

Comprehensive Tutorial on Running ROS 2 Humble and Gazebo Ignition Fortress on Linux, Windows, and macOS

Contents

1	Introduction	1
2	Minimum System Requirements	1
3	Version Compatibility	1
4	Installation Instructions	2
4.1	Linux (Ubuntu 22.04 LTS)	2
	Install ROS 2 Humble (Binary)	2
	Install Gazebo Ignition Fortress (Binary)	4
4.2	Windows 10/11 with Windows Subsystem for Linux (WSL2)	5
	Setting up WSL2 with Ubuntu 22.04 LTS	5
	Verifying GUI Functionality on Ubuntu 22.04 with WSL2	9
	Launching Ubuntu 22.04 with WSL2 from Windows	9
	Installing ROS 2 Humble and Gazebo Ignition Fortress	10
	Uninstalling Ubuntu 22.04 from Windows System	11
4.3	macOS	12
	Install ROS 2 Humble (Source)	12
	Install Gazebo Ignition Fortress (Source)	13
5	Exploring Cyclone DDS as an RMW Implementation (optional)	14
5.1	Installing Eclipse Cyclone DDS	14
	Switching to Cyclone DDS	14
	Testing Eclipse Cyclone DDS	14
6	Conclusion	14
7	Additional Resources	14

1 Introduction

Welcome to this comprehensive tutorial, covering the installation and running of ROS 2 Humble and Gazebo Ignition Fortress on Linux (Ubuntu 22.04), Windows 10/11 with WSL2, and macOS. This guide serves as an all-in-one resource.

2 Minimum System Requirements

- **Processor:** Intel Core i5 (or equivalent)
- **RAM:** 8GB minimum, 16GB recommended
- **Disk Space:** 20GB minimum
- **Operating System:** Ubuntu Jammy (for ROS 2 Humble), Windows 10/11, macOS Catalina or newer

3 Version Compatibility

For this tutorial, the versions being used are ROS 2 Humble and Gazebo Ignition Fortress, which are compatible with each other.

4 Installation Instructions

4.1 Linux (Ubuntu 22.04 LTS)

Install ROS 2 Humble (Binary)

The following steps detail the installation instructions on Ubuntu 22.04 LTS, which is the most recommended environment for using ROS 2 Humble and Gazebo Fortress.

Set Locale

Configuring the system's locale ensures consistent and correct software behavior. The following command checks the system's current locale configuration.

```
locale # check for UTF-8
```

Inspect the output; if you see UTF-8, the locale is already appropriately set.

```
mohamed@Udacity:~$ locale
LANG=en_US.UTF-8
LANGUAGE=
LC_CTYPE="en_US.UTF-8"
LC_NUMERIC="en_US.UTF-8"
LC_TIME="en_US.UTF-8"
LC_COLLATE="en_US.UTF-8"
LC_MONETARY="en_US.UTF-8"
LC_MESSAGES="en_US.UTF-8"
LC_PAPER="en_US.UTF-8"
LC_NAME="en_US.UTF-8"
LC_ADDRESS="en_US.UTF-8"
LC_TELEPHONE="en_US.UTF-8"
LC_MEASUREMENT="en_US.UTF-8"
LC_IDENTIFICATION="en_US.UTF-8"
LC_ALL=en_US.UTF-8
```

If UTF-8 is not present, these commands help configure the locale to the UTF-8 standard.

```
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
```

Post-setup, rerun the locale command to ensure proper configuration.

Setup Sources

Prepare the system for ROS 2 installation by adding necessary software tools and incorporating Ubuntu's official ROS 2 repository, ensuring trusted and up-to-date package access.

Ensure your system has the necessary tools and repositories for ROS 2 installation.

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

Next, add the ROS 2 GPG key, ensuring the authenticity of downloaded packages.

```
sudo apt update && sudo apt install curl -y
sudo curl -sSL https://raw.githubusercontent.com/ros/rosdistro/master/ros.key -o /usr/share/keyrings/ros-archive-keyring.gpg
```

Then, integrate the ROS 2 repository with your system for easy package access.

```
echo "deb [arch=$(dpkg --print-architecture)] signed-by=/usr/share/keyrings/ros-archive-keyring.gpg
http://packages.ros.org/ros2/ubuntu $(. /etc/os-release && echo $UBUNTU_CODENAME) main" | sudo tee /etc/apt/sources.list.d/ros2.list > /dev/null
```

Install ROS 2 Packages

With repositories set up, now install the ROS 2 Humble packages.

```
sudo apt update
sudo apt upgrade
sudo apt install ros - humble - desktop
```

Note: The installation of ros-humble-desktop requires over 2 GB of storage.

```
Need to get 437 MB of archives.
After this operation, 2,067 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Environment Setup

Configuring your environment correctly is essential to integrate ROS 2 functionalities within terminal sessions. By sourcing the setup script, you ensure that all ROS 2 commands are accessible from the terminal:

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

For a more permanent solution, adding this source command to your ‘.bashrc’ file is recommended. Doing so ensures that ROS 2 commands are readily available every time you launch a new terminal. Here’s how you can achieve that:

```
echo "source /opt/ros/humble/setup.bash" >> ~/.bashrc
```

The same command is shown in the next screenshot.

Verify ROS 2 Installation with Demo Nodes

After installation, test ROS 2 by running these example nodes (**in two different terminals**). In

the first terminal, start the demo talker node.

```
ros2 run demo_nodes_cpp talker
```

This node will begin publishing messages.

```
mohamed@Udacity:~$ ros2 run demo_nodes_cpp talker
[INFO] [1696472471.648449771] [talker]: Publishing: 'Hello World: 1'
[INFO] [1696472472.648370444] [talker]: Publishing: 'Hello World: 2'
[INFO] [1696472473.648348641] [talker]: Publishing: 'Hello World: 3'
[INFO] [1696472474.648325103] [talker]: Publishing: 'Hello World: 4'
[INFO] [1696472475.648280689] [talker]: Publishing: 'Hello World: 5'
```

In the second terminal, start the demo listener node.

```
ros2 run demo_nodes_py listener
```

This node listens to and prints out the published messages.

```
mohamed@Udacity:~$ ros2 run demo_nodes_py listener
[INFO] [1696476994.921504527] [listener]: I heard: [Hello World: 5]
[INFO] [1696476995.903328434] [listener]: I heard: [Hello World: 6]
[INFO] [1696476996.903362388] [listener]: I heard: [Hello World: 7]
[INFO] [1696476997.903278854] [listener]: I heard: [Hello World: 8]
[INFO] [1696476998.903570937] [listener]: I heard: [Hello World: 9]
[INFO] [1696476999.903472545] [listener]: I heard: [Hello World: 10]
```

If the listener prints the messages from the talker, it indicates that ROS 2 is correctly installed and functioning. This successful communication between the nodes signifies that your ROS 2 setup is ready for further exploration and development!

Uninstallation

⚠ **Caution:** The following instructions are provided for comprehensive knowledge and should be used with care. It’s typically unnecessary to uninstall ROS 2 at this point, but this will let you know how to uninstall ROS 2 if needed.

To completely remove ROS 2 from your system, execute the following commands.

```
sudo apt remove ~nros - humble -* && sudo apt autoremove
```

Install Gazebo Ignition Fortress (Binary)

This section provides steps to install Gazebo Ignition Fortress using binary packages.

Preparation: Installing Essential Tools

Begin by updating your system’s package list. Then, install the tools essential for the subsequent installation steps.

