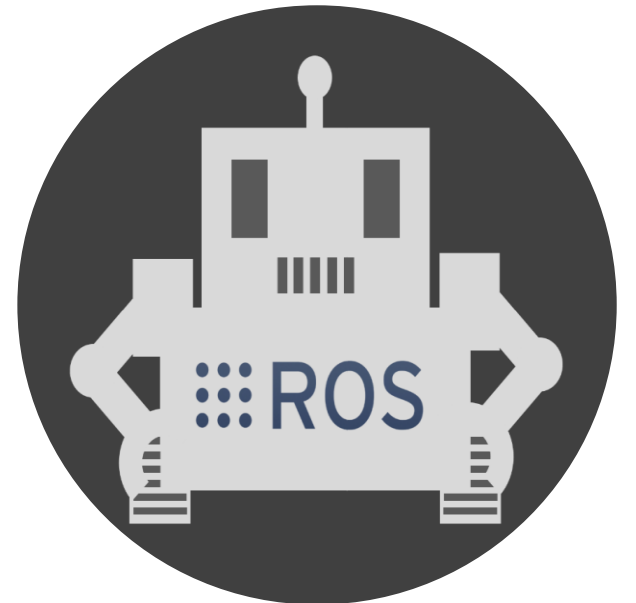


# ROS 기초 강의

## Chapter 2. ROS Node

구선생 로보틱스



# 강의 자료 다운로드

---



ROS 기초 강의 강의노트

[https://drive.google.com/drive/folders/1rRwS2j98HNyj5Is\\_yVXEGj30ILvMPtrz?usp=drive\\_link](https://drive.google.com/drive/folders/1rRwS2j98HNyj5Is_yVXEGj30ILvMPtrz?usp=drive_link)

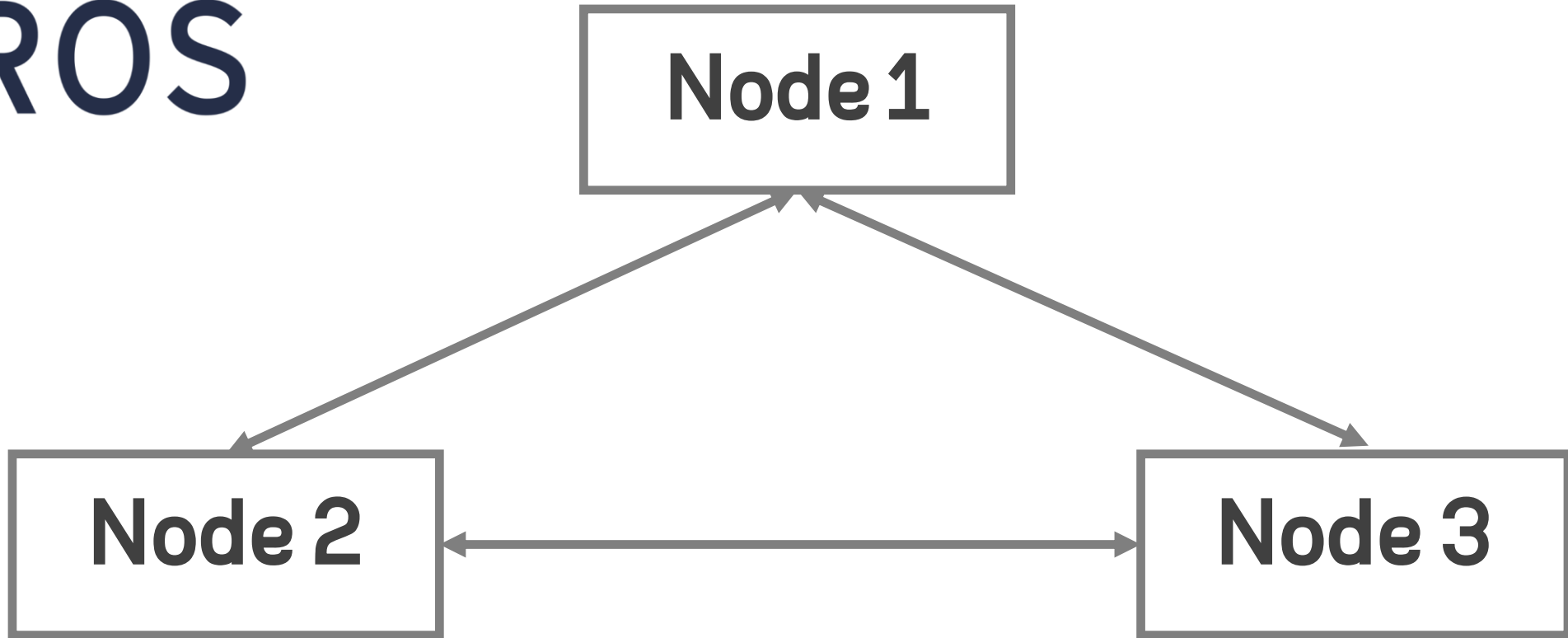
1. ROS Node 기초
2. ROS Node 명령어
3. ROS Node 생성

