

RACECAR Neo PC Software Installation Guide

MIT BWSI Autonomous RACECAR Course - Updated January 31st 2024

Welcome to MIT BWSI RACECAR! This updated tutorial will guide you through the process of setting up the RACECAR simulation software from the very beginning. Whether you are a new programmer or a seasoned professional, this guide will cover all the main points, troubleshooting guides, and things to look out for when installing the software.

racecar-neo-installer.....	1
Legacy Information.....	2
1) Visual Studio Code.....	3
2) Python 3.9.7.....	8
2.1) Install Ubuntu (Windows Only).....	12
2.2) Install X Server (Windows Only).....	12
3) Run Installer Setup Script.....	15
4) Test RACECAR Setup.....	18
*) Troubleshooting Guide.....	24
Problems with WSL: Windows 11.....	24
Error: 0x800701bc WSL 2 requires an update to its kernel component.....	25
Ubuntu account creation error.....	26
Ubuntu: The system cannot find the path specified.....	26

racecar-neo-installer

Template repository for native RACECAR installation (Entry Point)

For Windows 10/11, macOS 10.15+ (Catalina+) [>2019], and Linux (tested distros include Ubuntu 20+ & Debian 10+)

RACECAR will NOT work on Chromebook laptops!!!

racecar-neo-installer is a flexible and efficient way of installing the RACECAR Neo software onto your computer. The repository only contains a scripts folder inside a template racecar-student folder. Upon cloning the folder into your computer and running the setup.py script, the compiler will ask you to input your computer's operating system, RACECAR IP address, and curriculum type.



Then, it will run through a series of python library dependency installations, bash alias scripts, and git clone commands to set up the Racecar Neo installation to your unique specifications.

Since the installation process slightly varies between operating systems and types of RACECAR curriculum, the installer strives to simplify the setup process (making it less painful to go through), and make the course more accessible to those with less technical experience.

Legacy Information

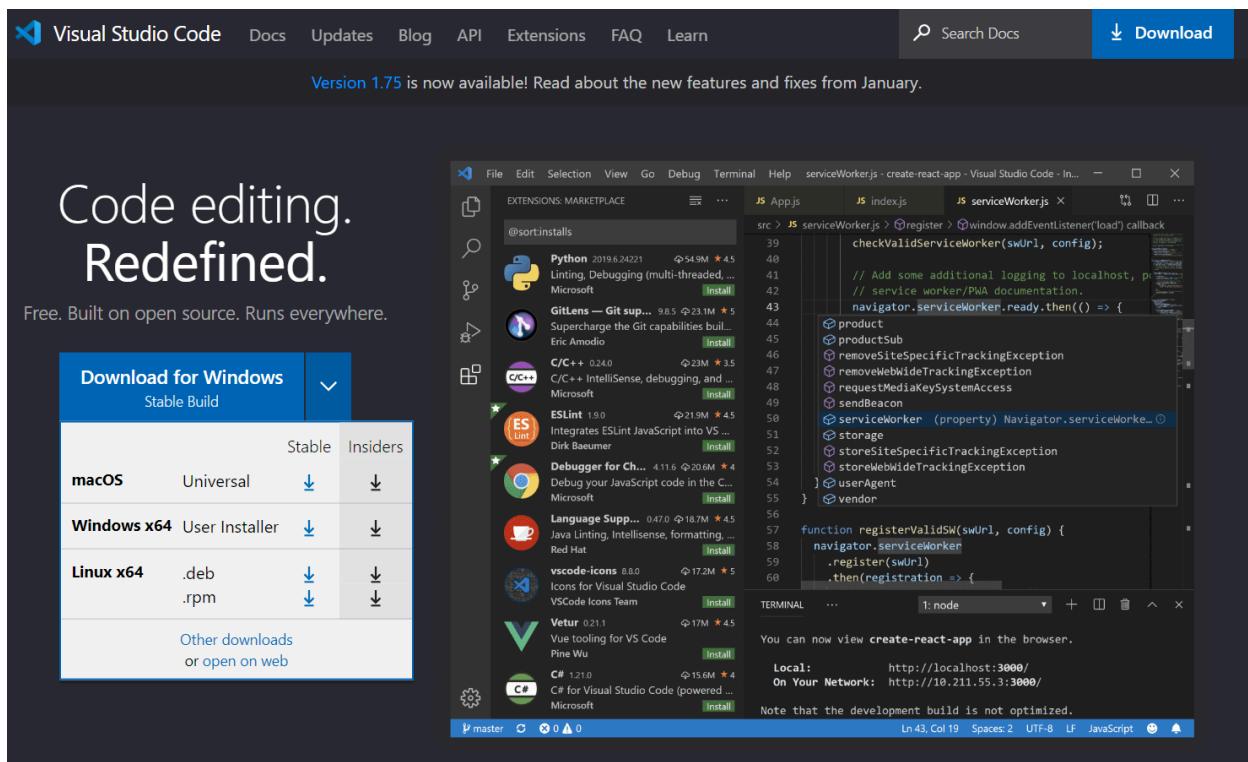
RACECAR Neo is based on RACECAR-MN (Model Neo), the course version supported between the years of 2019-2022. **RACECAR-MN development is no longer ongoing** and is deprecated. More on RACECAR-MN and the outdated installation process can be found here: [BWSI RACECAR-MN PC Software Installation Guide \[Part 1/2\]](#)

If you have previously installed RACECAR through the RACECAR-MN setup guide, we highly recommend you to upgrade to the new installer.

1) Visual Studio Code

1. Visit the official Visual Studio Code (VSC) website: <https://code.visualstudio.com/>.

Depending on which operating system your computer runs on, the primary download link will change. Download the **stable** version of VSC for your computer.

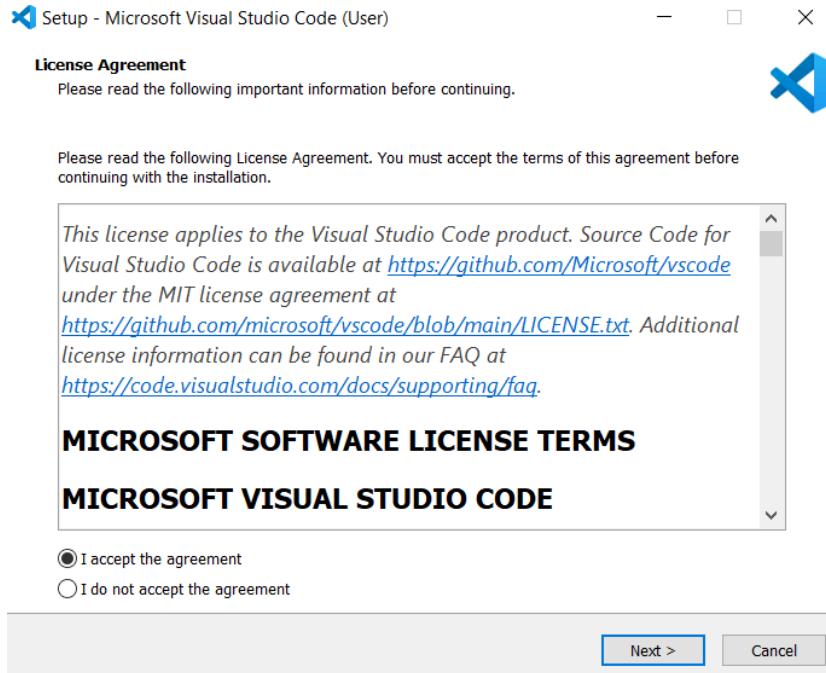


Thanks for downloading VS Code for Windows!