```
sudo apt - get update
sudo apt - get install lsb - release wget gnupg
```

Installing Ignition Fortress:

To install Ignition Fortress, you’ll first add the OSRF repository key and its repository to your system. Follow the steps below:

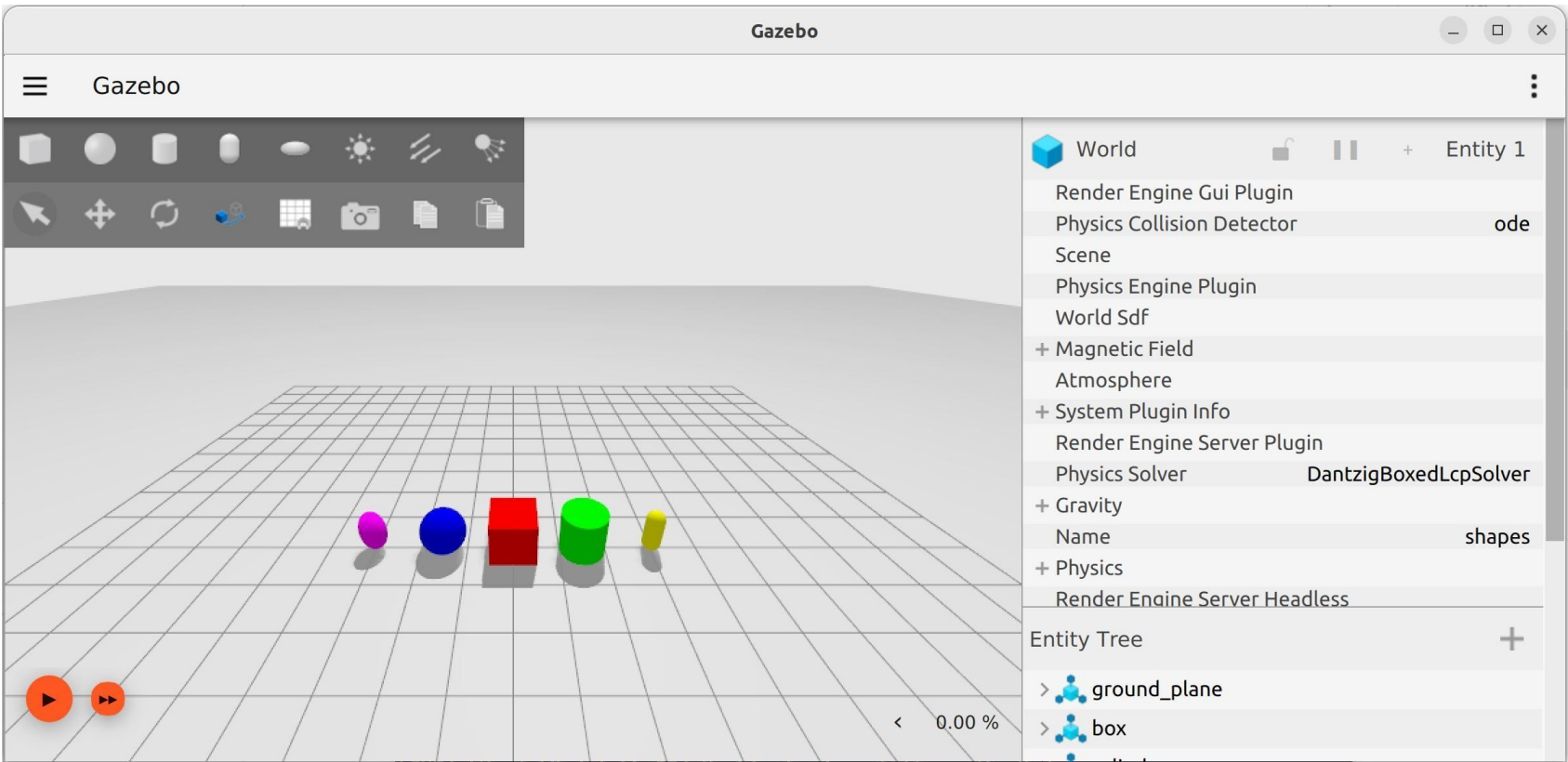
```
sudo wget https://packages.osrfoundation.org/gazebo.gpg -O /usr/share/keyrings/pkgs-osrf-archive-keyring.gpg
echo "deb [ arch=$( dpkg -- print - architecture ) signed - by =/usr/share/keyrings/pkgs-osrf-archive-keyring.gpg] http://packages.osrfoundation.org/gazebo/ubuntu-stable $( lsb_release - cs) main " | sudo tee /etc/apt/sources.list.d/gazebo-stable.list >/dev/null
sudo apt - get update
sudo apt - get install ignition - fortress
```

Verify Gazebo Installation with Demo World

After successfully installing Gazebo, launch the simulator with the demo shapes world with the command:

```
ign gazebo -v 4 shapes.sdf
```

Then, Gazebo Fortress will be launched with the shapes world, as shown below.



Successfully launching Gazebo Fortress GUI indicates the simulator is ready for the exciting projects in your Robotics ND journey!

Uninstallation

☞ Should you wish to remove Gazebo Fortress from your system, execute the commands outlined below.