# ROS Node 기초

## ROS Node란 무엇인가?

- ROS아래에 실행되는 단위 프로세스

ROS



# ROS Node 기초

---

## ROS Node 실행 방법

### 1) roscore 실행

```
$ roscore
```

### 2) turtlesim\_node 실행

```
$ rosrun turtlesim turtlesim_node
```

### 3) turtle\_teleop\_key 실행

```
$ rosrun turtlesim turtle_teleop_key
```

**! roscore는 Node 실행 전에 실행 되어 있어야 하며  
PC에서 한 개만 실행되어야 함**

# ROS Node 기초

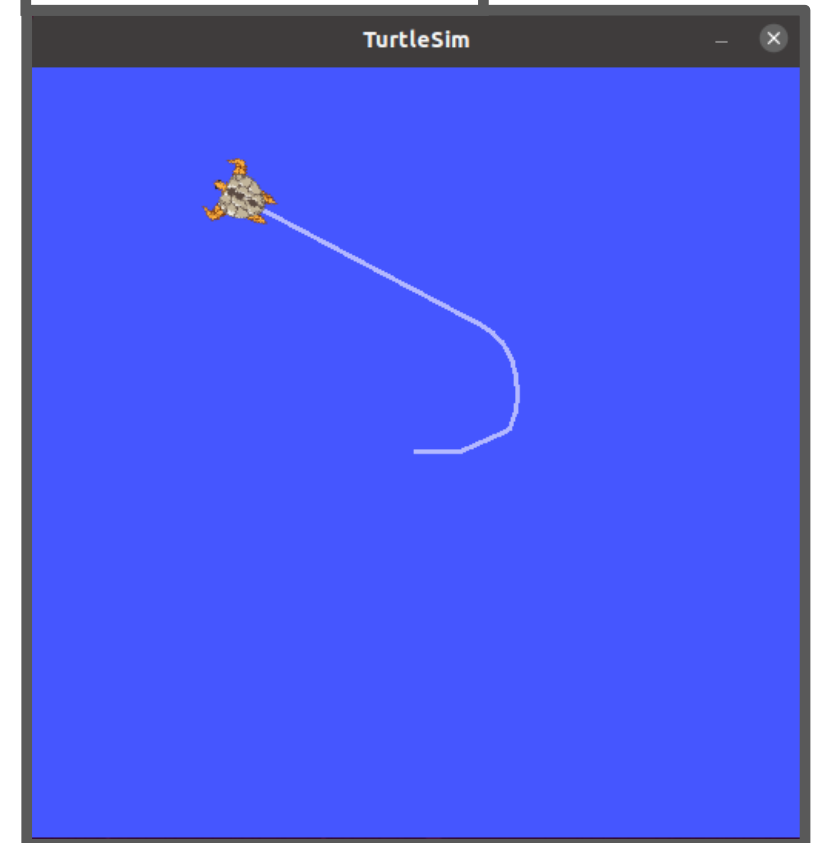
## ROS Node 실행 방법

키보드 방향키를 이용하여 거북이를 조종하기

turtle\_teleop\_key

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

turtlesim\_node



1. ROS Node 기초
2. ROS Node 명령어
3. ROS Node 생성

# ROS Node 명령어

---

- 실행중인 Node list 확인

```
$ rosnode list
```

- Node 정보 보기

```
$ rosnode info [노드_이름]
```

- 실행중인 Node 종료

```
$ rosnode kill [노드_이름]
```

기타 명령어는 아래 위키 참고

<http://wiki.ros.org/rosnode>



1. ROS Node 기초
2. ROS Node 명령어
3. ROS Node 생성

# ROS Node 생성

---

## 1. 워크 스페이스 생성

1) catkin\_ws 폴더 생성

```
$ mkdir catkin_ws
```

2) catkin\_ws 경로 이동

```
$ cd catkin_ws
```

3) src 폴더 생성

```
$ mkdir src
```

```
ubuntu@ubuntu:~$ mkdir catkin_ws
ubuntu@ubuntu:~$ cd catkin_ws
ubuntu@ubuntu:~/catkin_ws$ mkdir src
ubuntu@ubuntu:~/catkin_ws$ ls
src
```

# ROS Node 생성

## 2. ROS Node 생성

### 1) 패키지 생성

```
$ catkin_create_pkg basic_tutorial roscpp
```

```
ubuntu@ubuntu:~/catkin_ws$ cd src
ubuntu@ubuntu:~/catkin_ws/src$ catkin_create_pkg basic_tutorial roscpp
Created file basic_tutorial/package.xml
Created file basic_tutorial/CMakeLists.txt
