

[🏠](#) / [Installation](#) / Ubuntu (deb packages)

You're reading the documentation for an older, but still supported, version of ROS 2. For information on the latest version, please have a look at [Kilted](#).

Ubuntu (deb packages)

Table of Contents

- [Resources](#)
- [Set locale](#)
- [Setup Sources](#)
- [Install ROS 2 packages](#)
- [Environment setup](#)
 - [Sourcing the setup script](#)
- [Try some examples](#)
 - [Talker-listener](#)
- [Next steps after installing](#)
- [Using the ROS 1 bridge](#)
- [Additional RMW implementations \(optional\)](#)
- [Troubleshooting](#)
- [Uninstall](#)

Deb packages for ROS 2 Humble Hawksbill are currently available for Ubuntu Jammy (22.04). The target platforms are defined in [REP 2000](#).

Resources

- Status Page:
 - ROS 2 Humble (Ubuntu Jammy): [amd64](#), [arm64](#)
- [Jenkins Instance](#)
- [Repositories](#)

Set locale

Make sure you have a locale which supports `UTF-8` . If you are in a minimal environment (such as a docker container), the locale may be something minimal like `POSIX` . We test with the following settings. However, it should be fine if you're using a different UTF-8 supported locale.

```
$ locale # check for UTF-8

$ sudo apt update && sudo apt install locales
$ sudo locale-gen en_US en_US.UTF-8
$ sudo update-locale LC_ALL=en_US.UTF-8 LANG=en_US.UTF-8
$ export LANG=en_US.UTF-8

$ locale # verify settings
```

Setup Sources

You will need to add the ROS 2 apt repository to your system.

First ensure that the [Ubuntu Universe repository](#) is enabled.

```
$ sudo apt install software-properties-common
$ sudo add-apt-repository universe
```

The [ros-apt-source](#) packages provide keys and apt source configuration for the various ROS repositories.

Installing the ros2-apt-source package will configure ROS 2 repositories for your system. Updates to repository configuration will occur automatically when new versions of this package are released to the ROS repositories.

```
$ sudo apt update && sudo apt install curl -y
$ export ROS_APT_SOURCE_VERSION=$(curl -s https://api.github.com/repos/ros-
infrastructure/ros-apt-source/releases/latest | grep -F "tag_name" | awk -F" '{print $4}')"
$ curl -L -o /tmp/ros2-apt-source.deb "https://github.com/ros-infrastructure/ros-apt-
source/releases/download/ ${ROS_APT_SOURCE_VERSION} /ros2-apt-
source_${ROS_APT_SOURCE_VERSION}.${(. /etc/os-release && echo $VERSION_CODENAME)_all.deb" # If
using Ubuntu derivatives use $UBUNTU_CODENAME
$ sudo dpkg -i /tmp/ros2-apt-source.deb
```

Install ROS 2 packages

Update your apt repository caches after setting up the repositories.

```
$ sudo apt update
```

ROS 2 packages are built on frequently updated Ubuntu systems. It is always recommended that you ensure your system is up to date before installing new packages.

```
$ sudo apt upgrade
```

⚠ Warning

Due to early updates in Ubuntu 22.04 it is important that `systemd` and `udev` -related packages are updated before installing ROS 2. The installation of ROS 2's dependencies on a freshly installed system without upgrading can trigger the **removal of critical system packages**.

Please refer to [ros2/ros2#1272](#) and [Launchpad #1974196](#) for more information.

Desktop Install (Recommended): ROS, RViz, demos, tutorials.

```
$ sudo apt install ros-humble-desktop
```

ROS-Base Install (Bare Bones): Communication libraries, message packages, command line tools. No GUI tools.

```
$ sudo apt install ros-humble-ros-base
```

Development tools: Compilers and other tools to build ROS packages

```
sudo apt install ros-dev-tools
```

Environment setup

Sourcing the setup script

Set up your environment by sourcing the following file.

```
$ source /opt/ros/humble/setup.bash
```

ⓘ Note

Replace `.bash` with your shell if you're not using bash. Possible values are: `setup.bash` , `setup.sh` , `setup.zsh` .