```
sudo apt remove ignition - fortress && sudo apt autoremove
```


4.2 Windows 10/11 with Windows Subsystem for Linux (WSL2)

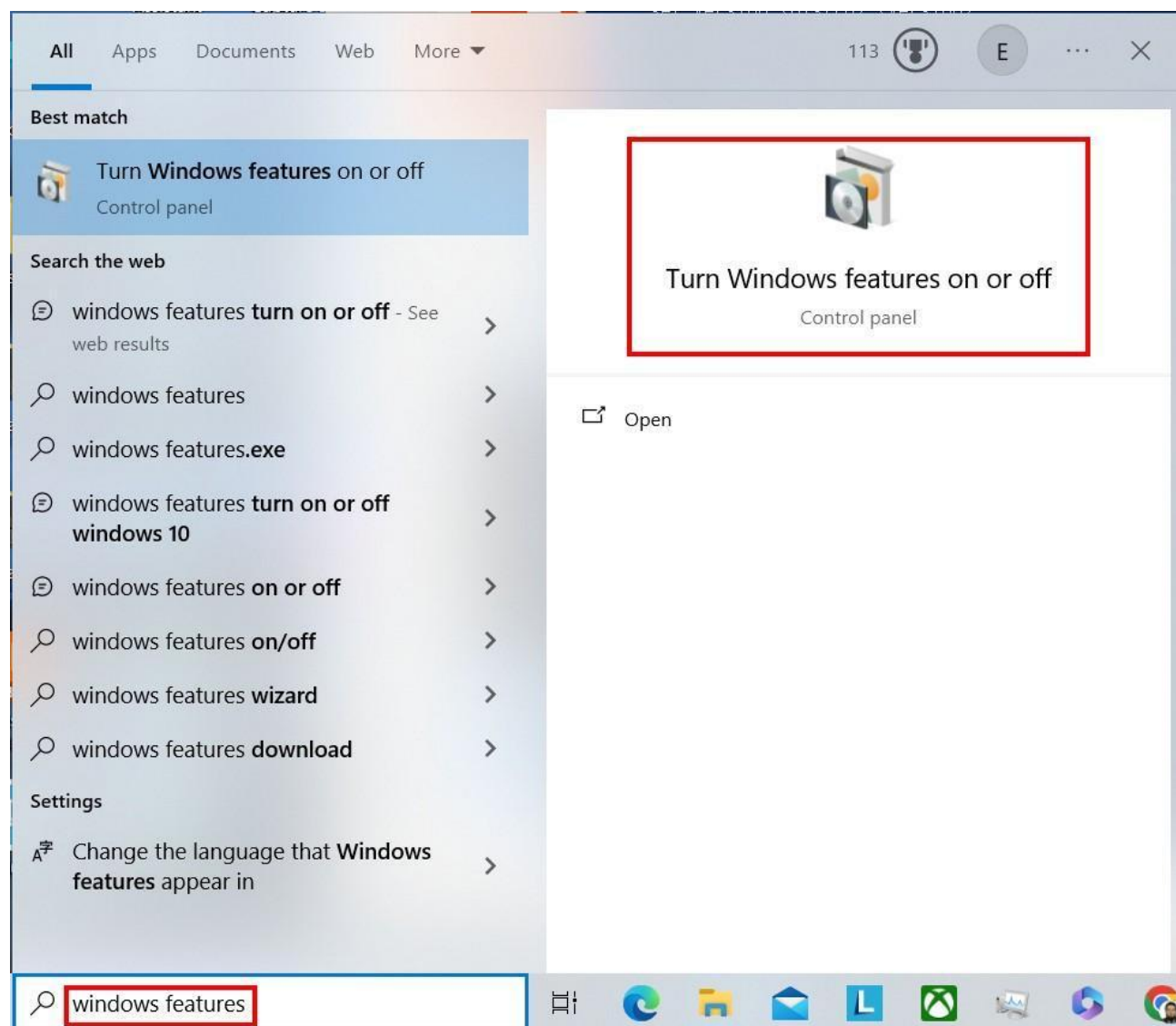
This section provides step-by-step instructions for setting Windows Subsystem for Linux 2 (WSL2) with Ubuntu 22.04 LTS.

Setting up WSL2 with Ubuntu 22.04 LTS

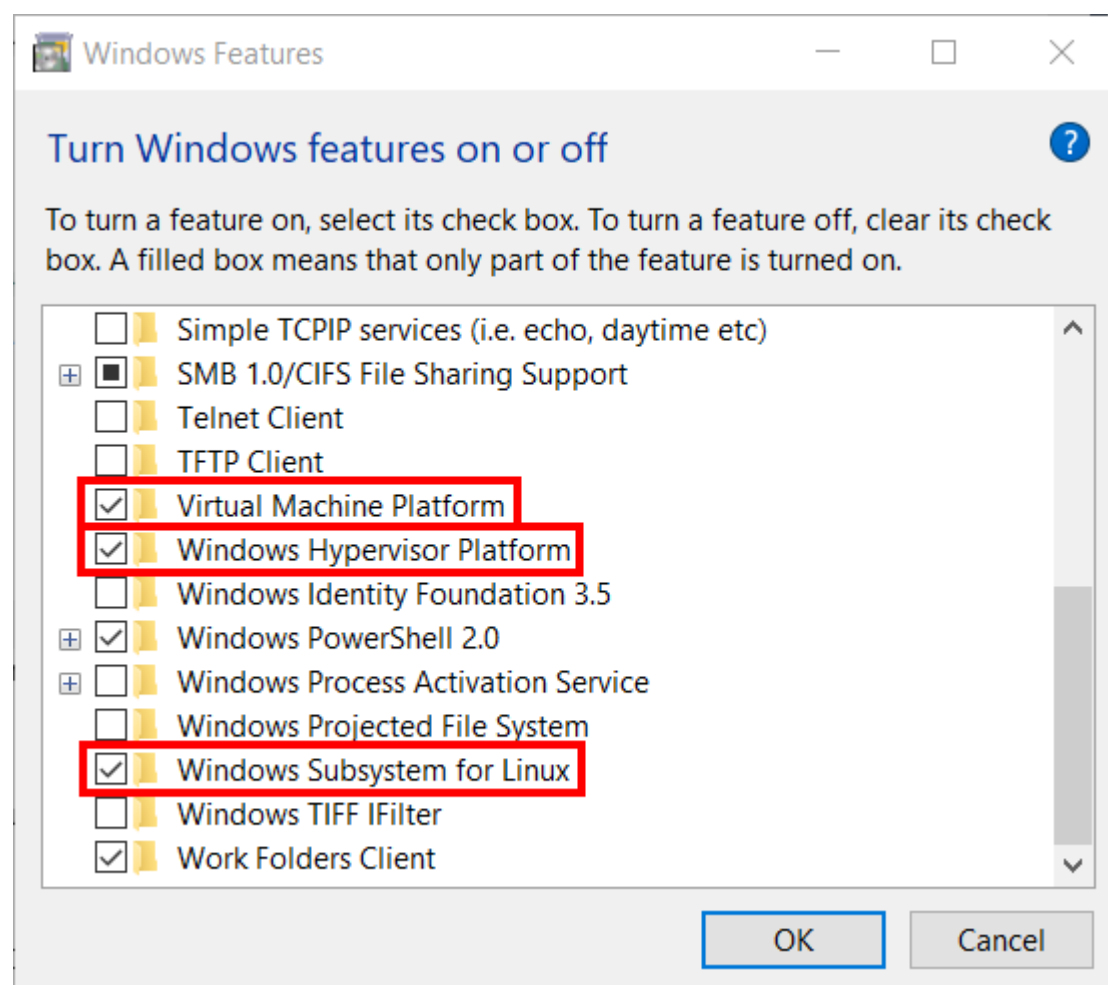
Follow the next steps to set up WSL2 and install Ubuntu 22.04 LTS.

Enable Necessary Windows Features:

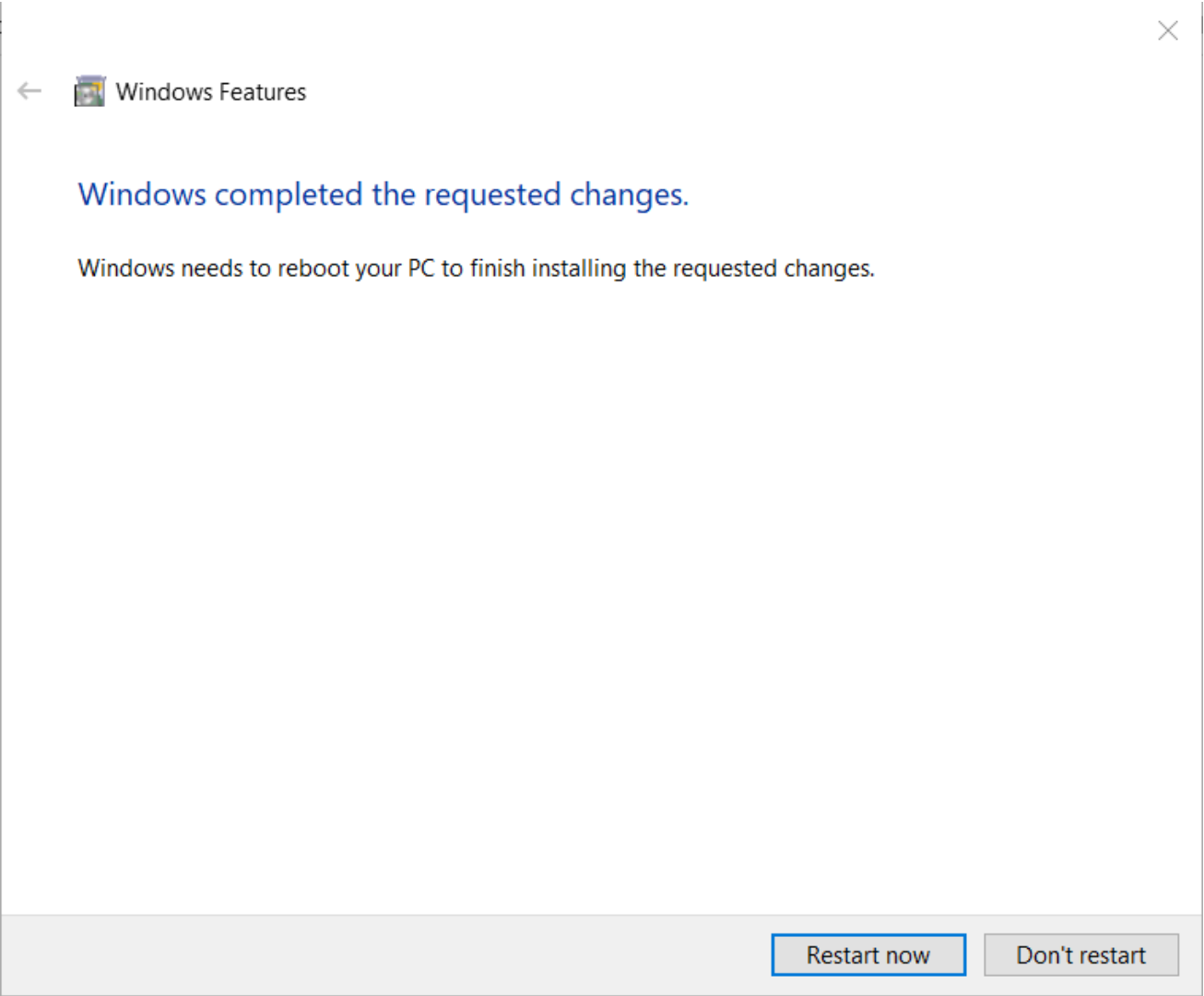
From the Start Menu, search for **Windows Features** and select the **Turn Windows features on or off**.



From the **Windows Features**, enable the **Virtual Machine Platform**, **Windows Hypervisor Platform**, and the **Windows Subsystem for Linux** features, as shown in the following figure.

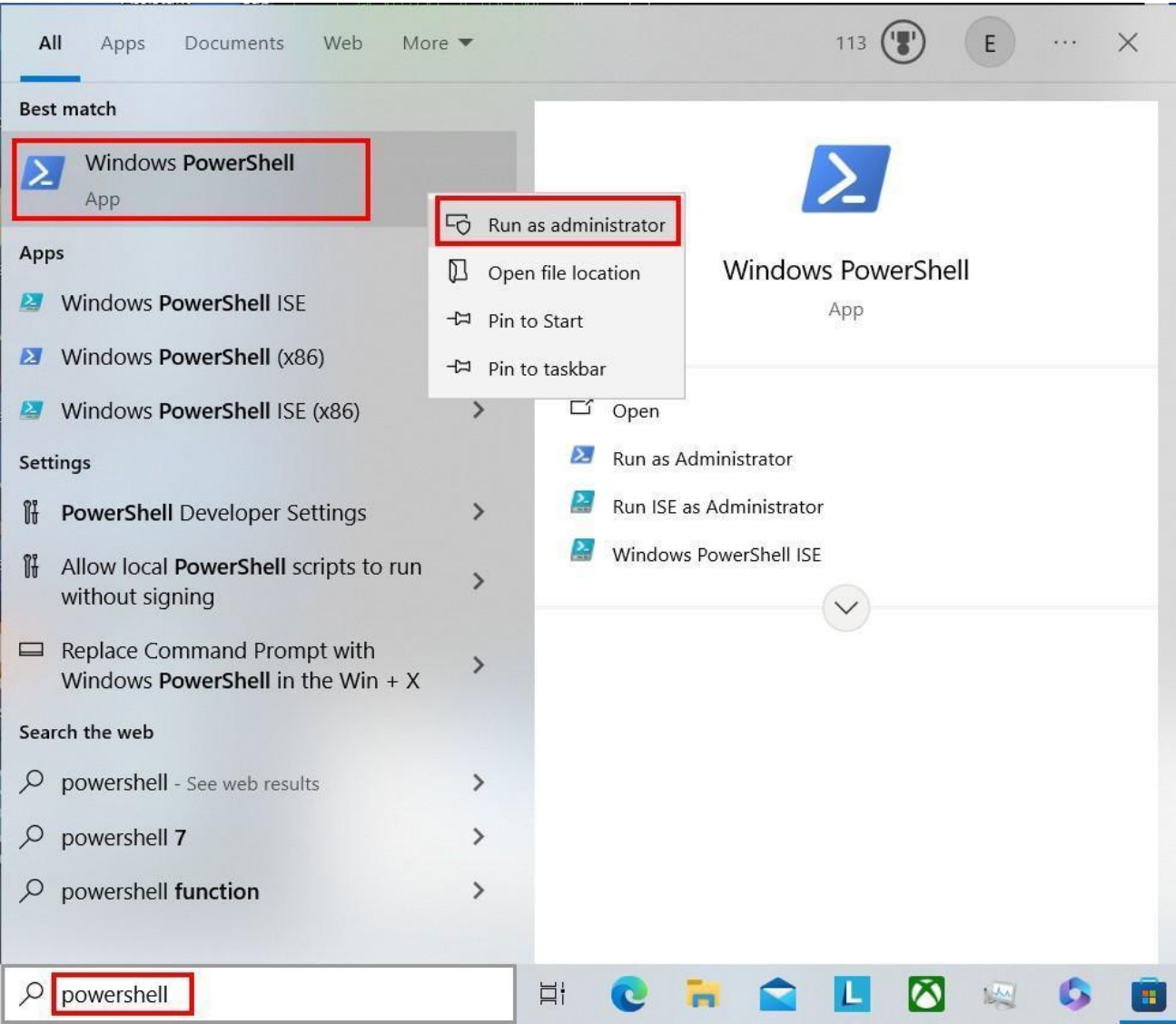


Windows will search for the required files and then prompt you for a restart. Please restart the system to apply the requested changes.



Update the WSL and Set the Default Version to 2:

From the Start menu, search about **Power Shell** and open it as **Administrator**.



Run the following commands to set the default version to 2 and update WSL.

