

## 06. TF 개념 익히기

AI ROBOT

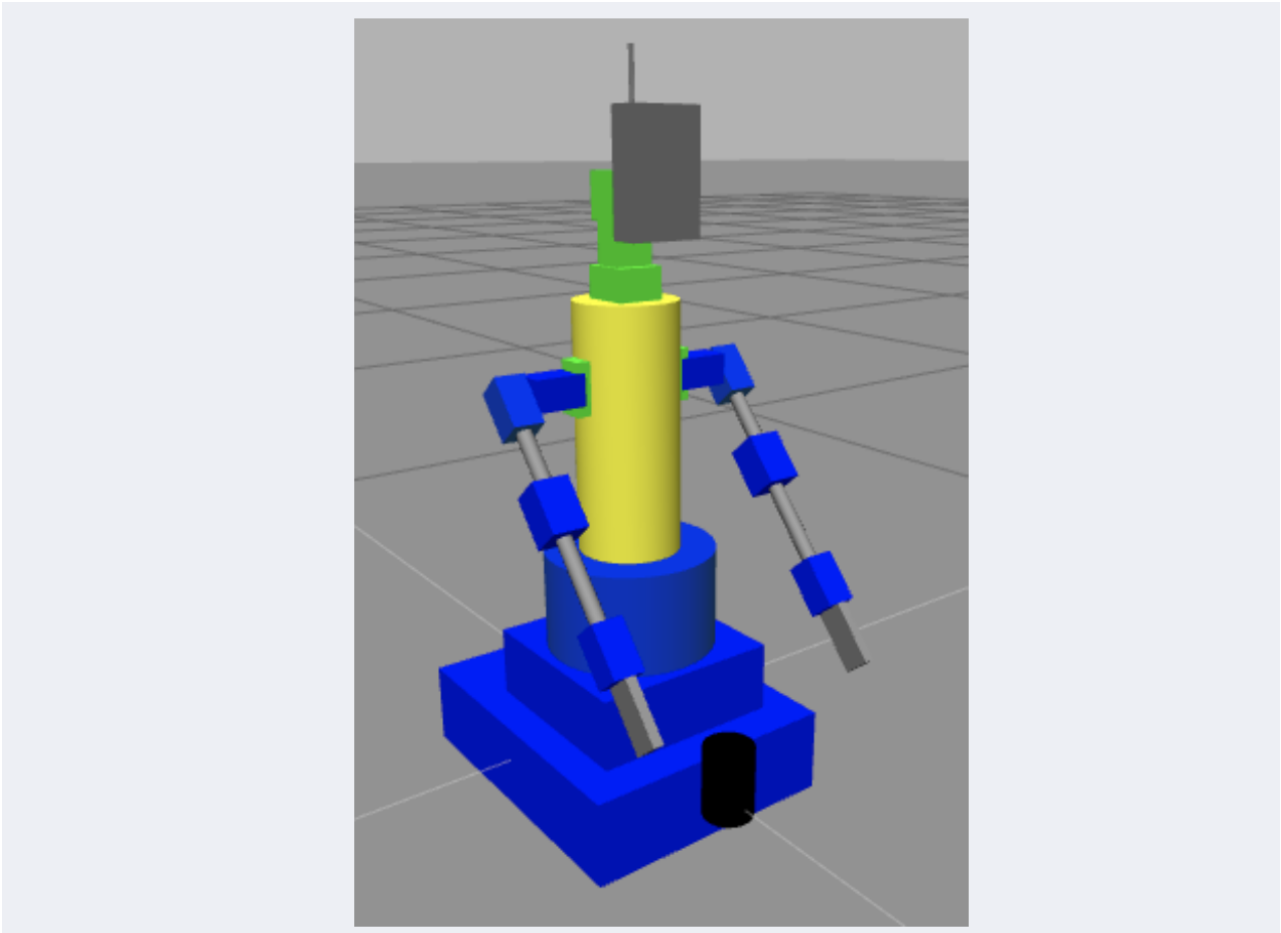
Exported on 06/08/2021

## Table of Contents

<b>1</b>	<b>TF .....</b>	<b>3</b>
1.1	Transformation .....	3
1.2	간단하지 않은 좌표의 해석.....	4
1.3	TF의 개념을 간단하게 실습을 해보자 .....	5
1.4	roslaunch turtle_tf turtle_tf_demo.launch .....	5
1.5	teleop_key 실행.....	6
1.6	추적하는 turtlesim .....	7
1.7	현재 토픽의 흐름 .....	8
1.8	현재 tf의 상황을 기록.....	8
1.9	evince frames.pdf.....	8
1.10	같은 역할로 rosrn rqt_tf_tree rqt_tf_tree.....	9
1.11	tf topic을 한 번만 보자.....	9
1.12	rosrn tf tf_echo turtle1 turtle2 .....	10
1.13	turtle1에서본 turtle2의 좌표 .....	10
1.14	rosrn rviz rviz .....	11
