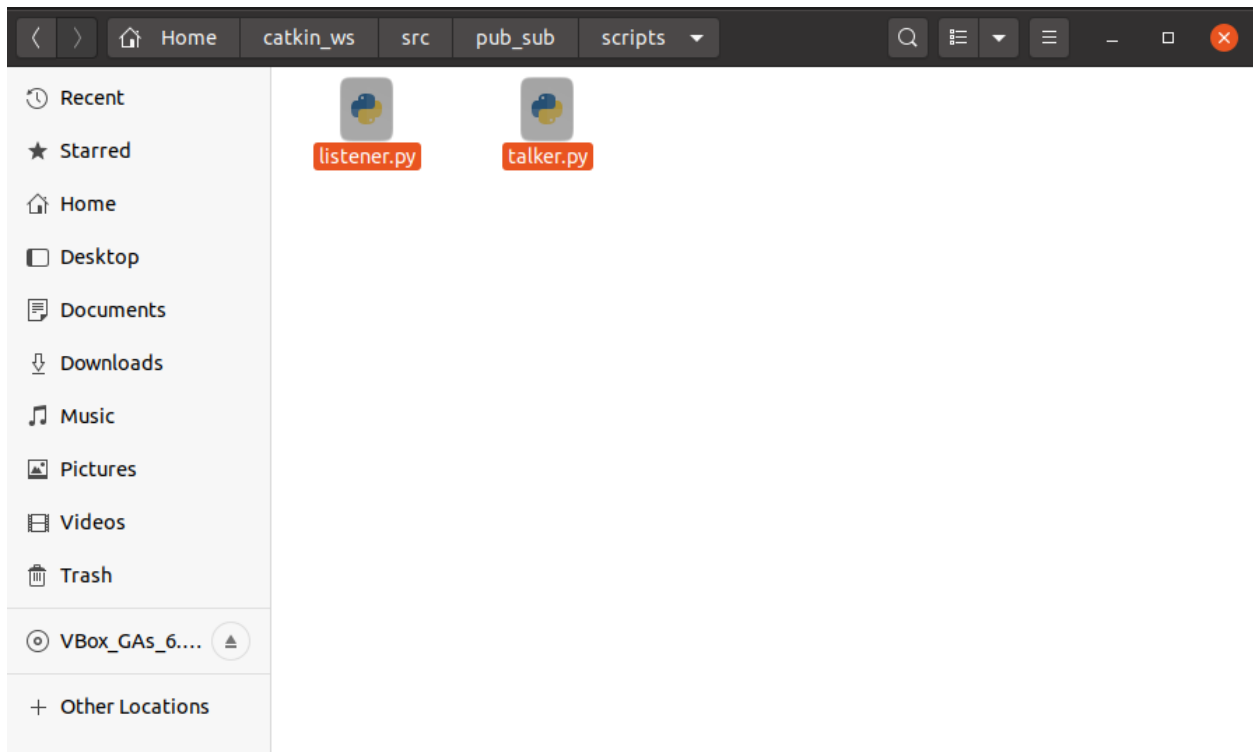
Here we create a catkin workspace by entering the 4 codes shown in the picture above in a **new terminal window**.