```
wsl -- set - default - version 2 wsl
-- update
wsl -- shutdown
```

The same commands are shown below, but the update may take longer on your system.

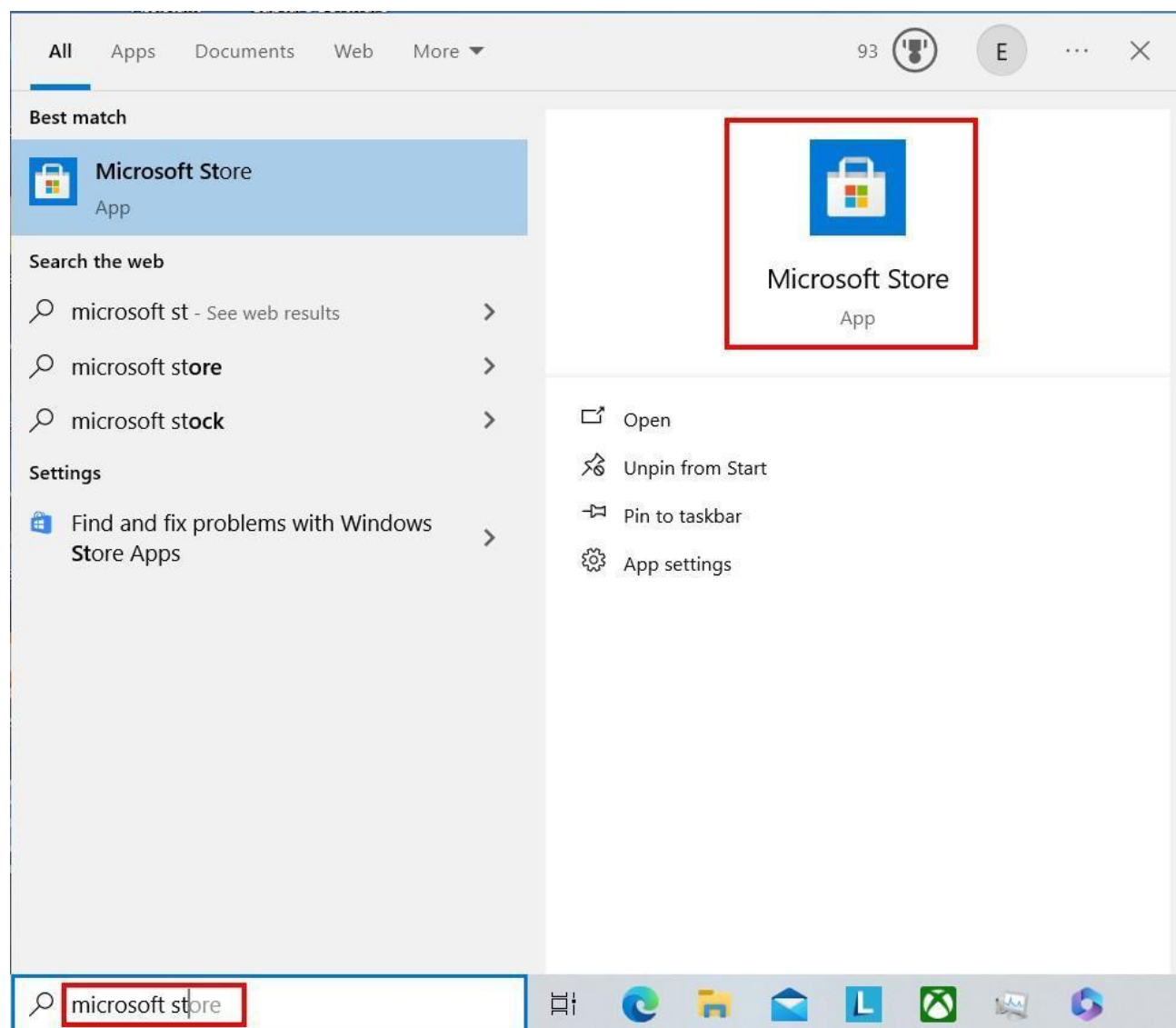
```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