1.15	frame의 상황과 함께 관찰할 수 있다.....	11
<b>2</b>	<b>TF - pub and sub.....</b>	<b>12</b>
2.1	설치 .....	12
2.2	jupyter notebook 실행 .....	12
2.3	python2로 새문서를 열어서 rospy가 import 되면 OK.....	12
2.4	현재 turtle_tf_demo를 실행하고 Jupyter notebook을 실행해둬 .....	13
2.5	간단히 원리만 확인하는 차원에서 import 모듈.....	13
2.6	turtle2에서 바라본 turtle1의 상대 좌표와 자세를 받고 .....	14
2.7	거리를 계산해본다면 .....	14
2.8	이렇게 해볼 수 있을듯 .....	14
2.9	robot state publisher .....	14

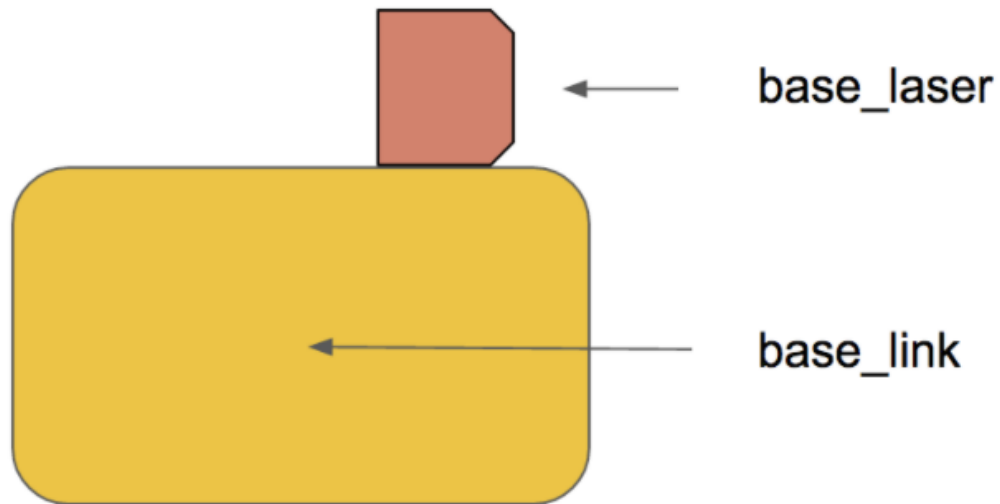
# 1 TF

## 1.1 Transformation



- 로봇은 3D 좌표를 아주 많이 가지고 있는 경우가 많다

## 1.2 간단하지 않은 좌표의 해석



- laser가 읽은 정보를 base\_link 입장에서 해석하려면?