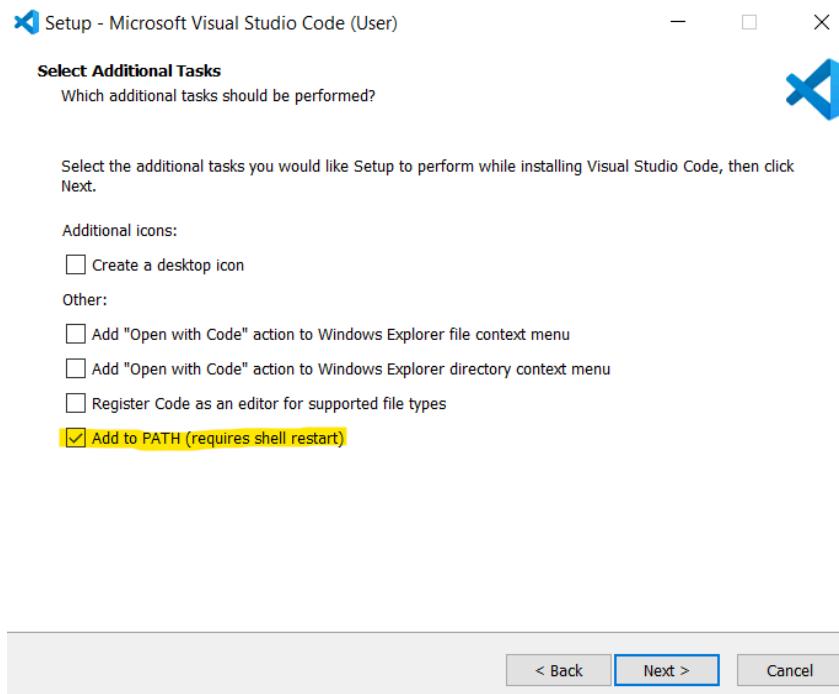
Download not starting? Try this [direct download link](#).

Please take a few seconds and help us improve ... [click to take survey](#).

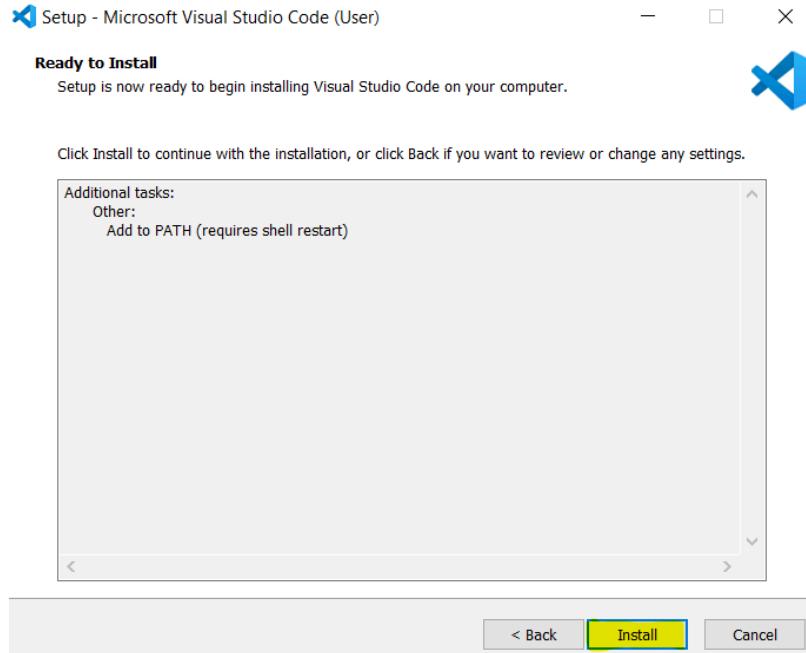
2. Accept the license agreement on page 1.



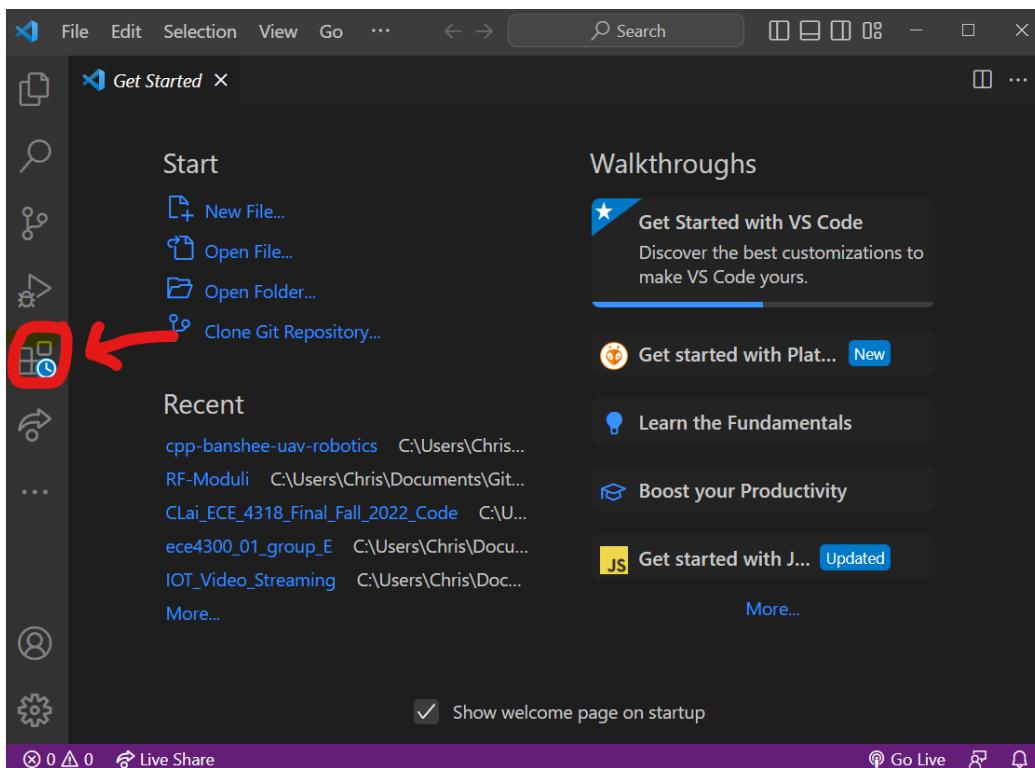
3. On the next page titled **additional tasks**, make sure that the **add to PATH (requires shell restart)** checkbox is checked. (WINDOWS ONLY)