A terminal window titled "wasted@V-wasted: ~/catkin_ws/src/pub_sub/scripts" with standard window controls. The terminal shows the following commands and output:

```
wasted@V-wasted:~$ cd catkin_ws/src
wasted@V-wasted:~/catkin_ws/src$ catkin_init_workspace
File "/home/wasted/catkin_ws/src/CMakeLists.txt" already exists
wasted@V-wasted:~/catkin_ws/src$ ls
CMakeLists.txt
wasted@V-wasted:~/catkin_ws/src$ catkin_create_pkg pub_sub std_msgs rospy roscpp
Created file pub_sub/package.xml
Created file pub_sub/CMakeLists.txt
Created folder pub_sub/include/pub_sub
Created folder pub_sub/src
Successfully created files in /home/wasted/catkin_ws/src/pub_sub. Please adjust
the values in package.xml.
wasted@V-wasted:~/catkin_ws/src$ cd pub_sub
wasted@V-wasted:~/catkin_ws/src/pub_sub$ mkdir scripts
wasted@V-wasted:~/catkin_ws/src/pub_sub$ ls
CMakeLists.txt  include  package.xml  scripts  src
wasted@V-wasted:~/catkin_ws/src/pub_sub$ cd scripts
wasted@V-wasted:~/catkin_ws/src/pub_sub/scripts$ touch talker.py
wasted@V-wasted:~/catkin_ws/src/pub_sub/scripts$ touch listener.py
wasted@V-wasted:~/catkin_ws/src/pub_sub/scripts$ ls
listener.py  talker.py
wasted@V-wasted:~/catkin_ws/src/pub_sub/scripts$
```

The codes above in order are for:

- Moving to **catkin_ws/src** folder directory (you can preview those folders by browsing your Home files).
- Initializing the workspace.
- **Ls**: for listing the folder's contents (just for checking, not necessary).
- Then we create a new package in our workspace folder, I named it: **pub_sub** (stands for: publisher_subscriber).