### 1.3 TF의 개념을 간단하게 실습을 해보자

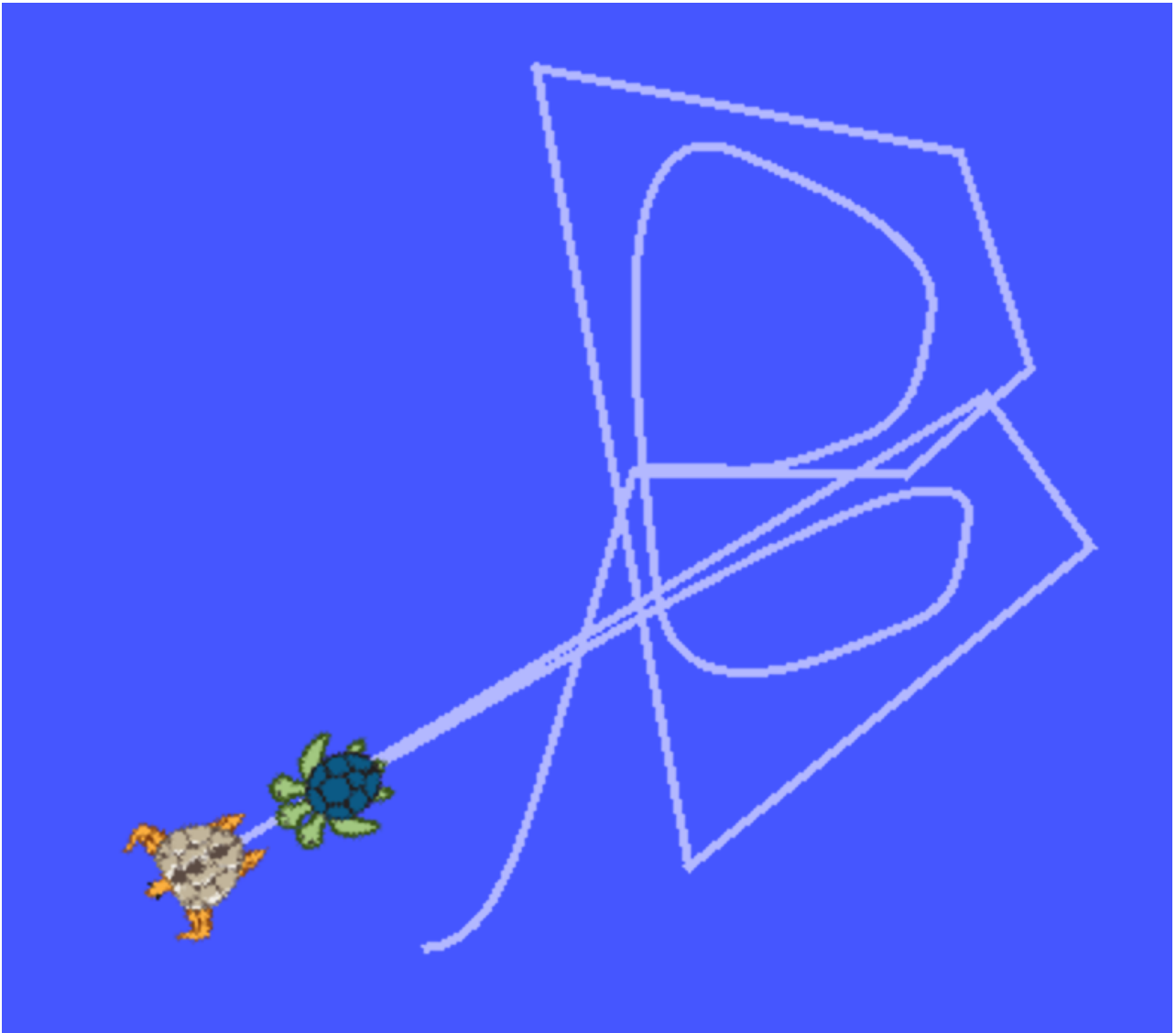
#### 1.4 `roslaunch turtle_tf turtle_tf_demo.launch`



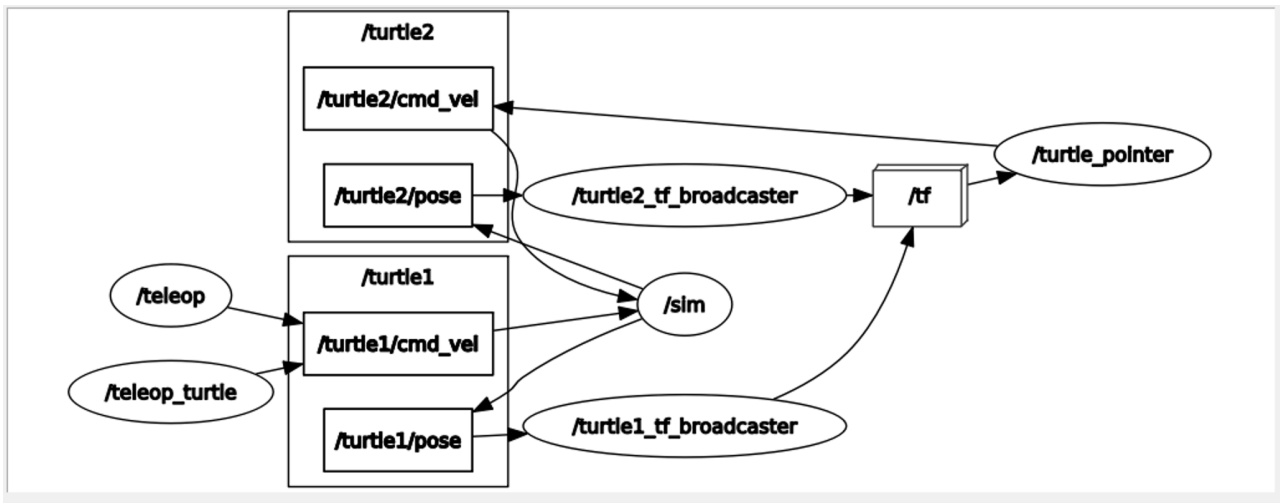
## 1.5 teleop\_key 실행

```
pw@melodic:~$ rosrun turtlesim turtle_teleop_key
Reading from keyboard
-----
Use arrow keys to move the turtle. 'q' to quit.
```