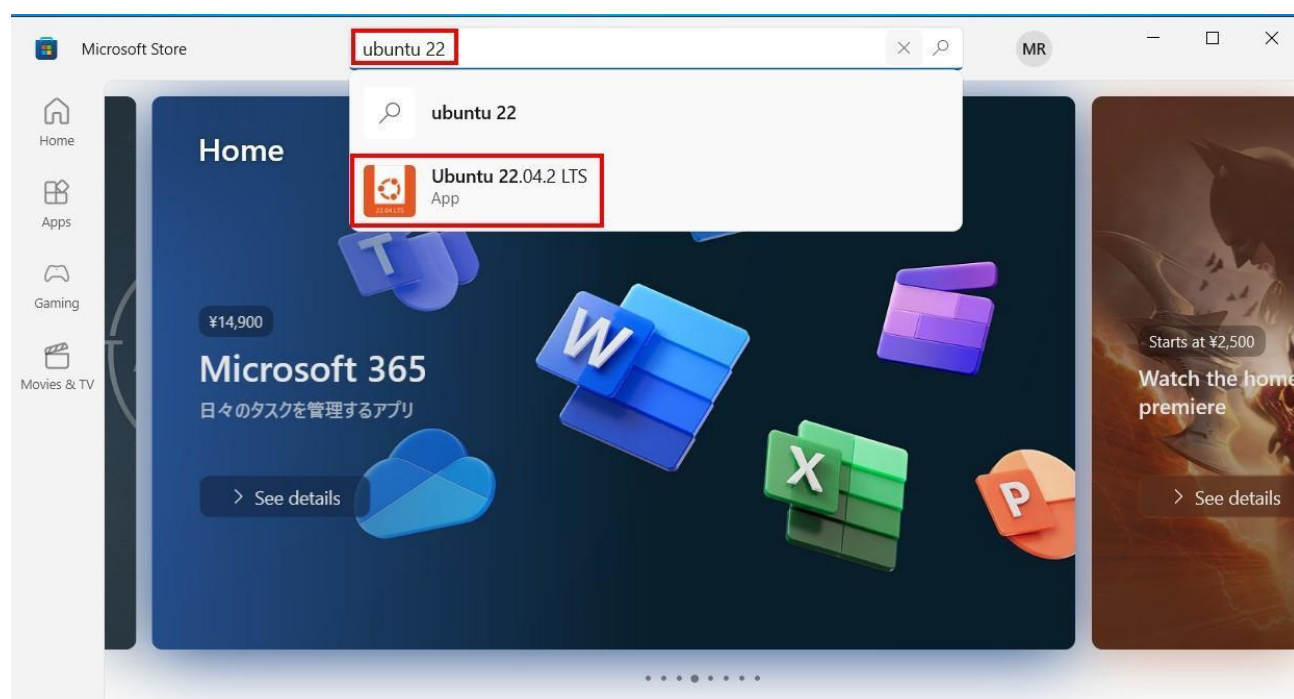
PS C:\WINDOWS\system32> wsl --set-default-version 2
For information on key differences with WSL 2 please visit https://aka.ms/wsl2
The operation completed successfully.
PS C:\WINDOWS\system32> wsl --update
Checking for updates.
The most recent version of Windows Subsystem for Linux is already installed.
PS C:\WINDOWS\system32> wsl --shutdown
PS C:\WINDOWS\system32>
```

Install Ubuntu 22.04 LTS from the Microsoft Store:

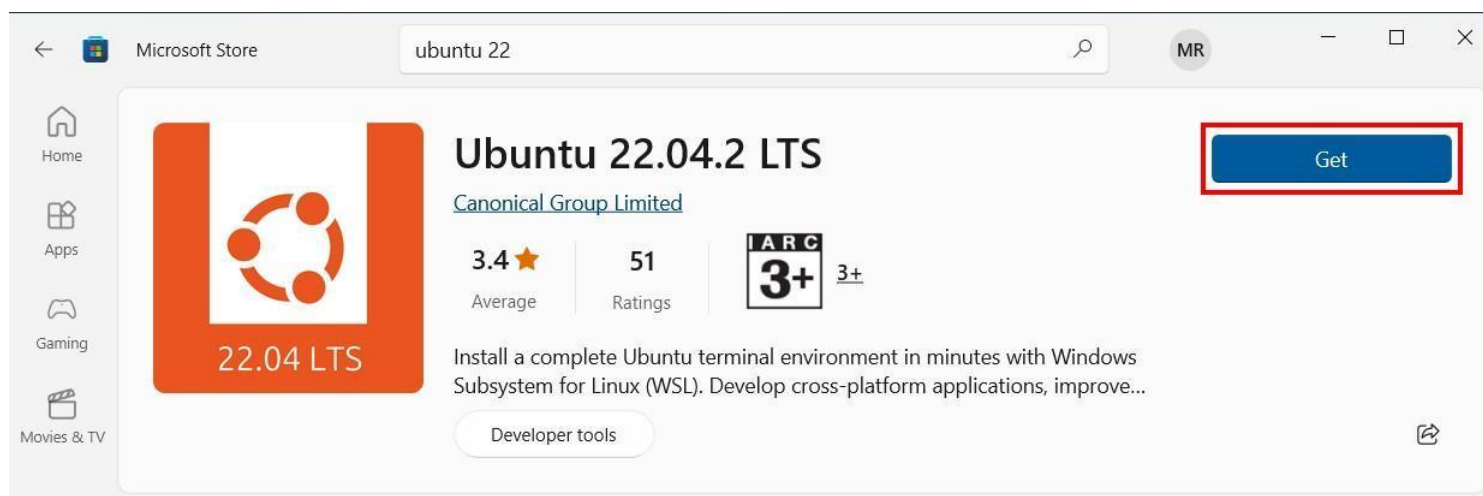
From the Start Menu, search for **Microsoft Store** and select it.



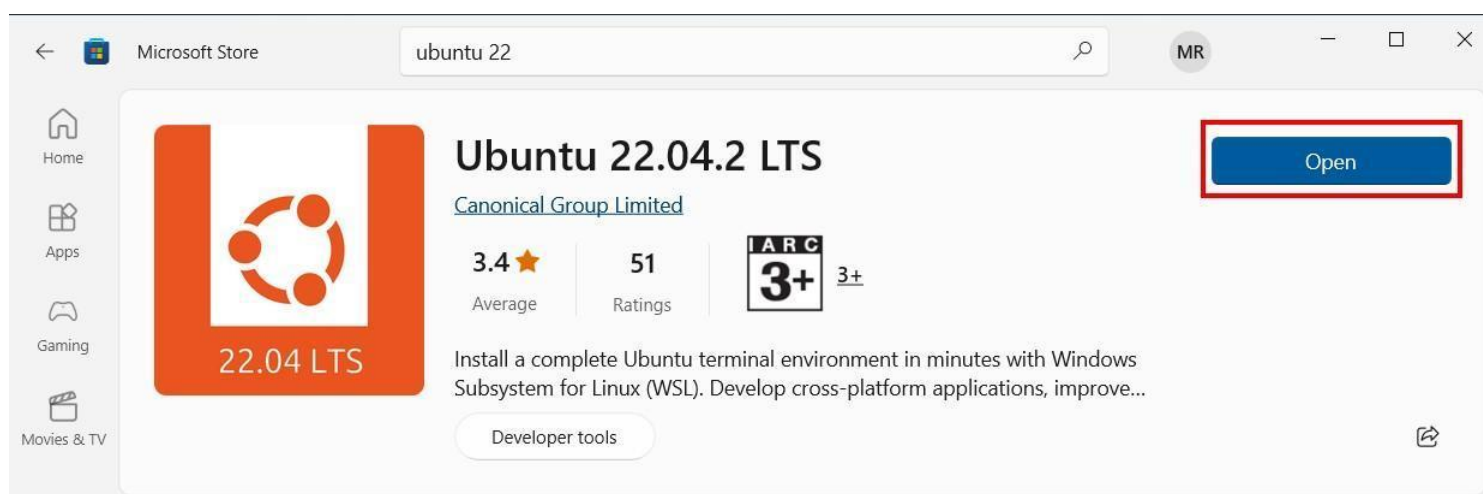
Search for **Ubuntu 22.04 LTS** in the **Microsoft Store** and select it.



After clicking on the **Ubuntu 22.04 LTS**, download it by pressing the **Get** button.



The download process will start, but notice this is not the installation yet. Once the download is complete, press the **Open** button to start your instance of the Ubuntu 22.04 LTS. This is the step when the installation is done.



It will take a few seconds and then prompt you for your username and password. Fill them in as follows, and your first terminal from the Ubuntu machine is ready to be used.

```
udacity@LAPTOP-JLIGL6NF: ~  
Installing, this may take a few minutes...  
Please create a default UNIX user account. The username does not need to match your Windows username.  
For more information visit: https://aka.ms/wslusers  
Enter new UNIX username: udacity  
New password:  
Retype new password:  
passwd: password updated successfully  
Installation successful!  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 4.4.0-19041-Microsoft x86_64)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:        https://ubuntu.com/advantage  
  
This message is shown once a day. To disable it please create the  
/home/udacity/.hushlogin file.  
udacity@LAPTOP-JLIGL6NF:~$
```

Install Dependency to Enable GUI:

To utilize GUI applications within Ubuntu 22.04 running on WSL2, we must install X11, the Linux Windowing System, along with its associated utilities. Here are the steps to achieve this:

```
sudo apt update  
sudo apt install x11 - apps
```

Note: Installing the x11-apps package enables the display of GUI applications, but it won't immediately launch any GUI interface. To validate the GUI setup, we will use an Ubuntu GUI application as a test case. In the subsequent section, we will demonstrate this by installing and running the gnome-calculator.

Verifying GUI Functionality on Ubuntu 22.04 with WSL2

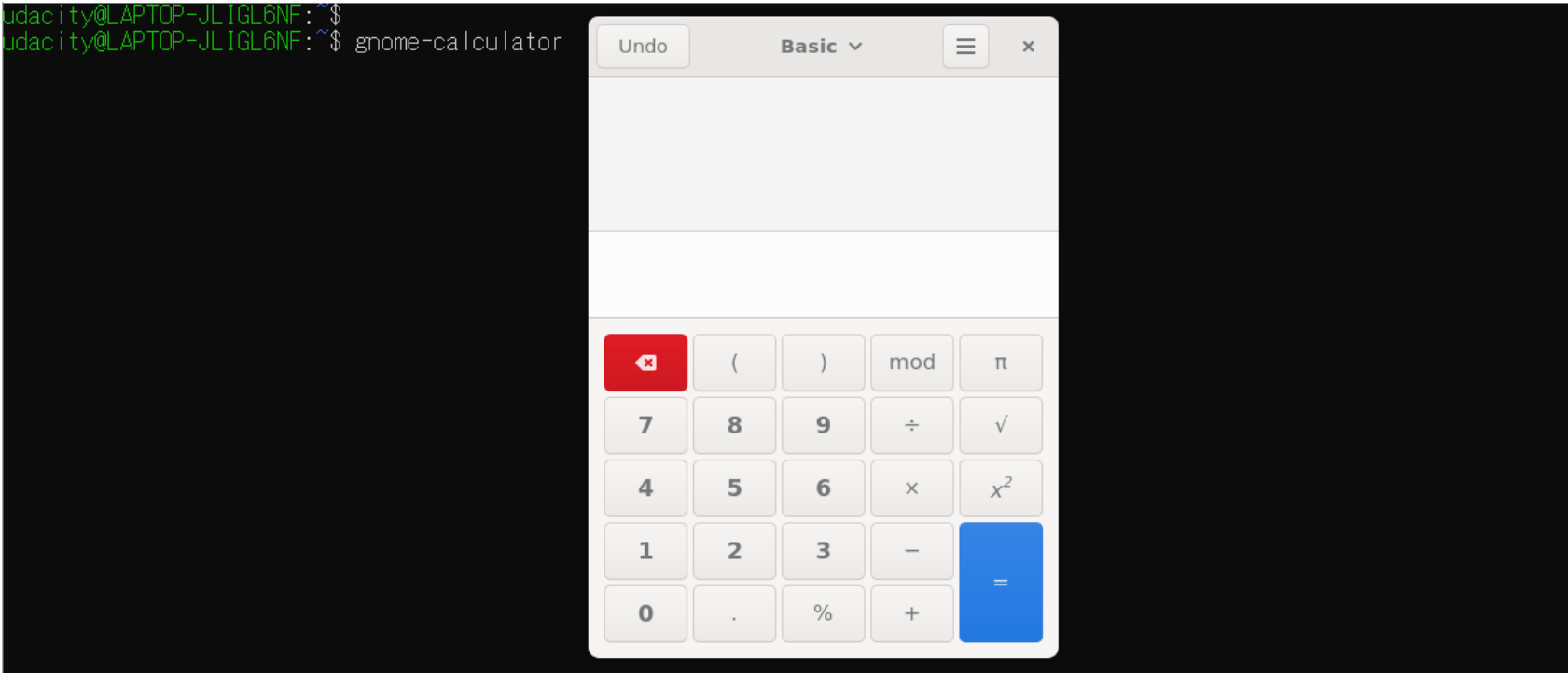
To confirm that GUI applications operate as expected, we'll install the `gnome-calculator`.