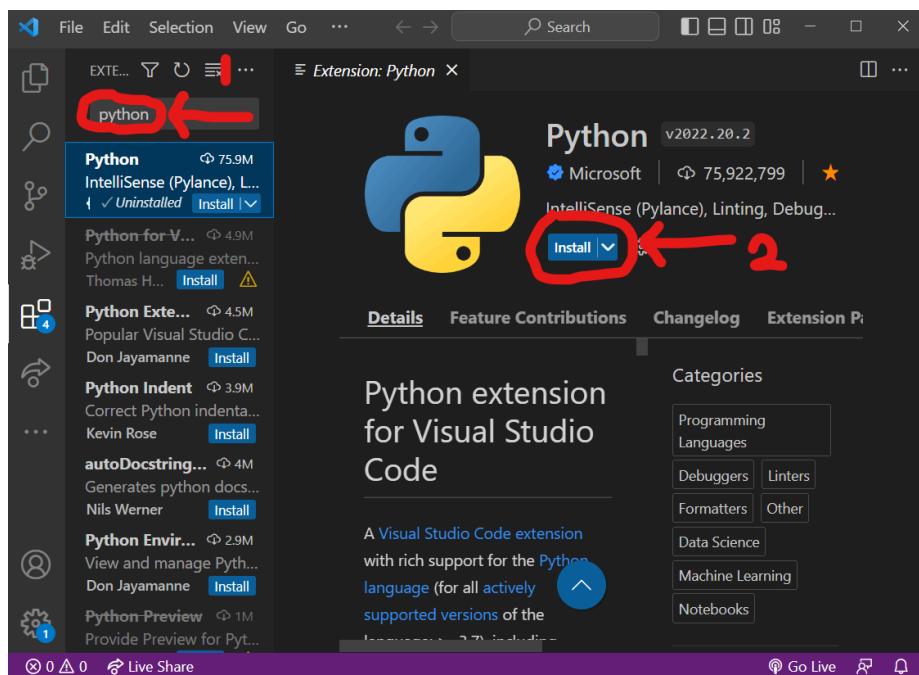
4. Complete the installation by pressing the **install** button.



5. Open Visual Studio Code and navigate to the “**Extensions**” tab. This button looks like a 2x2 array of squares with the top right square taken out.



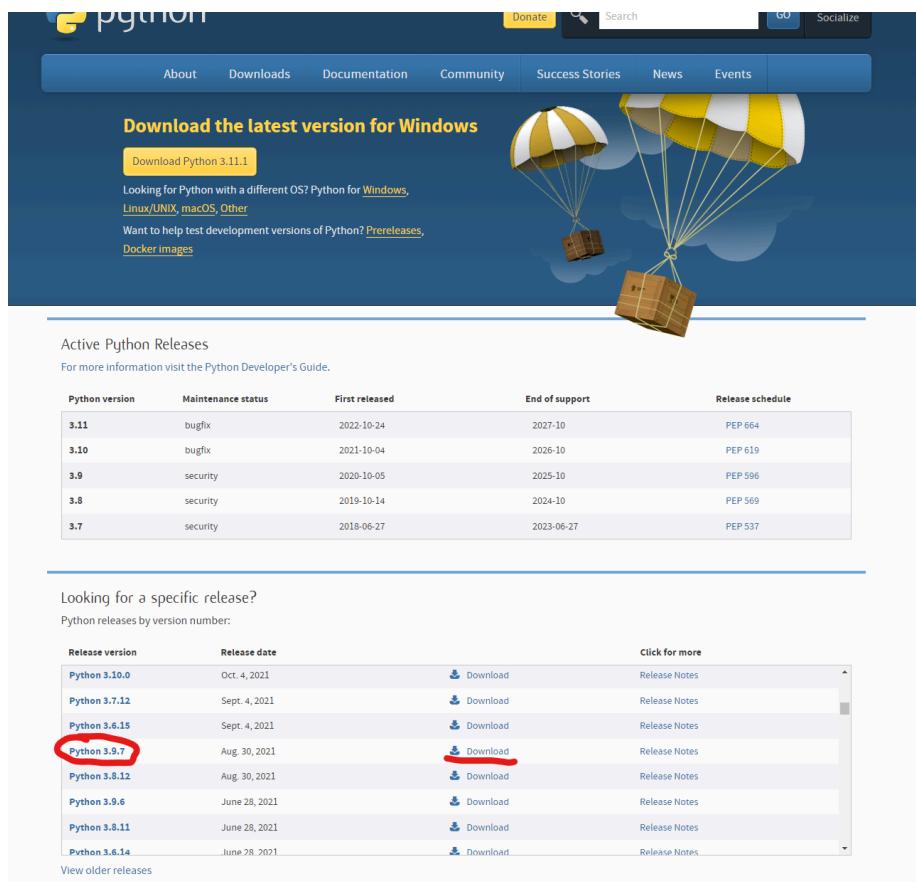
6. Type in “**python**” into the search bar and download the **python extension by Microsoft**. This will provide your compiler with IntelliSense (Pylance) capabilities. **Press the install button** to install this software into your computer.
 - a. [!] A common misconception is thinking that the Python coding language has been installed onto your computer after this task has been completed. But this is wrong. The Python extension allows VSC to compile Python code, as well as provide autocomplete, autocorrect, and error-checking features. The Python extension DOES NOT install the Python coding language onto the computer. We must complete this task separately.



2) Python 3.9.7

In this section, we will be installing the Python programming language and compiler into your computer. Keep in mind that Python and Visual Studio Code are different entities! VSC is able to work with Python, but VSC is not Python!

1. Go to the following website: <https://www.python.org/downloads/> and scroll down until you see the download link for **Python 3.9.7**. Install the executable or application file into your computer.
 - a. [!] This tutorial utilizes Python 3.9.7 as a demonstration of a stable version of Python in 2023, which is also compatible with BWSI RACECAR. This tutorial may be outdated in the future. But, as a note of safety, do not use the latest version of Python, as it may have bugs and unrefined parts that make it difficult to troubleshoot later down the line.