## 1.6 추적하는 turtlesim



## 1.7 현재 토픽의 흐름



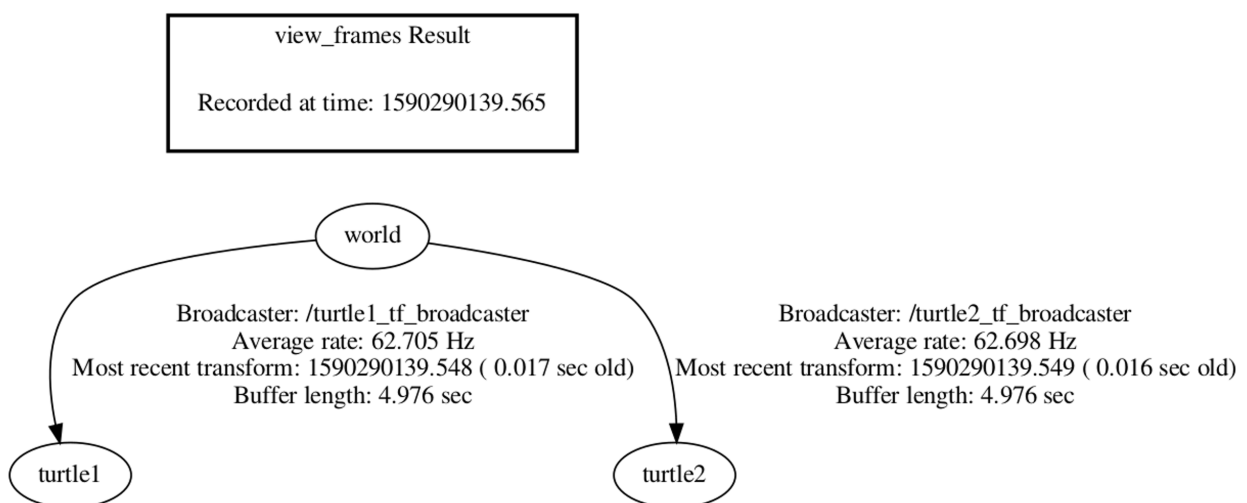
## 1.8 현재 tf의 상황을 기록

```

pw@melodic:~$
pw@melodic:~$ rosrn tf view_frames
Listening to /tf for 5.0 seconds
Done Listening
dot - graphviz version 2.40.1 (20161225.0304)

Detected dot version 2.40
frames.pdf generated
  