Created folder basic_tutorial/include/basic_tutorial
Created folder basic_tutorial/src
Successfully created files in /home/ubuntu/catkin_ws/src/basic_tutorial. Please
adjust the values in package.xml.
ubuntu@ubuntu:~/catkin_ws/src$ ls
basic_tutorial
ubuntu@ubuntu:~/catkin_ws/src$ cd basic_tutorial/
ubuntu@ubuntu:~/catkin_ws/src/basic_tutorial$ ls
CMakeLists.txt  include  package.xml  src
ubuntu@ubuntu:~/catkin_ws/src/basic_tutorial$
```


# ROS Node 생성

## 2. ROS Node 생성

### 2) 소스코드 파일 생성

```
$ nano basic.cpp
```

```
ubuntu@ubuntu:~/catkin_ws/src/basic_tutorial$ cd src  
ubuntu@ubuntu:~/catkin_ws/src/basic_tutorial/src$ nano basic.cpp
```



∴ 소스코드는 패키지의 src 경로에서 생성되어야 한다.

# ROS Node 생성

## 2. ROS Node 생성

### 3) 소스코드 작성

```
1  #include <ros/ros.h>
2
3  int main(int argc, char **argv)
4  {
5      ros::init(argc, argv, "basic_node");
6      ros::NodeHandle n;
7
8      ros::Rate loop_rate(60);
9
10     while (ros::ok())
11     {
12         ROS_INFO("Hello world!");
13         ros::spinOnce();
14         loop_rate.sleep();
15     }
16
17     return 0;
18 }
```

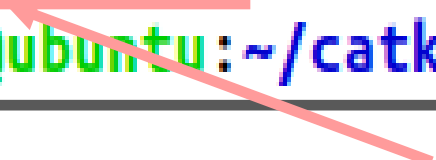
# ROS Node 생성

## 3. CMakeLists.txt 작성

### 1) CMakeLists.txt 편집

```
$ nano CMakeLists.txt
```

```
ubuntu@ubuntu:~/catkin_ws/src/basic_tutorial/src$ cd ..  
ubuntu@ubuntu:~/catkin_ws/src/basic_tutorial$ ls  
CMakeLists.txt  include  package.xml  src  
ubuntu@ubuntu:~/catkin_ws/src/basic_tutorial$ nano CMakeLists.txt
```



∴ CMakeLists.txt는 패키지 경로에 있다.