- We use "**cd pub_sub**" to move to the directory of the package we just created.
- We make a folder inside named scripts.
- We move to scripts using "**cd scripts**".
- We create 2 new files called: **talker.py** & **listener.py** (will be our python files).



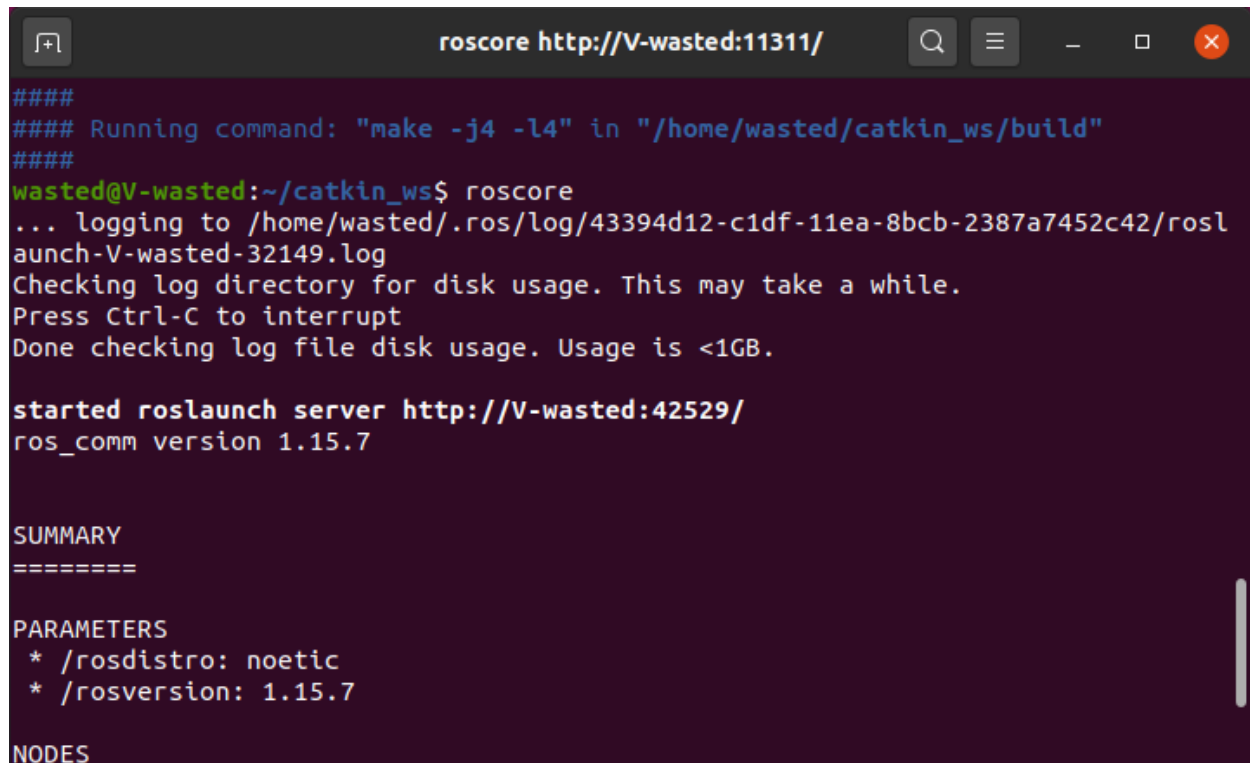
Here we can see the files we just created using the terminal window codes. We open them and start writing our codes (they will be attached in my github link).

```
wasted@V-wasted: ~/catkin_ws

wasted@V-wasted:~/catkin_ws/src/pub_sub/scripts$ chmod +x talker.py
wasted@V-wasted:~/catkin_ws/src/pub_sub/scripts$ chmod +x listener.py
wasted@V-wasted:~/catkin_ws/src/pub_sub/scripts$ cd ..
wasted@V-wasted:~/catkin_ws/src/pub_sub$ cd ..
wasted@V-wasted:~/catkin_ws/src$ cd ..
wasted@V-wasted:~/catkin_ws$ catkin_make
Base path: /home/wasted/catkin_ws
Source space: /home/wasted/catkin_ws/src
Build space: /home/wasted/catkin_ws/build
Devel space: /home/wasted/catkin_ws/devel
Install space: /home/wasted/catkin_ws/install
####
#### Running command: "cmake /home/wasted/catkin_ws/src -DCATKIN_DEVEL_PREFIX=
/home/wasted/catkin_ws/devel -DCMAKE_INSTALL_PREFIX=/home/wasted/catkin_ws/ins
tall -G Unix Makefiles" in "/home/wasted/catkin_ws/build"
####
CMake Warning (dev) in CMakeLists.txt:
  No project() command is present.  The top-level CMakeLists.txt file must
  contain a literal, direct call to the project() command.  Add a line of
  code such as

    project(ProjectName)
```