```

## 1.9 evince frames.pdf

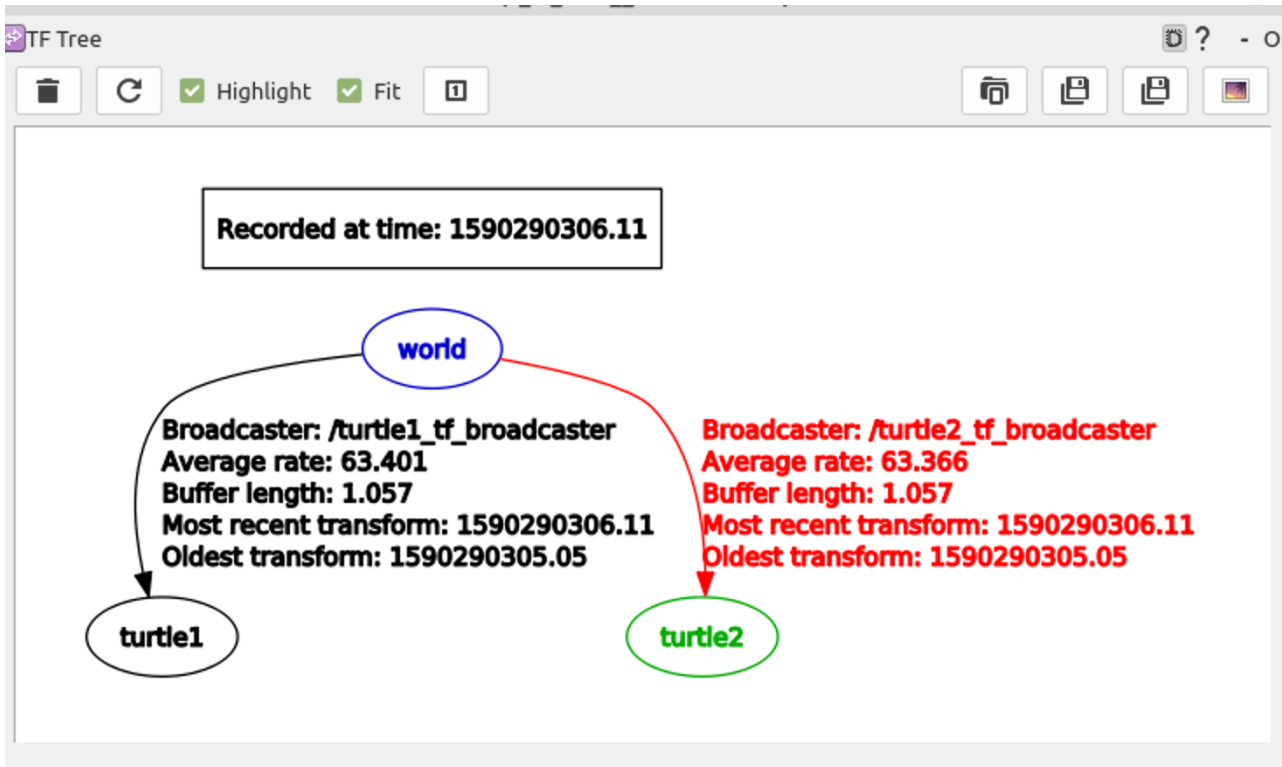


- /turtle1 및 /turtle2라는 두 개의 프레임이 브로드 캐스트
- 공통 부모는 /world 프레임



- **Broadcaster:** TF 데이터의 브로드 캐스터 이름

### 1.10 같은 역할로 `roslaunch rqt_tf_tree rqt_tf_tree`



### 1.11 tftopic을 한 번만 보자

```

pw@melodic:~$ rostopic echo -n1 /tf
transforms:
  header: package.xml
    seq: 0
    stamp:
      secs: 1590290388
      nsecs: 524173021
    frame_id: "world"
  child_frame_id: "turtle2"
  transform:
    translation:
      x: 6.00517559052
      y: 3.0319609642
      z: 0.0
    rotation:
      x: 0.0
      y: 0.0
      z: 0.953937220198
      w: 0.300006299801