Making life easy:

This is meant to make your lab assignments easier.
Please perform the operations only once in one terminal,

```
$ echo "source /opt/ros/humble/setup.bash" >> ~/.bashrc
$ mkdir -p ~/ros2_ws/src
$ cd ~/ros2_ws
$ colcon build
$ echo "source ~/ros2_ws/install/setup.bash" >> ~/.bashrc
$ source ~/.bashrc
```

Try some examples

Talker-listener

If you installed `ros-humble-desktop` above you can try some examples.

In one terminal, source the setup file and then run a C++ `talker` :

```
$ source /opt/ros/humble/setup.bash
$ ros2 run demo_nodes_cpp talker
```

In another terminal source the setup file and then run a Python `listener` :

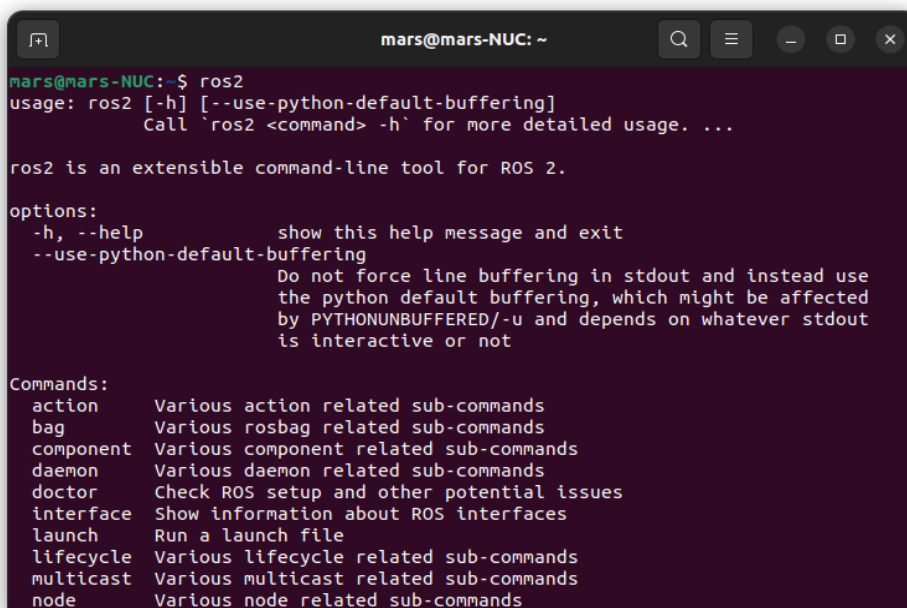
```
$ source /opt/ros/humble/setup.bash
$ ros2 run demo_nodes_py listener
```

You should see the `talker` saying that it's `Publishing` messages and the `listener` saying `I heard` those messages. This verifies both the C++ and Python APIs are working properly. Hooray!

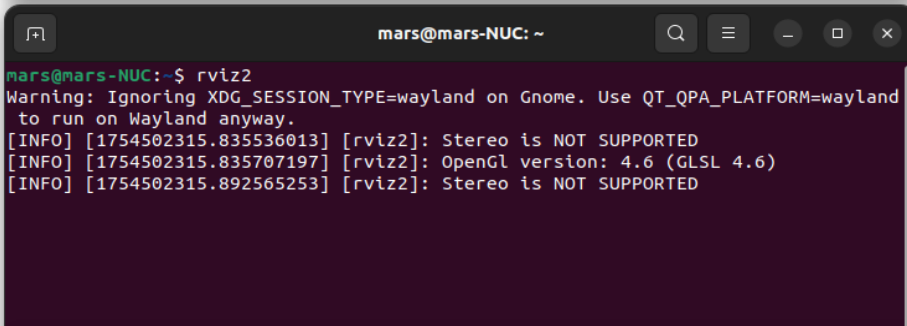
Submissions:

