**Lesson 1: Instaling and using ROS(ROS Melodic Morenia)**

**1.1 Introducrtion**

ROS is an open-source, meta-operating system for your robot. It provides the services you would expect from an operating system, including hardware abstraction, low-level device control, implementation of commonly-used functionality, message-passing between processes, and package management. It also provides tools and libraries for obtaining, building, writing, and running code across multiple computers.

ROS currently only runs on Unix-based platforms. Software for ROS is primarily tested on Ubuntu and Mac OS X systems, though the ROS community has been contributing support for Fedora, Gentoo, Arch Linux and other Linux platforms.

**1.2 Installing ROS**

Setup your computer to accept software from packages.ros.org.



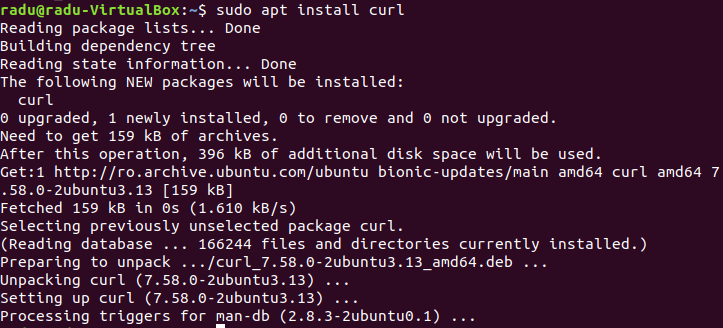
sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb\_release -sc) main" > /etc/apt/sources.list.d/ros-latest.list'

Set up your keys



curl -s https://raw.githubusercontent.com/ros/rosdistro/master/ros.asc | sudo apt-key add –

If you don’t have curl installed already,



sudo apt install curl

Update your packages

**sudo apt update**

There are many different libraries and tools in ROS.

**Desktop-Full Install: (Recommended)** : ROS, [rqt](http://wiki.ros.org/rqt), [rviz](http://wiki.ros.org/rviz), robot-generic libraries, 2D/3D simulators and 2D/3D perception

**sudo apt install ros-melodic-desktop-full**

**Desktop Install:**ROS, [rqt](http://wiki.ros.org/rqt), [rviz](http://wiki.ros.org/rviz), and robot-generic libraries

**sudo apt install ros-melodic-desktop**

**ROS-Base: (Bare Bones)** ROS package, build, and communication libraries. No GUI tools.

**sudo apt install ros-melodic-ros-base**

I recommand the Desktop-Full Install for the posibility to run every program.

****

For the variables to be added automatically to the bash, you need to use the next commands.



echo ”source /opt/ros/melodic/setup.bash” >> ~/.bashrc



source ~/.bashrc

Next step is to install some dependencies for building ROS packages.



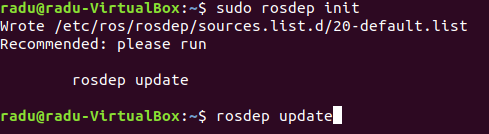
sudo apt install python-rosdep python-rosinstall python-rosinstall-generator python-wstool build-essential

If you want to compile your source or run some core components, you need to install and initialize rosdep.



sudo apt install python-rosdep

Initialize rosdep.



sudo rosdep init

rosdep update

**1.3 Creating a workspace for ROS**

mkdir -p ~/catkin\_ws/src

cd ~/catkin\_ws/

catkin\_make

source devel/setup.bash

The catkin\_make command is a convenience tool for working with catkin workspaces. Running it the first time in your workspace, it will create a CMakeLists.txt link in your 'src' folder.

To make sure your workspace is properly overlayed by the setup script, make sure ROS\_PACKAGE\_PATH environment variable includes the directory you're in.

echo $ROS\_PACKAGE\_PATH

should output:

/home/radu/catkin\_ws/src:/opt/ros/melodic/share

**This file MUST be sourced for every new terminal!**

Install RQT

Standard packages (rqt's core library + common plugins) can be installed by:

sudo apt-get install ros-melodic-rqt ros-melodic-rqt-common-plugins

Additionally, you can also install rqt\_robot\_plugins that provide features to be used when interacting with robots:

sudo apt-get install ros-melodic-rqt-robot-plugins

Upgrade installed packages

sudo apt-get update

sudo apt-get dist-upgrade

**1.4 Verify ROS Melodic installation**

Open a terminal press **ctrl-alt-T**

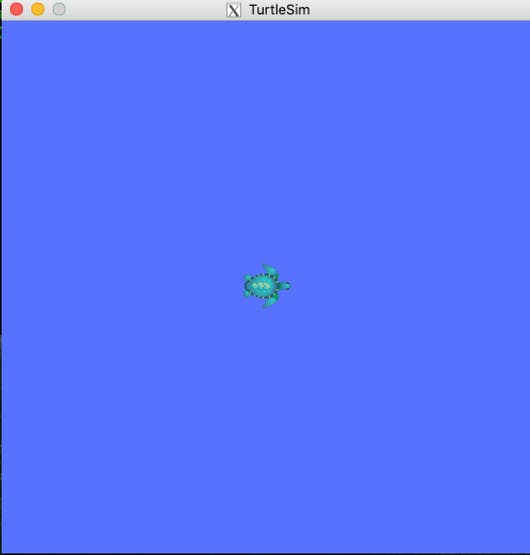
In the terminal run:

roscore

Open another terminal and run the simulation command:

rosrun turtlesim turtlesim\_node

If everything is correct, we will get the following screen:

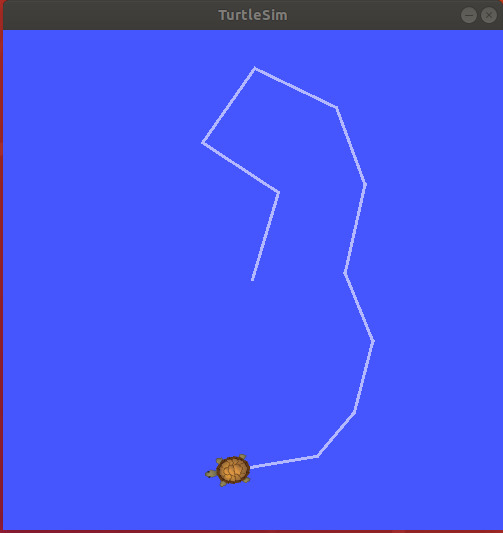


Open up yet another terminal window. Run the following:

rosrun turtlesim turtle\_teleop\_key

Use the arrow keys to move the turtle around the screen.

If everything goes right, you will obtain the following result on current terminal:

****

Close programs running in all of the terminals you’ve opened (CTRL+C).