```

## 1.12 rosrun tf tf\_echo turtle1 turtle2

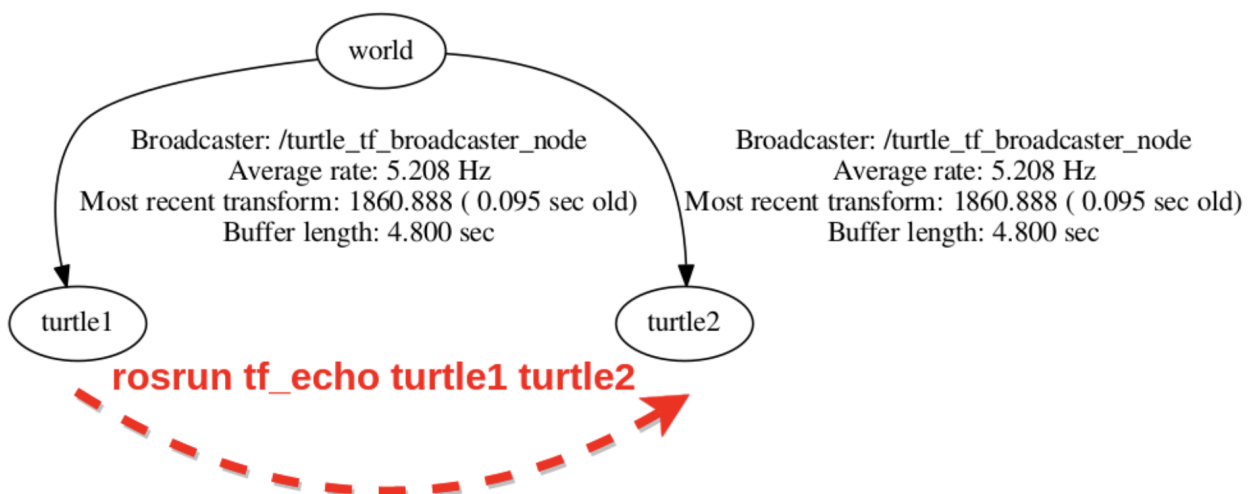
```

pw@melodic:~$ rosrun tf tf_echo turtle1 turtle2
At time 1590290451.660
- Translation: [0.000, 0.000, 0.000]
- Rotation: in Quaternion [0.000, 0.000, 0.996, 0.094]
            in RPY (radian) [0.000, -0.000, 2.953]
            in RPY (degree) [0.000, -0.000, 169.195]
At time 1590290452.381
- Translation: [0.000, 0.000, 0.000]
- Rotation: in Quaternion [0.000, 0.000, 0.996, 0.094]
            in RPY (radian) [0.000, -0.000, 2.953]
            in RPY (degree) [0.000, -0.000, 169.195]
At time 1590290453.373
- Translation: [0.000, 0.000, 0.000]
- Rotation: in Quaternion [0.000, 0.000, 0.996, 0.094]
            in RPY (radian) [0.000, -0.000, 2.953]
            in RPY (degree) [0.000, -0.000, 169.195]
pw@melodic:~$

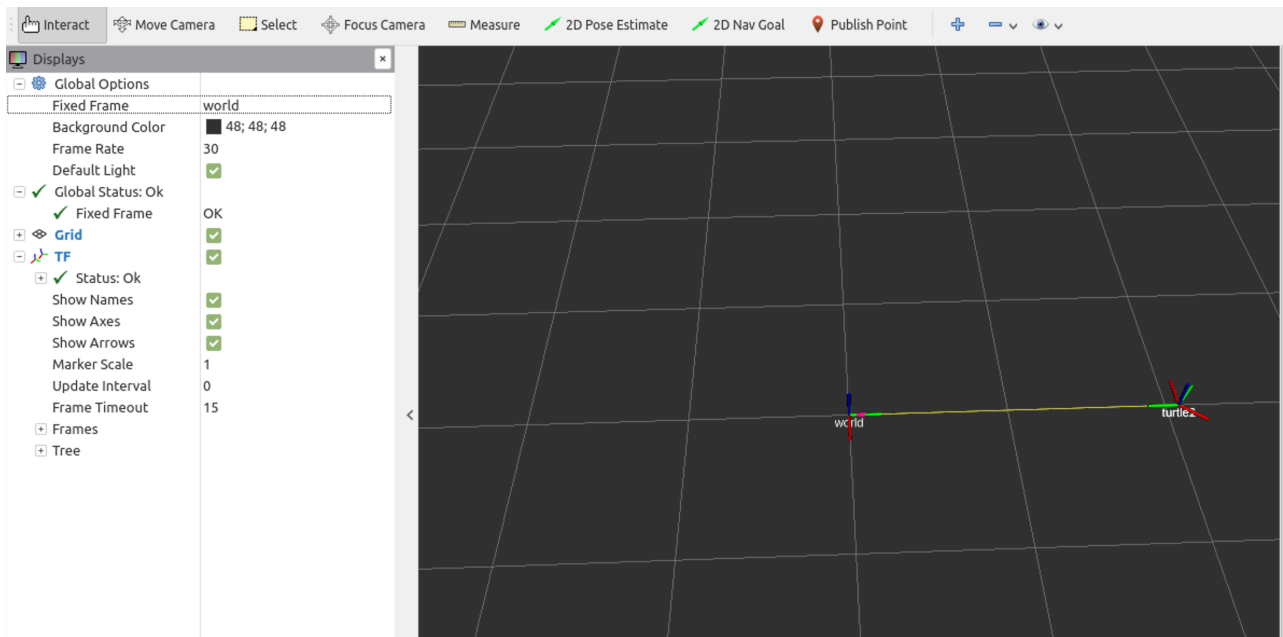
```