The screenshot shows the Python.org Downloads page. At the top, there's a navigation bar with links for About, Downloads, Documentation, Community, Success Stories, News, and Events. Below the navigation bar, a large banner features the text "Download the latest version for Windows" and a graphic of two boxes descending from the sky on yellow and white striped parachutes. A yellow button labeled "Download Python 3.11.1" is visible. Below the banner, there's a section titled "Active Python Releases" with a note about the Python Developer's Guide. A table lists active Python releases:

Python version	Maintenance status	First released	End of support	Release schedule
3.11	bugfix	2022-10-24	2027-10	PEP 664
3.10	bugfix	2021-10-04	2026-10	PEP 619
3.9	security	2020-10-05	2025-10	PEP 596
3.8	security	2019-10-14	2024-10	PEP 569
3.7	security	2018-06-27	2023-06-27	PEP 537

Below this table, there's a section titled "Looking for a specific release?" with a note about Python releases by version number. A table lists specific releases:

Release version	Release date	Click for more
Python 3.10.0	Oct. 4, 2021	Download Release Notes
Python 3.7.12	Sept. 4, 2021	Download Release Notes
Python 3.6.15	Sept. 4, 2021	Download Release Notes
Python 3.6.7	Aug. 30, 2021	Download Release Notes
Python 3.6.12	Aug. 30, 2021	Download Release Notes
Python 3.6.6	June 28, 2021	Download Release Notes
Python 3.6.11	June 28, 2021	Download Release Notes
Python 3.6.14	June 28, 2021	Download Release Notes

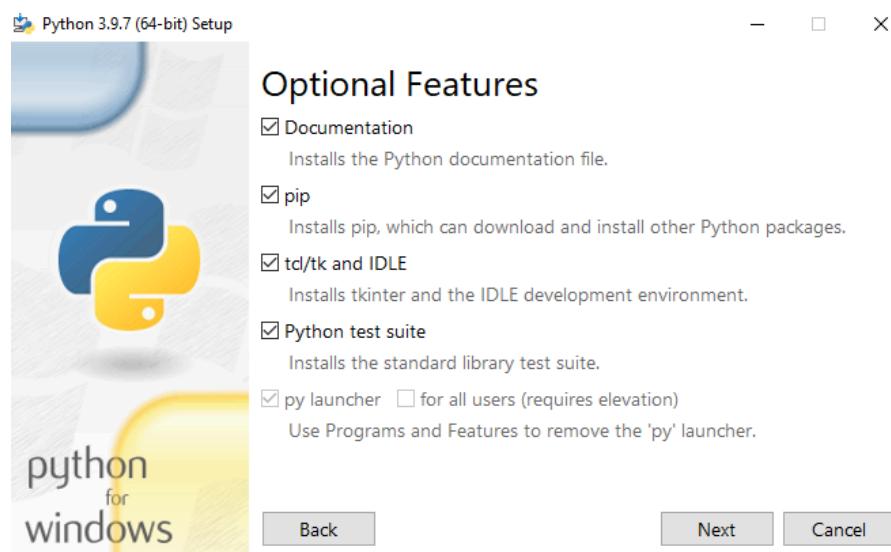
A red circle highlights the "Python 3.6.7" row in the table.

2. Depending on your operating system, you may be installing a different package. For Windows, [check to see](#) if you are operating on a 32-bit or 64-bit operating system

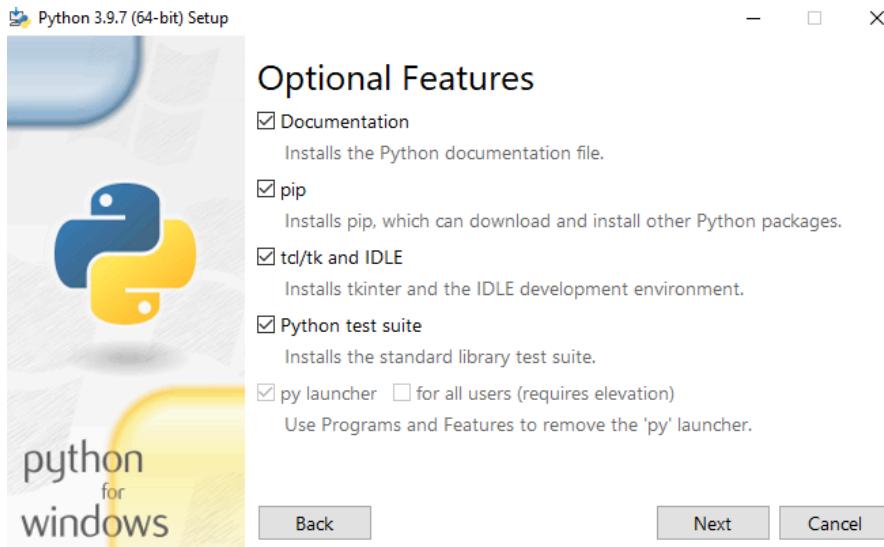
and download the package accordingly. I recommend downloading the “Installer” for Linux, Windows, or Mac.

Version	Operating System	Description	MD5 Sum	File Size	GPG
Gzipped source tarball	Source release		5f463f30b1fdcb545f156583630318b3	25755357	SIG
XZ compressed source tarball	Source release		fdedb060b483bc01850a3f412eea1d954	19123232	SIG
macOS 64-bit Intel installer	macOS	for macOS 10.9 and later	c68c2f885f26b09536857610644260d4	30038206	SIG
macOS 64-bit universal2 installer	macOS	for macOS 10.9 and later, including macOS 11 Big Sur on Apple Silicon (experimental)	825067610b16b03ec814630df1b65193	38144099	SIG
Windows embedable package (32-bit)	Windows		6d12e3e0f942830de8466a83d30a45fb	7652688	SIG
Windows embedable package (64-bit)	Windows		67e19ff32b3ef62a40bccd50e33b0f53	8473919	SIG
Windows help file	Windows		b92a78506ccf258d5ad0d98c341fc5d1	9263789	SIG
Windows installer (32-bit)	Windows		0d949bdfdbd0c8c66107a980a95efd85	27811736	SIG
Windows installer (64-bit)	Windows	Recommended	cc3eabc1f9d6c703d1d2a4e7c041bc1d	28895456	SIG