By the end of this assignment we require you to submit the following screenshots:

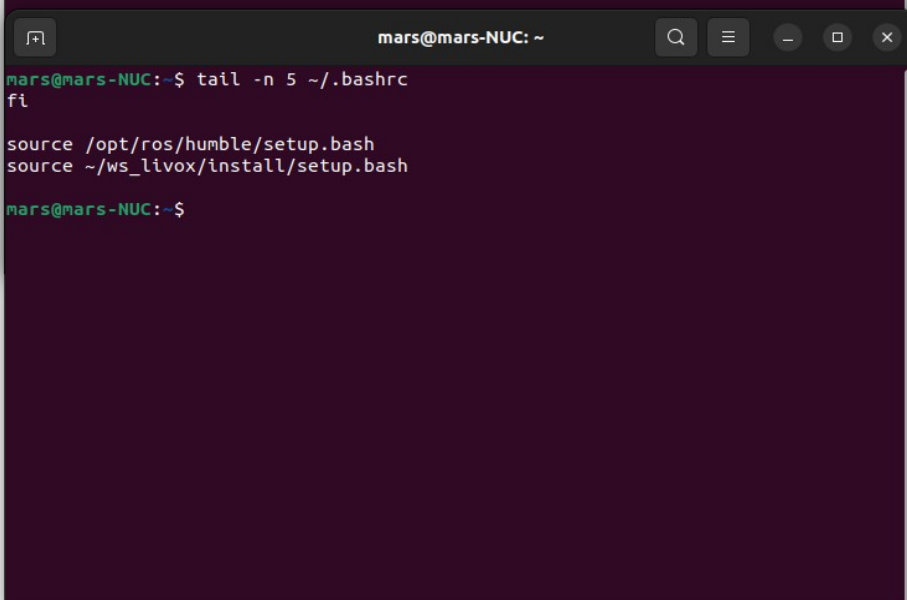
- 1> type “ros2” in terminal.
- 2> type “rviz2” in terminal.
- 3> type “tail -n 5 ~/.bashrc” in terminal.



```
mars@mars-NUC: ~  
mars@mars-NUC:~$ ros2  
usage: ros2 [-h] [--use-python-default-buffering]  
          Call 'ros2 <command> -h' for more detailed usage. ...  
  
ros2 is an extensible command-line tool for ROS 2.  
  
options:  
  -h, --help            show this help message and exit  
  --use-python-default-buffering  
                        Do not force line buffering in stdout and instead use  
                        the python default buffering, which might be affected  
                        by PYTHONUNBUFFERED/-u and depends on whatever stdout  
                        is interactive or not  
  
Commands:  
  action      Various action related sub-commands  
  bag         Various rosbag related sub-commands  
  component   Various component related sub-commands  
  daemon      Various daemon related sub-commands  
  doctor      Check ROS setup and other potential issues  
  interface   Show information about ROS interfaces  
  launch      Run a launch file  
  lifecycle   Various lifecycle related sub-commands  
  multicast   Various multicast related sub-commands  
  node        Various node related sub-commands
```



```
mars@mars-NUC: ~  
mars@mars-NUC:~$ rviz2  
Warning: Ignoring XDG_SESSION_TYPE=wayland on Gnome. Use QT_QPA_PLATFORM=wayland  
to run on Wayland anyway.  
[INFO] [1754502315.835536013] [rviz2]: Stereo is NOT SUPPORTED  
[INFO] [1754502315.835707197] [rviz2]: OpenGL version: 4.6 (GLSL 4.6)  
[INFO] [1754502315.892565253] [rviz2]: Stereo is NOT SUPPORTED
```



```
mars@mars-NUC: ~  
mars@mars-NUC:~$ tail -n 5 ~/.bashrc  
fi  
  
source /opt/ros/humble/setup.bash  
source ~/ws_livox/install/setup.bash  
  
mars@mars-NUC:~$
```

Upload all of
these images to
the classroom.