- 특정 frame간의 데이터만 볼 수 있다

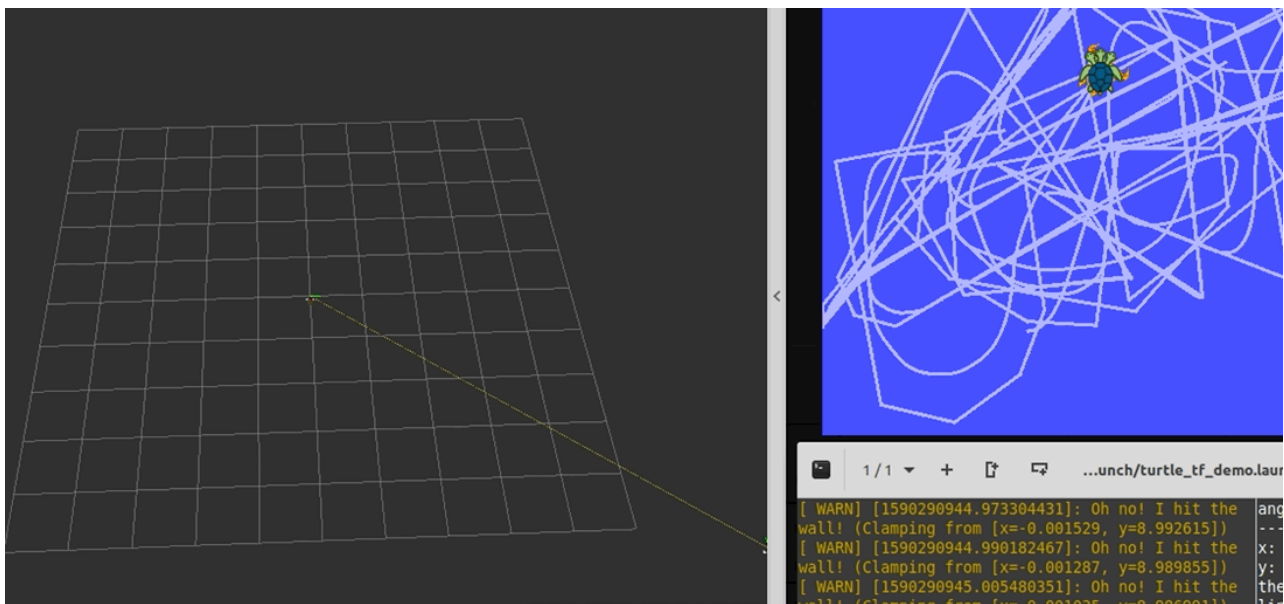
## 1.13 turtle1에서본 turtle2의 좌표



## 1.14 rosrn rviz rviz



## 1.15 frame의 상황과 함께 관찰할 수 있다



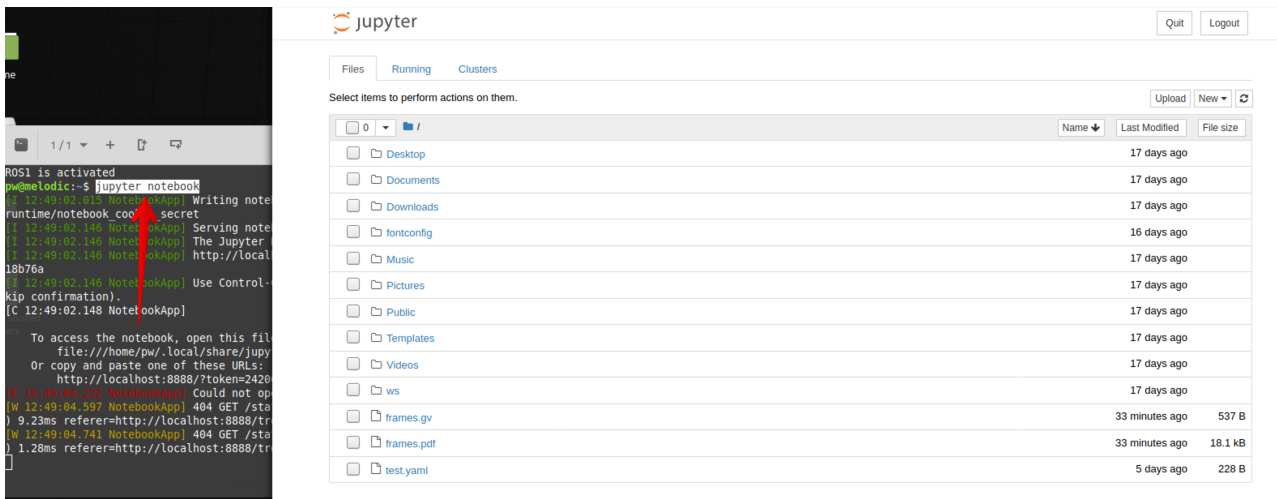
## 2 TF - pub and sub

### 2.1 설치

```
sudo apt install python-pip
pip install --upgrade pip
pip install jupyter
pip install ipython
```

- 그리고 reboot

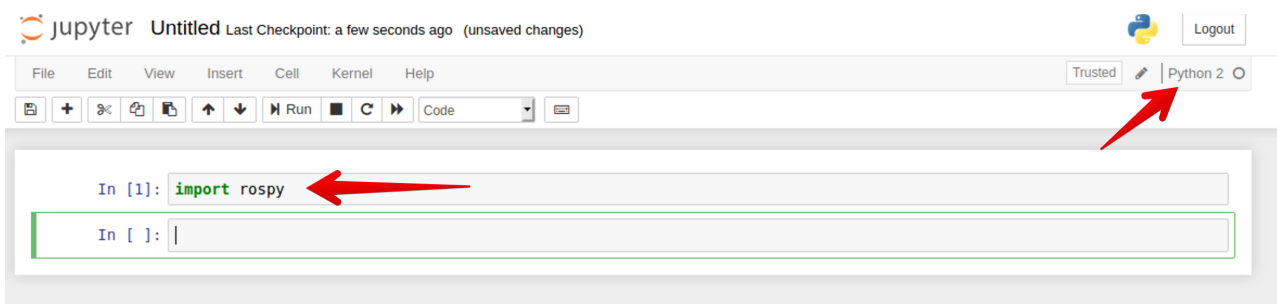
### 2.2 jupyter notebook 실행



The image shows a terminal window on the left and the Jupyter Notebook web interface on the right. In the terminal, the command `jupyter notebook` is executed, and the output shows the notebook server starting. A red arrow points to the `secret` token in the output. The Jupyter Notebook interface on the right shows a file browser with a table of files and folders. A red arrow points to the `secret` token in the terminal output.