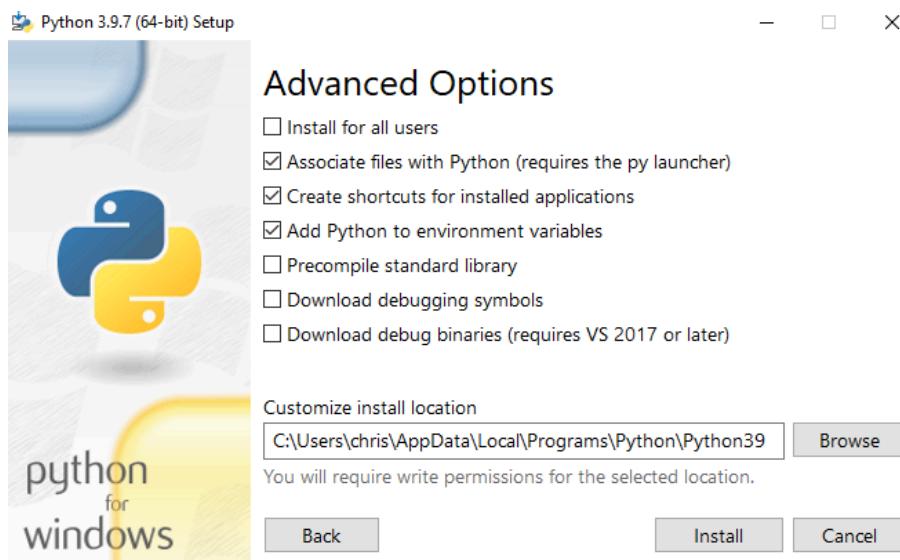
3. Open the executable or application file that you have downloaded.
 - a. **IMPORTANT: Check the box labeled “Add Python 3.9 to PATH”** (This box is normally unchecked. If left unchecked, this can lead to further issues down the line when running code from the command prompt, or downloading dependencies.)



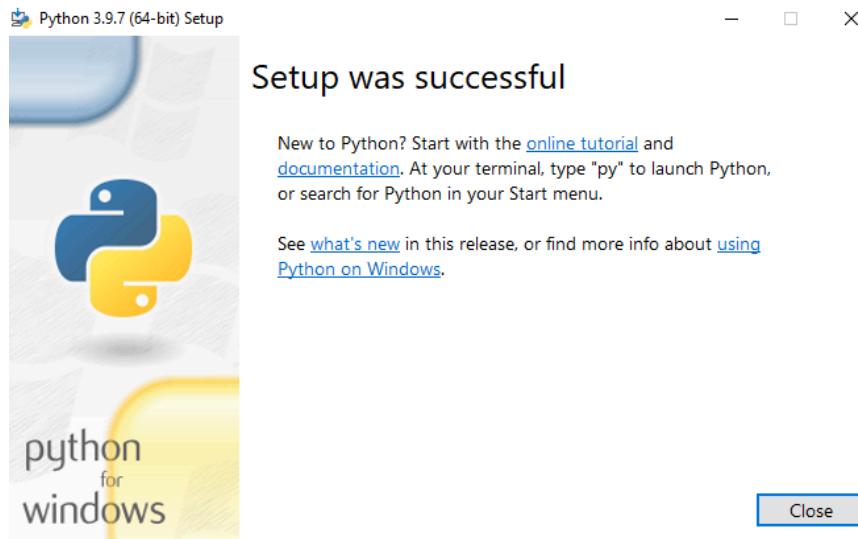
- Click on “**Customize Installation**”. Verify that “pip” and “tk” are checked. Click on “Next” afterwards to advance with the installation



- Nothing important needs to be done in the “Advanced Options” screen, unless your computer has specialized user profiles or you would like Python to be installed in a special directory.
 - Verify** that “Add Python to environment variables” is checked.



- Click “**Install**”. When you see the “Setup was Successful” screen, you are finished!