```
sudo apt install gnome - calculator
```

Once the calculator is installed, launch it using the following command.

```
gnome - calculator
```

If the GUI setup is successful, the `gnome-calculator` application should display seamlessly as shown below.

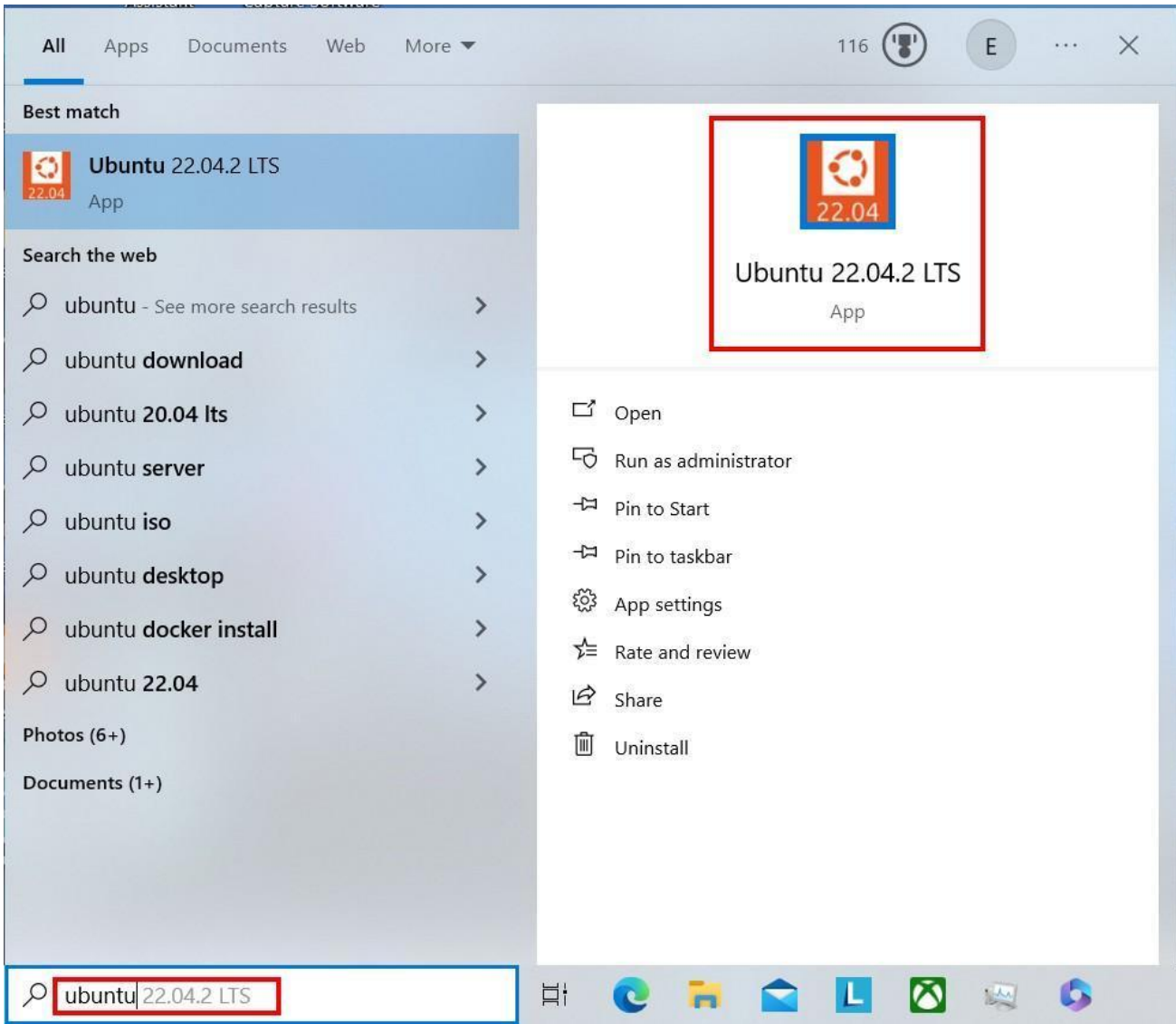


Launching Ubuntu 22.04 with WSL2 from Windows

By now, we have walked through the process of installing Ubuntu 22.04 on WSL2 and ensuring GUI applications function correctly. An essential final step is understanding how to initiate new terminal sessions in Ubuntu 22.04 with this setup.

To begin a new session, follow these straightforward steps:

1. Navigate to the Windows Start Menu.
2. Search for "Ubuntu 22.04."
3. Once the Ubuntu 22.04 option appears in the search results, click on it.
4. This action will open a new terminal session, similar to the one shown above where we launched the calculator.



Installing ROS 2 Humble and Gazebo Ignition Fortress

Once you have WSL2 with Ubuntu 22.04 set up, you can refer to this section to install ROS 2 Humble and to this for Gazebo Ignition Fortress as explained in the steps for Ubuntu 20.04 provided earlier.

Verify ROS 2 Installation on WSL2

After installing ROS2, verify its functionality by running the following example nodes. Open **(two different terminals)** from the Start Menu as described in the previous section 4.2.

In the first terminal, start the demo talker node.

```
ros2 run demo_nodes_cpp talker
```

This node will begin publishing messages.

```
udacity@LAPTOP-JL1GL6NF:~$  
udacity@LAPTOP-JL1GL6NF:~$ ros2 run demo_nodes_cpp talker  
[INFO] [1696600400.522390594] [talker]: Publishing: 'Hello World: 1'  
[INFO] [1696600401.522453946] [talker]: Publishing: 'Hello World: 2'  
[INFO] [1696600402.522337397] [talker]: Publishing: 'Hello World: 3'  
[INFO] [1696600403.522593449] [talker]: Publishing: 'Hello World: 4'  
[INFO] [1696600404.522590501] [talker]: Publishing: 'Hello World: 5'
```

In the second terminal, start the demo listener node.

```
ros2 run demo_nodes_py listener
```

This node listens to and prints out the published messages.