We used the codes above to make both `talker.py` & `listener.py` executable files. Then we go back to our `catkin_ws` folder directory by using the code "`cd ..`" three times. After that we execute the "`catkin_make`" command as changes has ben made in the `CMakeList` file.

A terminal window with a dark background and light text. The title bar shows 'roscore http://V-wasted:11311/'. The output shows the execution of 'roscore' in a terminal session. It includes messages about logging, disk usage checks, and the start of a roslaunch server. It also displays a summary of parameters like 'rostdistro: noetic' and 'rosversion: 1.15.7', and a section for nodes.

```
#####
##### Running command: "make -j4 -l4" in "/home/wasted/catkin_ws/build"
#####
wasted@V-wasted:~/catkin_ws$ roscore
... logging to /home/wasted/.ros/log/43394d12-c1df-11ea-8bcb-2387a7452c42/rosl
launch-V-wasted-32149.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://V-wasted:42529/
ros_comm version 1.15.7

SUMMARY
=====

PARAMETERS
* /rostdistro: noetic
* /rosversion: 1.15.7

NODES
```

We run `roscore` as we will use it in the next few steps in a new terminal window.

Problem scenario:

```
/usr/bin/env: 'python': No such file or directory
```

Possible Solution #1

If Python 3 is not installed, install it: `apt-get install python3`

Possible Solution #2

If Python 3 has been installed, run these commands: `whereis python3`

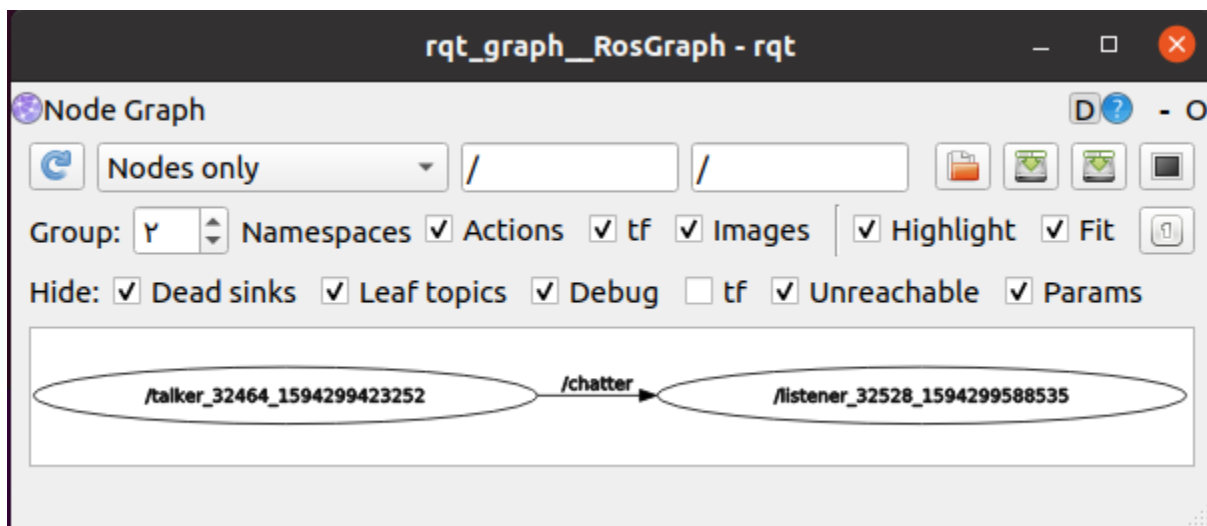
Then we create a symlink to it: `sudo ln -s /usr/bin/python3 /usr/bin/python`