7. To test that your installation is successful, open up a Command Prompt window and type in "python3" or "python".
 - a. **Tip:** Note that my version states 3.9.13 since that is the version of Python that has been bound to that command. Your version should be "3.9.7".
 - b. **[!]** RACECAR is compatible with Python versions 3.10 or lower. Any version above 3.10 may not be stable with RACECAR. If your default version of Python is different, please use a virtual environment when setting up RACECAR.
(venv) -> [<https://docs.python.org/3/library/venv.html>]

```
Command Prompt - python3
Microsoft Windows [Version 10.0.19044.2486]
(c) Microsoft Corporation. All rights reserved.

C:\Users\chris>python3
Python 3.9.13 (tags/v3.9.13:6de2ca5, May 17 2022, 16:36:42) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

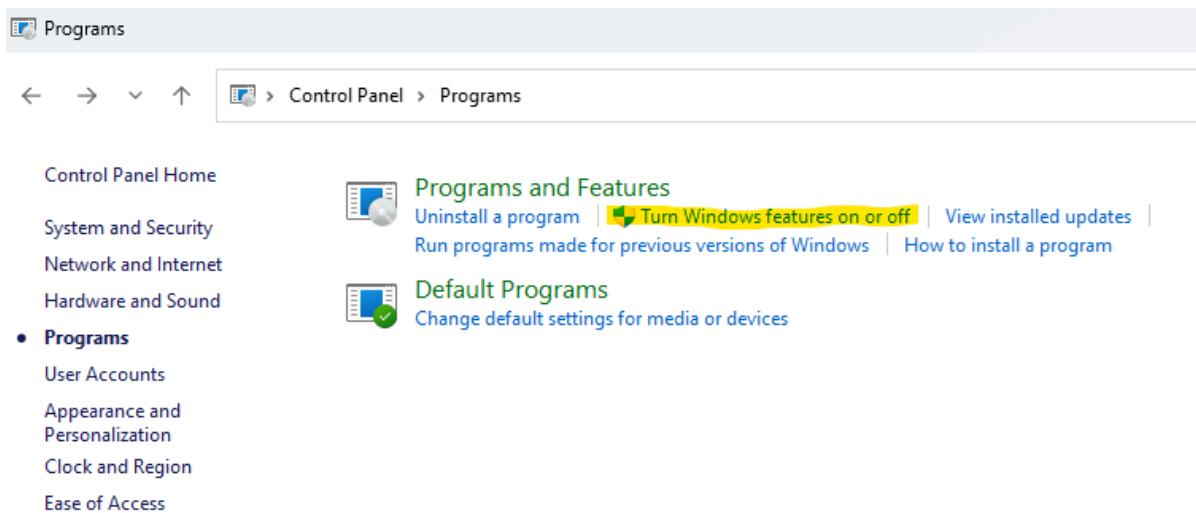
2.1) Install Ubuntu [Windows Only]

If you are on a Mac or Linux machine, skip this step and move on to the next section.

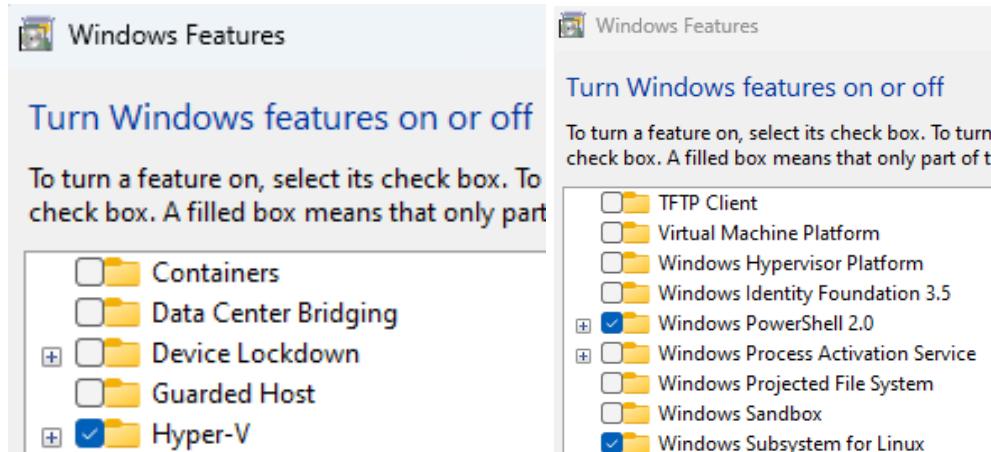
Every operating system has a terminal (sometimes referred to as a shell or console), which is a text-based program used to run low-level commands. The terminals on Mac (**Terminal**) and Linux (**bash, zsh, etc.**) are compatible with the RACECAR-MN, but the terminals on Windows (**cmd or PowerShell**) unfortunately are not.