Name	Last Modified	File size
Desktop	17 days ago	
Documents	17 days ago	
Downloads	17 days ago	
fontconfig	16 days ago	
Music	17 days ago	
Pictures	17 days ago	
Public	17 days ago	
Templates	17 days ago	
Videos	17 days ago	
ws	17 days ago	
frames.gv	33 minutes ago	537 B
frames.pdf	33 minutes ago	18.1 kB
test.yaml	5 days ago	228 B

### 2.3 python2로 새문서를 열어서 rospy가 import 되면 OK



The image shows the Jupyter Notebook web interface. A new file named 'Untitled' is being created. The interface shows the 'Python 2' kernel selected. A red arrow points to the 'Python 2' dropdown menu. Below the menu, the command `import rospy` is entered in the first code cell. A red arrow points to the `import rospy` command.

```
In [1]: import rospy
```

## 2.4 현재 turtle\_tf\_demo를 실행하고 Jupyter notebook을 실행해둠

```
[W 14:51:30.579 NotebookApp] 404 GET /static/components/react/react-dom.production.min.js (127.0.0.1) 1.17ms referer=http://localhost:8888/notebooks/Documents/Untitled.ipynb?kernel_name=python2
[W 14:51:30.698 NotebookApp] 404 GET /static/components/react/react-dom.production.min.js (127.0.0.1) 0.96ms referer=http://localhost:8888/notebooks/Documents/Untitled.ipynb?kernel_name=python2
[I 14:51:31.191 NotebookApp] Kernel started: 5432a713-52cc-4826-bcee-a221f278c41a
[W 14:51:31.202 NotebookApp] 404 GET /nbextensions/widgets/notebook/js/extension.js?v=20200524145106 (127.0.0.1) 1.51ms referer=http://localhost:8888/notebooks/Documents/Untitled.ipynb?kernel_name=python2
[I 14:51:31.564 NotebookApp] Adapting to protocol v5.1 for kernel 5432a713-52cc-4826-bcee-a221f278c41a
Cloud
[ ]
start debug_demo.launch turtle_tf_demo.launch turtle_tf_sensor.launch
pw@melodic:~$ roslaunch turtle_tf turtle_tf_demo.launch
... logging to /home/pw/.ros/log/8bb2c752-9d82-11ea-8ee9-001c420be203/roslaunch-melodic-6520.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://melodic:39993/

SUMMARY
```

## 2.5 간단히 원리만 확인하는 차원에서 import 모듈

```
In [1]: #!/usr/bin/env python

import rospy
import time
import tf
from turtlesim.msg import Pose
import geometry_msgs.msg
```

```
In [2]: rospy.init_node('turtle_tf_listener')
listener = tf.TransformListener()
turtle_vel = rospy.Publisher('turtle2/cmd_vel',
                             geometry_msgs.msg.Twist, queue_size=1)
```

## 2.6 turtle2에서 바라본 turtle1의 상대 좌표와 자세를 받고

```
In [3]: listener.lookupTransform('/turtle2', '/turtle1', rospy.Time(0))
Out[3]: ([0.0, 0.0, 0.0], [0.0, 0.0, -0.04248841695728023, 0.9990969594711337])

In [4]: (trans,rot) = listener.lookupTransform('/turtle2', '/turtle1',
                                             rospy.Time(0))
                                             trans
Out[4]: [0.0, 0.0, 0.0]
```

## 2.7 거리를 계산해본다면

```
In [5]: import numpy as np
        np.sqrt((np.array([1, 2, 3]) ** 2).mean())
Out[5]: 2.1602468994692869

In [6]: np.sqrt((np.array(trans) ** 2).mean())
Out[6]: 0.0
```

## 2.8 이렇게 해볼 수 있을듯

```
In [9]: for n in range(10):
        time.sleep(0.5)
        (trans,rot) = listener.lookupTransform('/turtle2', '/turtle1',
                                             rospy.Time(0))
        print(np.sqrt((np.array(trans) ** 2).mean()))

0.0
0.0
0.0
0.0
0.202933505379
0.682560812822
1.04779234707
1.33895284011
1.17638973613
0.703204783523
```

## 2.9 robot state publisher

- URDF를 작성하면 자동으로 TF를 broadcasting 함