# ROS Node 생성

## 3. CMakeLists.txt 작성

2) CMakeLists.txt에 add\_executable, target\_link\_libraries 추가

```
116 include_directories (  
117     # include  
118     ${catkin_INCLUDE_DIRS}  
119 )  
120  
121 add_executable(basic_node src/basic.cpp)  
122 target_link_libraries(basic_node ${catkin_LIBRARIES})
```

# ROS Node 생성

## 4. 컴파일

### 1) 컴파일 명령어 입력

```
$ catkin_make
```

```
ubuntu@ubuntu:~/catkin_ws$ catkin_make
Base path: /home/ubuntu/catkin_ws
Source space: /home/ubuntu/catkin_ws/src
Build space: /home/ubuntu/catkin_ws/build
Devel space: /home/ubuntu/catkin_ws/devel
Install space: /home/ubuntu/catkin_ws/install
Creating symlink "/home/ubuntu/catkin_ws/src/CMakeLists.txt" pointing to "/opt/ros/noetic/share/catkin/cmake/toplevel.cmake"
####
#### Running command: "cmake /home/ubuntu/catkin_ws/src -DCATKIN_DEVEL_PREFIX=/home/ubuntu/catkin_ws/devel -DCMAKE_INSTALL_PREFIX=/home/ubuntu/catkin_ws/install -G Unix Makefiles" in "/home/ubuntu/catkin_ws/build"
####
```

⋮

```
[ 50%] Building CXX object pigeon_ros_tutorial/basic/basic_tutorial/CMakeFiles/basic_node.dir/src/basic.cpp.o
[100%] Linking CXX executable /home/ubuntu/catkin_ws/devel/lib/basic_tutorial/basic_node
[100%] Built target basic_node
ubuntu@ubuntu:~/catkin_ws$
```

∴ 컴파일 명령어는 워크 스페이스 경로에서 입력해야한다.



# ROS Node 생성

## 5. 실행

### 1) roscore 실행

```
ubuntu@ubuntu:~/catkin_ws$ roscore
... logging to /home/ubuntu/.ros/log/1f553ace-30f3-11ee-a6d7-39c151f3ecd1/ros-launch-ubuntu-9845.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://ubuntu:46049/
ros_comm version 1.16.0

SUMMARY
=====

PARAMETERS
* /rostdistro: noetic
* /rosversion: 1.16.0

NODES
```

# ROS Node 생성

---

## 5. 실행

2) 환경 설정 [워크 스페이스 환경에서 명령어 입력]

```
$ source devel/setup.bah
```

3) 실행

```
$ rosrun basic_tutorial basic_node
```

∴ 패키지 검색이 안된다면 아래 명령어를 입력

```
$ rospack find basic_tutorial
```

# ROS Node 생성

## 5. 실행

- Node 실행 결과

```
ubuntu@ubuntu:~/catkin_ws$ rosrunc basic_tutorial basic_node
[ INFO] [1690953206.782717590]: Hello world!
[ INFO] [1690953206.799495072]: Hello world!
[ INFO] [1690953206.816431705]: Hello world!
[ INFO] [1690953206.832759855]: Hello world!
[ INFO] [1690953206.849607581]: Hello world!
[ INFO] [1690953206.866108836]: Hello world!
[ INFO] [1690953206.882873785]: Hello world!
[ INFO] [1690953206.899377913]: Hello world!
[ INFO] [1690953206.916197473]: Hello world!
[ INFO] [1690953206.932578555]: Hello world!
[ INFO] [1690953206.949489446]: Hello world!
[ INFO] [1690953206.966350395]: Hello world!
```

# 감사합니다

구선생 로보틱스