To program the RACECAR-MN on a Windows machine, you will need to install Ubuntu on Windows. This will allow us to use bash, a Linux terminal program.

1. Open Control Panel and type “Windows Features” in the search bar. Select “**Turn Windows features on or off**”.



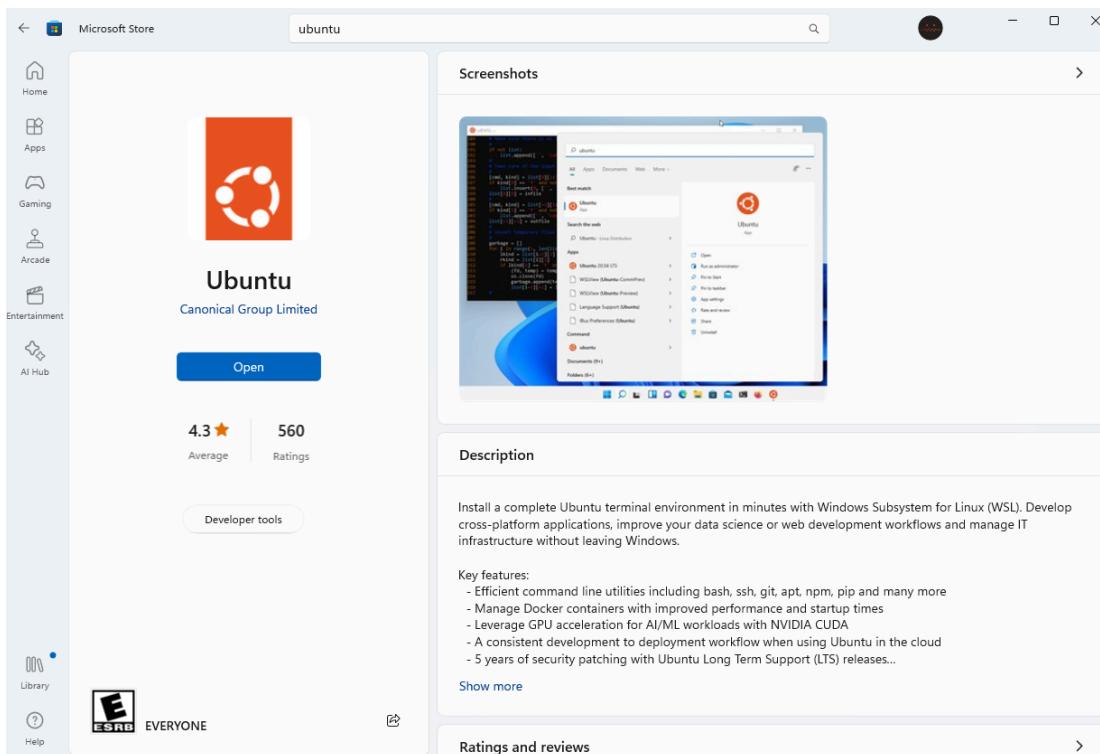
2. In the window titled “Windows Features”, scroll through the list and select the boxes as follows:
 - a. **Hyper-V (if it exists)**
 - b. **Windows Subsystem for Linux**



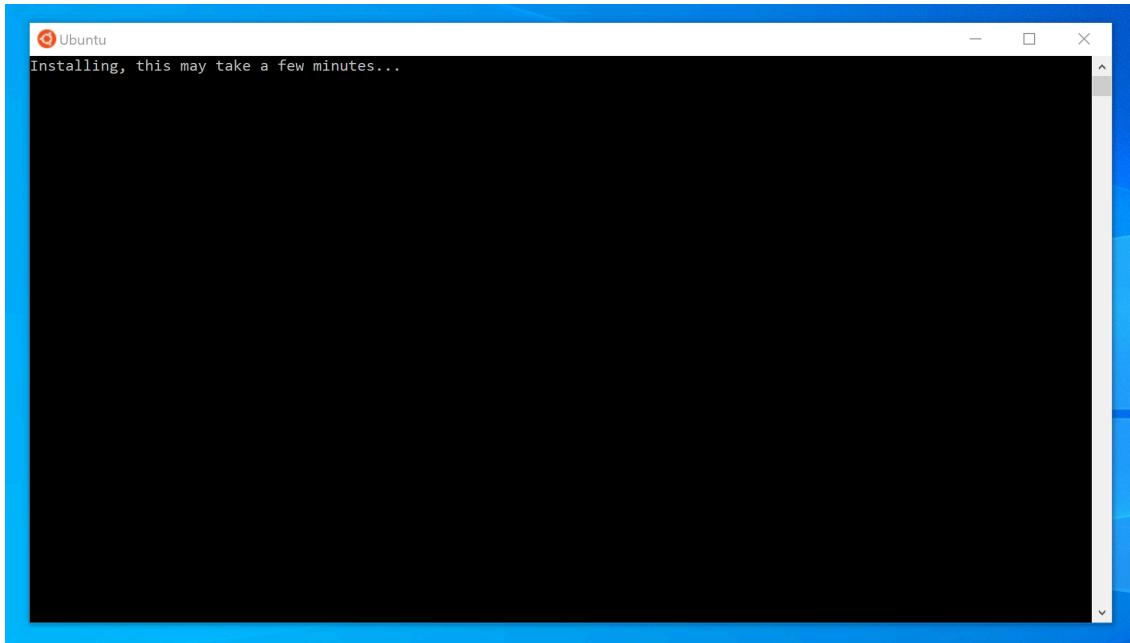
3. Click "OK". You will be prompted to restart your computer to install these changes.

Restart your computer now and come back to this guide for the next steps.

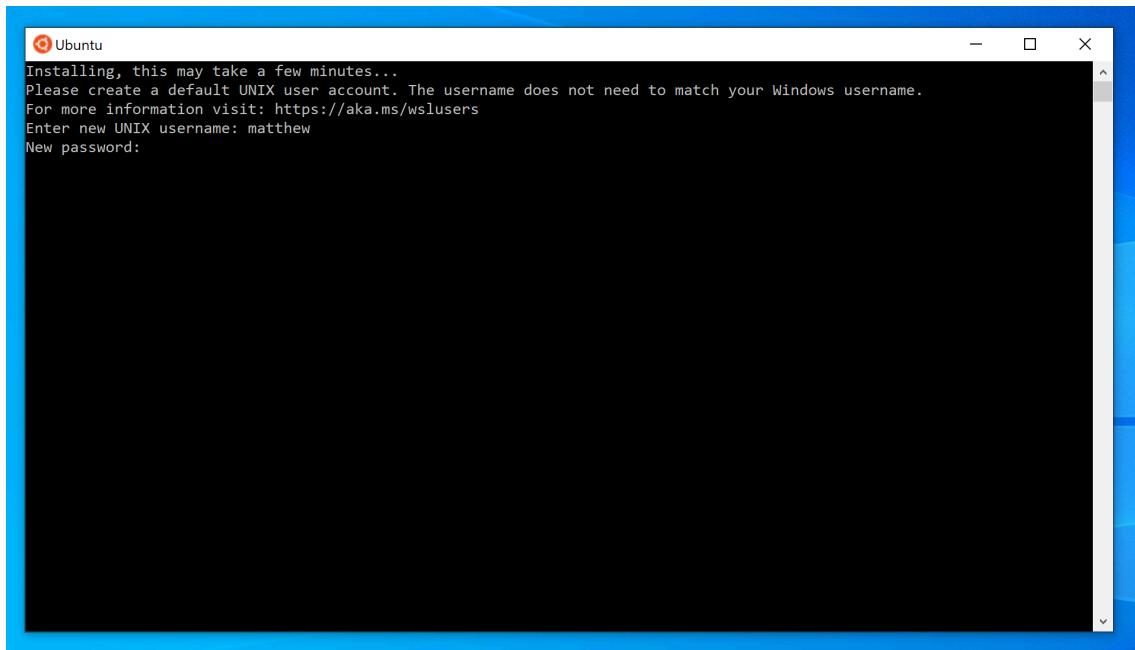
4. Install **Ubuntu** from the **Windows Store**.

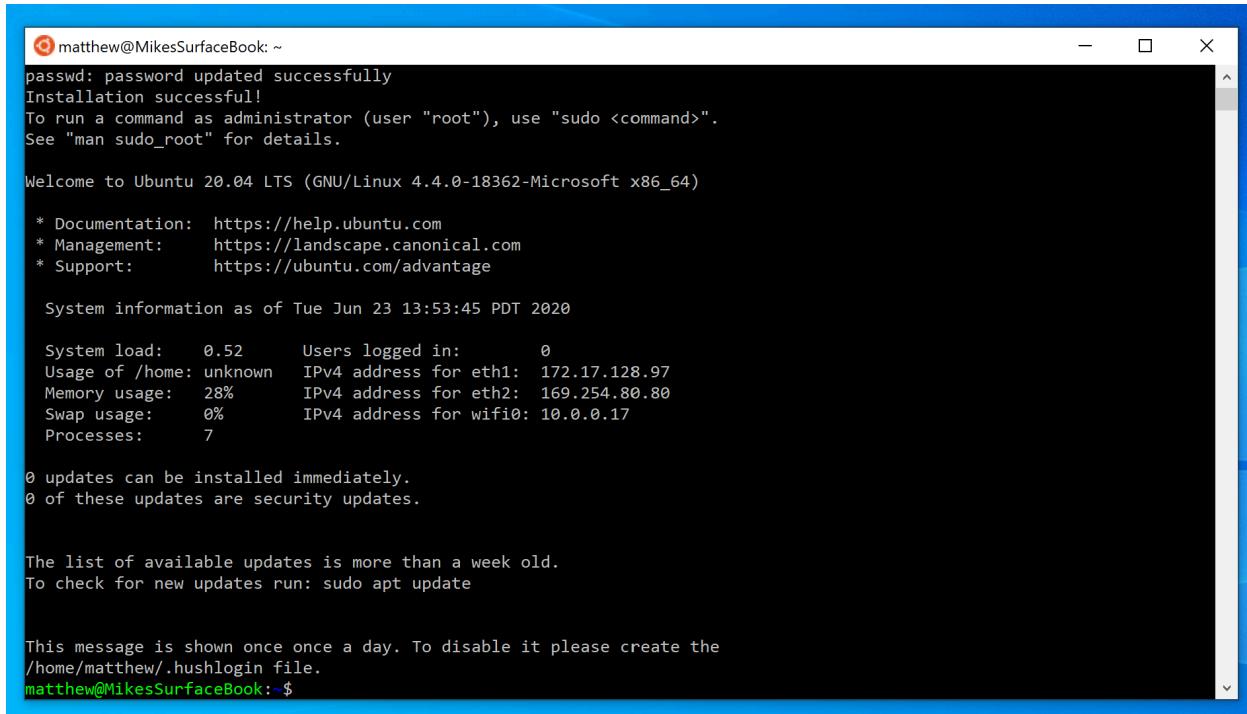


5. Launch Ubuntu from the start menu, which will open a bash terminal. The first time you open bash, it will need to install for a few minutes.



6. Finally, you will be asked to create a username and password. This will be the password that you use whenever bash asks you for your **sudo** password.
 - a. **[!] Your username should not contain spaces.**
 - b. **[!] In Linux, passwords are not shown when typed for security. It may appear that you are typing nothing on the screen, but it is simply hidden in the background.**