```
udacity@LAPTOP-JL1GL6NF:~$  
udacity@LAPTOP-JL1GL6NF:~$ ros2 run demo_nodes_py listener  
[INFO] [1696600463.108017075] [listener]: I heard: [Hello World: 7]  
[INFO] [1696600464.098880544] [listener]: I heard: [Hello World: 8]  
[INFO] [1696600465.098965114] [listener]: I heard: [Hello World: 9]  
[INFO] [1696600466.097635983] [listener]: I heard: [Hello World: 10]  
[INFO] [1696600467.098860453] [listener]: I heard: [Hello World: 11]  
[INFO] [1696600468.097141923] [listener]: I heard: [Hello World: 12]
```

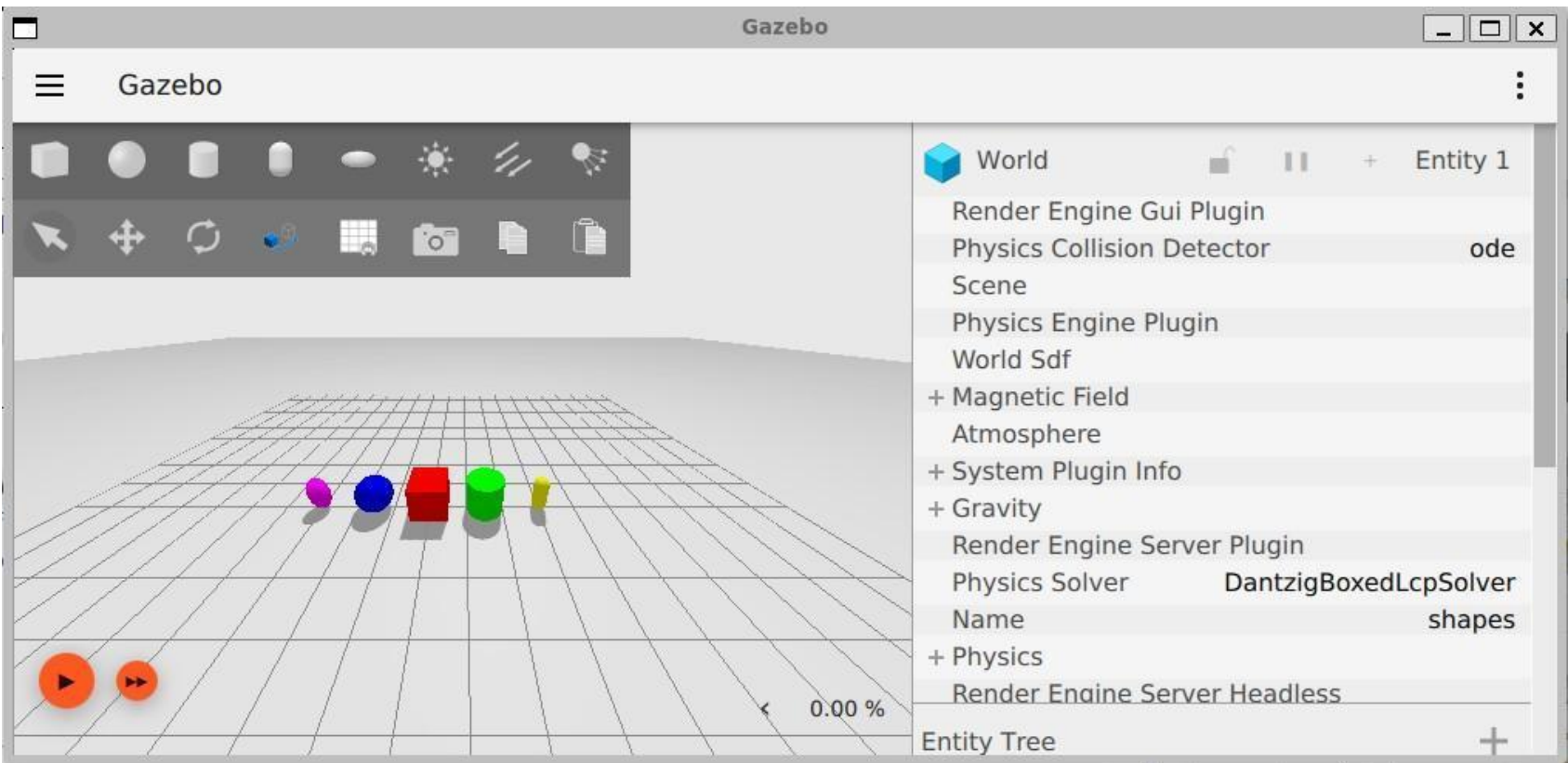
If the listener prints the messages from the talker, it indicates that ROS 2 is correctly installed using WSL2 and functioning. This successful communication between the nodes signifies that your ROS 2 setup is ready for further exploration and development!

Verify Gazebo Installation on WSL2

Note: After successful installation, launch the Gazebo simulator with the command:

```
ign gazebo -v 4 shapes.sdf
```

Then, Gazebo Fortress will be launched with the shapes world, as shown below.



Seamlessly launching Gazebo Fortress GUI indicates the simulator is ready for the exciting projects in your Robotics ND journey!

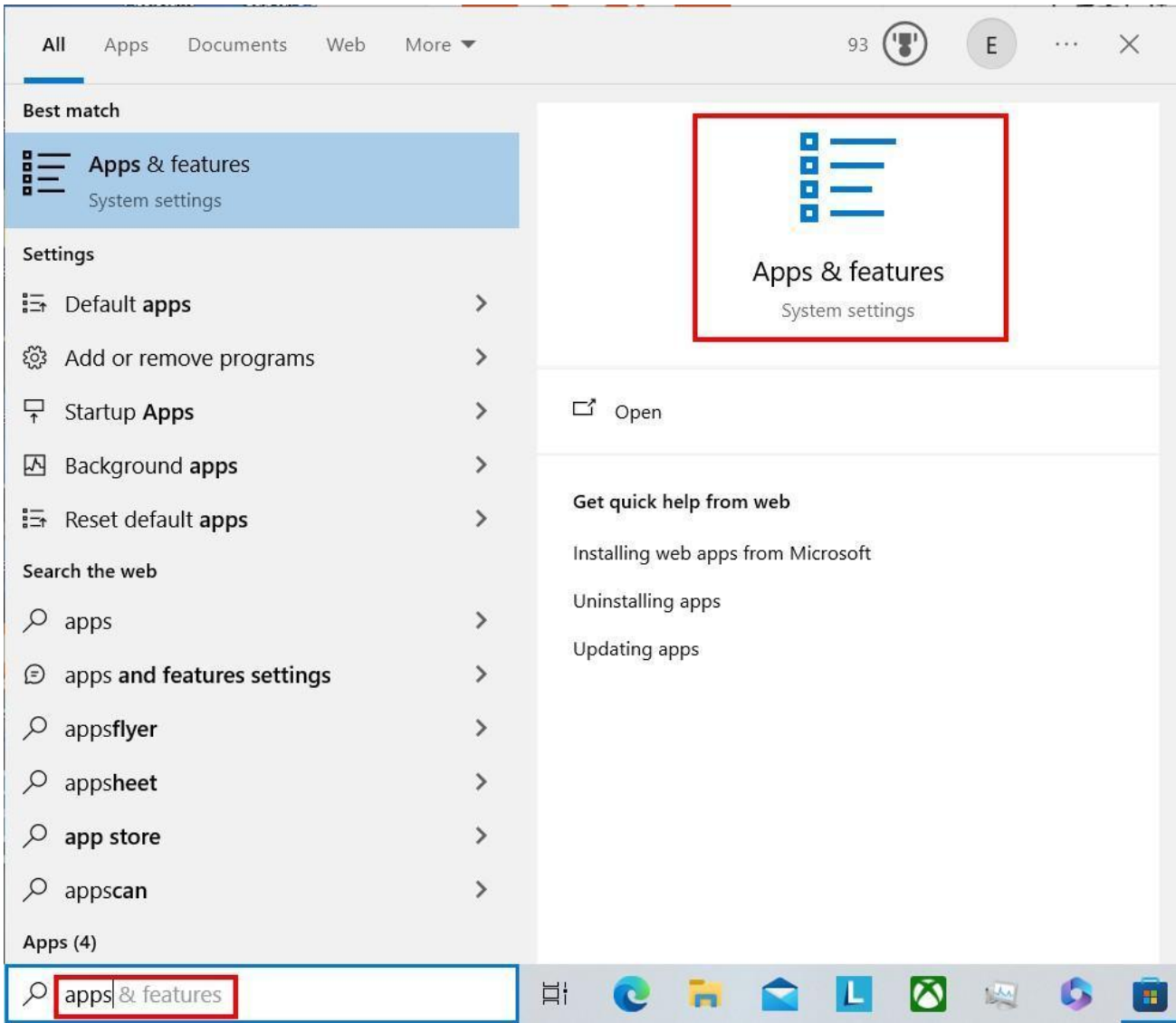
Uninstalling ROS 2 Humble and Gazebo Ignition Fortress

When using WSL 2, we can uninstall ROS 2 or Gazebo, like when using Ubuntu 22.04. This section explains how to uninstall ROS 2, and this one does the same for Gazebo.

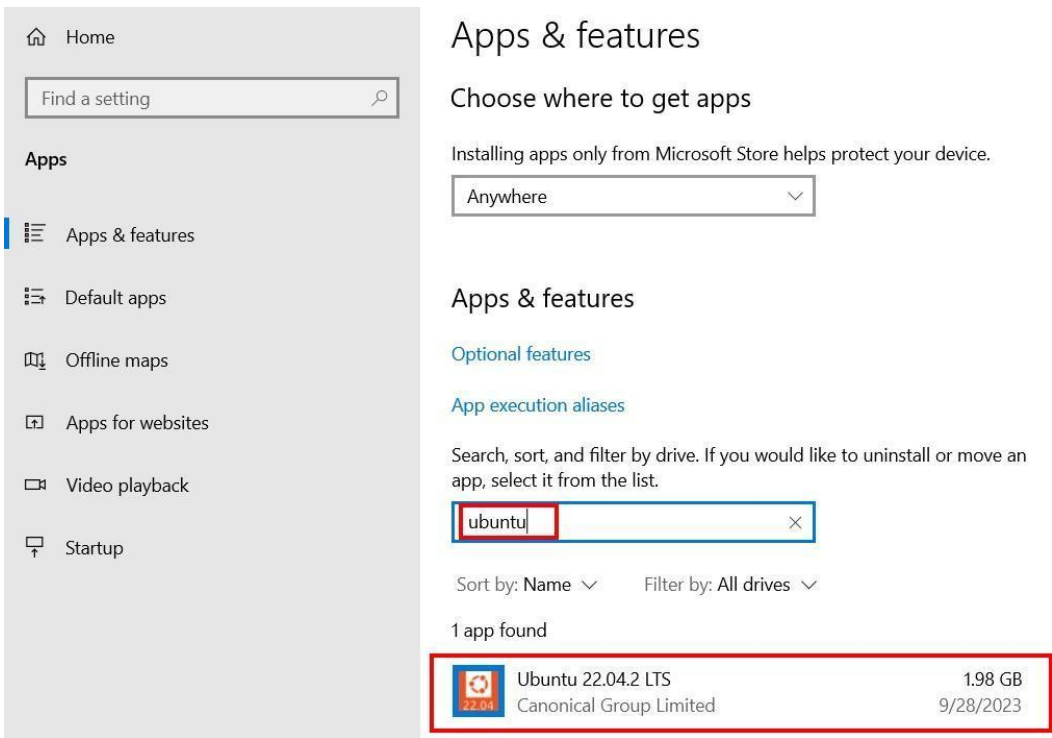
Uninstalling Ubuntu 22.04 from Windows System

If you need to uninstall Ubuntu 22.04 entirely from your Windows system, follow the next steps.

From the Start Menu, search for **Apps & features**, and open it.



Search about Ubuntu 22.04, select it, and press uninstall as shown below to remove it from your Windows environment.



4.3 macOS

This section provides installation instructions for ROS 2 Humble and Gazebo Ignition Fortress on macOS.

Install ROS 2 Humble (Source)

System Requirements

ROS 2 currently supports macOS Mojave (10.14).

Install Prerequisites

You need the following things installed to build ROS 2:

Xcode:

If you don't already have it installed, install Xcode. Xcode is a necessary tool for software development on macOS, providing a robust set of features for development. **Note:** Versions of Xcode later than 11.3.1 can no longer be installed on macOS Mojave, so you must manually install an older version. More details can be found [here](#).