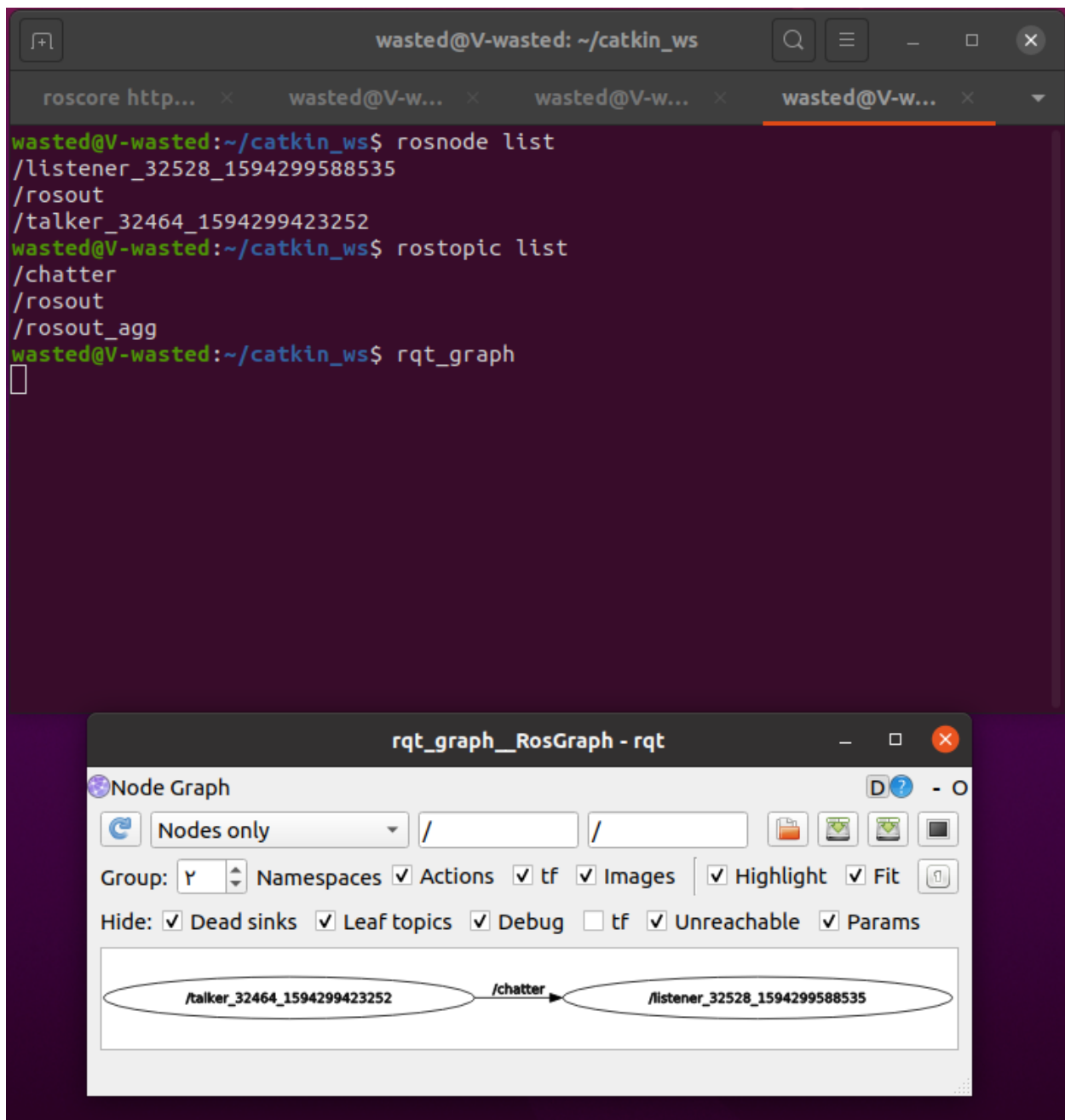
This is a potential error you may run by during the next step, and that's how to deal with it (if not, please ignore this).

```
wasted@V-wasted: ~/catkin_ws
roscore http://V-was... x wasted@V-wasted: ~... x wasted@V-wasted: ~... x
wasted@V-wasted:~/catkin_ws$ source devel/setup.bash
wasted@V-wasted:~/catkin_ws$ rosrn pub_sub listener.py
[INFO] [1594299588.697299]: /listener_32528_1594299588535I heard hello world 1
594299588.6960251
[INFO] [1594299588.802090]: /listener_32528_1594299588535I heard hello world 1
594299588.8005602
[INFO] [1594299588.897343]: /listener_32528_1594299588535I heard hello world 1
594299588.8961334
[INFO] [1594299589.004863]: /listener_32528_1594299588535I heard hello world 1
594299589.0035083
[INFO] [1594299589.098485]: /listener_32528_1594299588535I heard hello world 1
594299589.0972028
[INFO] [1594299589.201960]: /listener_32528_1594299588535I heard hello world 1
594299589.2006223
[INFO] [1594299589.317250]: /listener_32528_1594299588535I heard hello world 1
594299589.315902
[INFO] [1594299589.397966]: /listener_32528_1594299588535I heard hello world 1
594299589.3967912
[INFO] [1594299589.508652]: /listener_32528_1594299588535I heard hello world 1
594299589.507215
[INFO] [1594299589.599382]: /listener_32528_1594299588535I heard hello world 1
594299589.5978572
```

Make sure you enter the first code whenever you open a new terminal window: “source devel/setup.bash”. Then start running the `talker.py` & `listener.py` nodes using: “rosrn pub_sub talker.py” and “rosrn pub_sub listener.py” (of course the package name will change according to the one you made).

Results:





To show the current active nodes, topics and request the graph, enter the three commands above.