```
xcode - select -- install
sudo xcode - select -- switch / Applications/ Xcode . app / Contents/ Developer sudo xcodebuild -
license
```

brew:

Follow installation instructions at brew.sh. Homebrew, commonly known as brew, is a package manager that simplifies software installation on macOS. **Optional:** Check that brew is happy with your system configuration by running 'brew doctor'. Fix any problems that it identifies. Then, use brew to install more libraries, as shown below.

```
brew doctor
brew install asio assimp bison bullet cmake console_bridge cppcheck \ cunit eigen freetype
graphviz opencv openssl orocos - kdl pcre poco \ pyqt5 python qt@5 sip spdlog tinynxml tinynxml2
```

Environment Variables:

Environment variables play a pivotal role in system configurations and are crucial for the seamless operation of ROS 2.

```
# Add the openssl dir for DDS - Security
# if you are using BASH , then replace ' zshrc ' with ' bashrc '
echo " export OPENSSL_ROOT_DIR=$( brew -- prefix openssl)" >> ~/. zshrc

# Add the Qt directory to the PATH and CMAKE_PREFIX_PATH
export CMAKE_PREFIX_PATH = $CMAKE_PREFIX_PATH : $( brew -- prefix qt@5 ) export PATH =
$PATH : $( brew -- prefix qt@5 )/ bin
```

Use 'python3 -m pip' to install the following libraries:

```
python 3 -m pip install -U \
  argcomplete catkin_pkg colcon - common - extensions coverage \ cryptography empy flake8
flake8 - blind - except == 0 .1 .1 flake8 - builtins \ flake8 - class - newline flake8 - comprehensions
flake8 - deprecated \
  flake8 - docstrings flake8 - import - order flake8 - quotes \
  importlib - metadata lark == 1 .1 .1 lxml matplotlib mock mypy == 0 .931 netifaces \ nose pep8 psutil
pydocstyle pydot pygraphviz pyparsing == 2 .4 .7 \
  pytest - mock rosdep rosdistro setuptools == 59 .6 .0 vcstool
```

Disable System Integrity Protection (SIP)

macOS/OS X versions >=10.11 have System Integrity Protection enabled by default. To deactivate it, follow these instructions. Disabling SIP allows specific software to run without the stringent macOS security checks.

Get the ROS 2 Code

Create a workspace and clone all repos:

```
mkdir -p ~/ ros2 _humble / src cd
~/ ros2 _humble
vcs import -- input https :// raw . githubusercontent . com / ros2 / ros2 / humble / ros2 . repos src
```

Build the ROS 2 Code

Run the 'colcon' tool to build everything: 'colcon' is a command-line tool used to build sets of software packages.

```
cd ~/ ros2 _humble /
colcon build -- symlink - install -- packages - skip - by - dep python_qt_binding
```

Note: due to an unresolved issue with SIP, Qt@5, and PyQt5, we need to turn off python qt binding to have the build succeed. This will be removed when the issue is resolved; see: [GitHub Issue](#).

Environment Setup

Source the ROS 2 setup file. This step ensures the terminal recognizes ROS 2 commands.

```
. ~/ros2_humble / install/ setup . zsh
```

Try Some Examples

In one terminal, set up the ROS 2 environment and then run a C++ talker:

```
ros2 run demo_nodes_cpp talker
```

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

```
ros2 run demo_nodes_py listener
```

Stay Up to Date

See Maintain source checkout to refresh your source installation periodically.

Uninstallation

If you installed your workspace with ‘colcon’, “uninstalling” could be just opening a new terminal and not sourcing the workspace’s setup file.

To free up space, you can delete the entire workspace directory:

```
rm -rf ~/ros2_humble
```

Install Gazebo Ignition Fortress (Source)

Note to Islam: We have a blocker here. In brief, ROS 2 supports macOS Mojave (10.14), and Gazebo Ignition Fortress supports macOS Catalina 10.15 and macOS BigSur 10.16. This means there is no common version on which both are supported. My suggestion to solve this conflict is to use Docker. However, preparing docker images/files is beyond the requirements of this tutorial. I want to discuss with you what to do.

5 Exploring Cyclone DDS as an RMW Implementation (optional)

While ROS 2 defaults to Fast DDS for its underlying middleware, its flexibility allows users to switch the middleware (RMW) at runtime. We highly recommend Cyclone DDS among the available options due to its performance, robustness, and active open-source development within the Eclipse IoT project. You can delve deeper into the intricacies of Cyclone DDS by visiting [this link](#). For a comprehensive guide on working with various ROS 2 middleware implementations, the ROS 2 documentation serves as an invaluable resource.

5.1 Installing Eclipse Cyclone DDS

Installation from apt repository

The most straightforward method of installation is via the ROS 2 apt repository:

```
sudo apt install ros - humble - rmw - cyclonedds - cpp
```

Switching to Cyclone DDS

To shift from your current RMW to Cyclone DDS, specify the appropriate environment variable:

```
export RMW_IMPLEMENTATION = rmw_cyclonedds_cpp
```

To permanently switch to Cyclone DDS as your default RMW every time you open a new terminal session, specify the appropriate environment variable in your `~/.bashrc`:

```
echo " export RMW_IMPLEMENTATION = rmw_cyclonedds_cpp " >> ~/.bashrc
```

This ensures that Cyclone DDS is always set as the default RMW implementation when you start a new terminal session.

Testing Eclipse Cyclone DDS

Once you have Eclipse Cyclone DDS set up, you can test it using the talker and listener nodes:

```
ros2 run demo_nodes_cpp talker ros2 run  
demo_nodes_cpp listener
```

6 Conclusion

You should now have a detailed guide on installing and running ROS 2 Humble and Gazebo Ignition Fortress on multiple operating systems. Happy exploring!

7 Additional Resources

- [ROS 2 Humble Official Documentation](#).
- [Gazebo Ignition Fortress Official Documentation](#).
- [Ubuntu 22.04 LTS Official Documentation](#)