```
matthew@MikesSurfaceBook: ~
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 20.04 LTS (GNU/Linux 4.4.0-18362-Microsoft x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

 System information as of Tue Jun 23 13:53:45 PDT 2020

System load:  0.52      Users logged in:      0
Usage of /home: unknown  IPv4 address for eth1:  172.17.128.97
Memory usage: 28%       IPv4 address for eth2:  169.254.80.80
Swap usage:   0%       IPv4 address for wifi0: 10.0.0.17
Processes:     7

0 updates can be installed immediately.
0 of these updates are security updates.

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

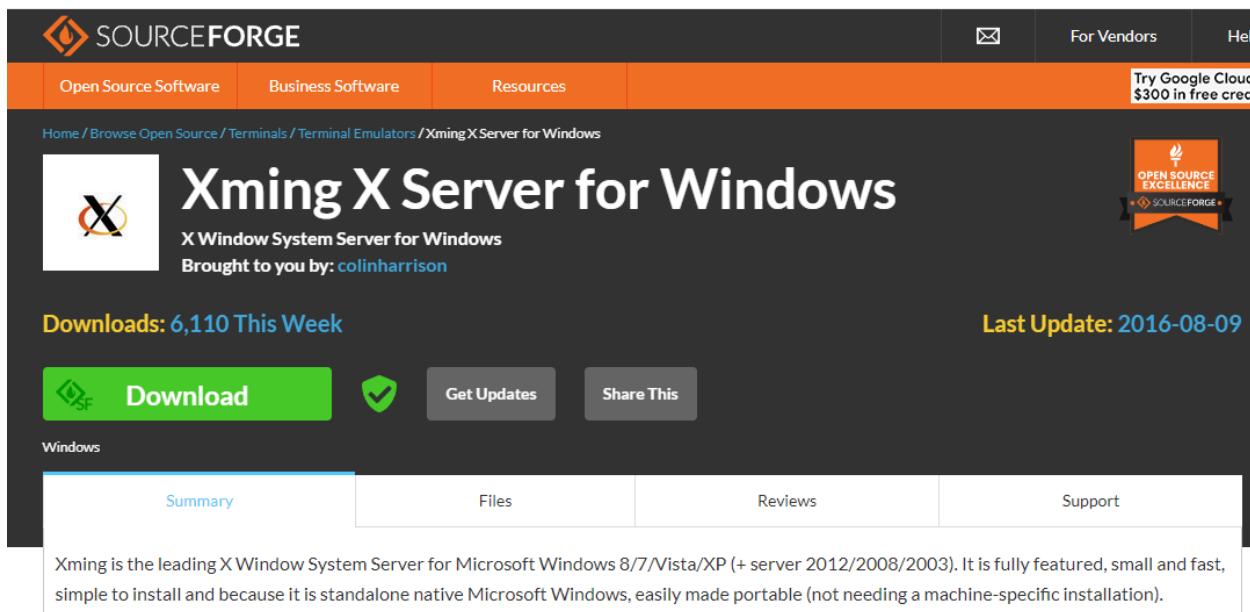
This message is shown once once a day. To disable it please create the
/home/matthew/.hushlogin file.
matthew@MikesSurfaceBook:~$
```

2.2) Install X Server [Windows Only]

If you are on a Mac or Linux machine, skip this step and move on to the next section.

1. Download the Xming .exe installer from this link:

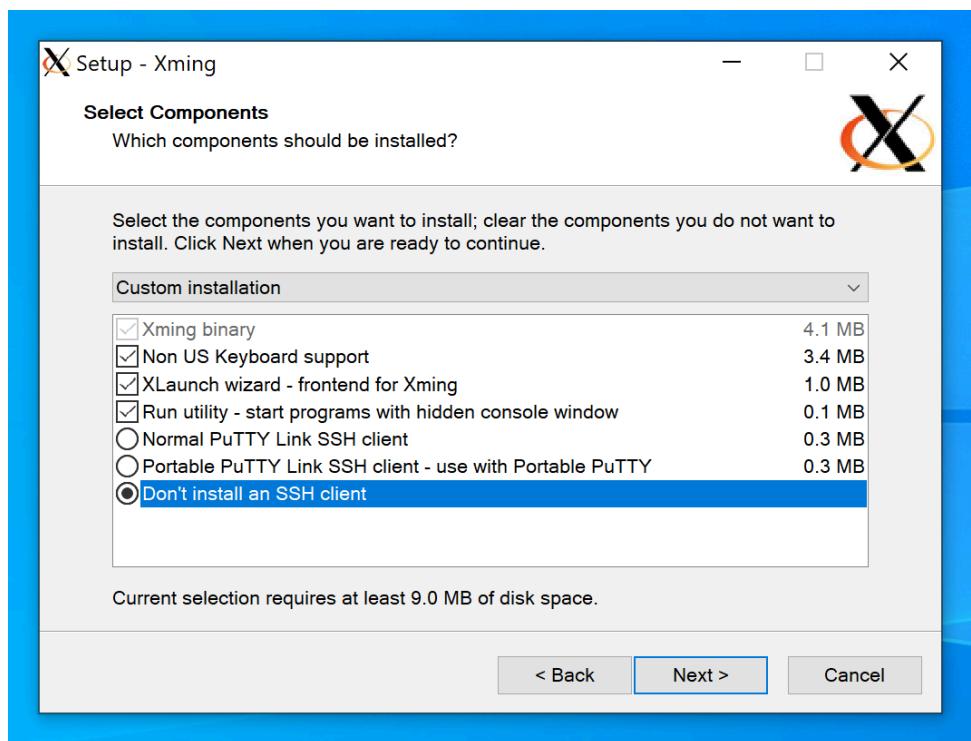
<https://sourceforge.net/projects/xming/>



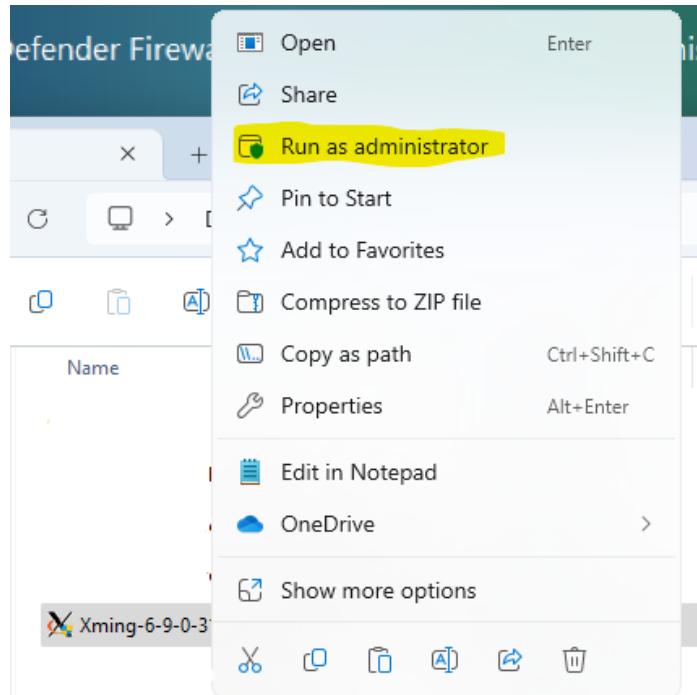
2. Launch the Xming installer once it finishes downloading. This will open a new window as shown below. Use the default settings until you reach the "**Select Components**" page.



3. On the “**Select Components**” page, select “**Don’t install an SSH client**”. Make sure that the box for “**XLaunch wizard - frontend for Xming**” is checked.

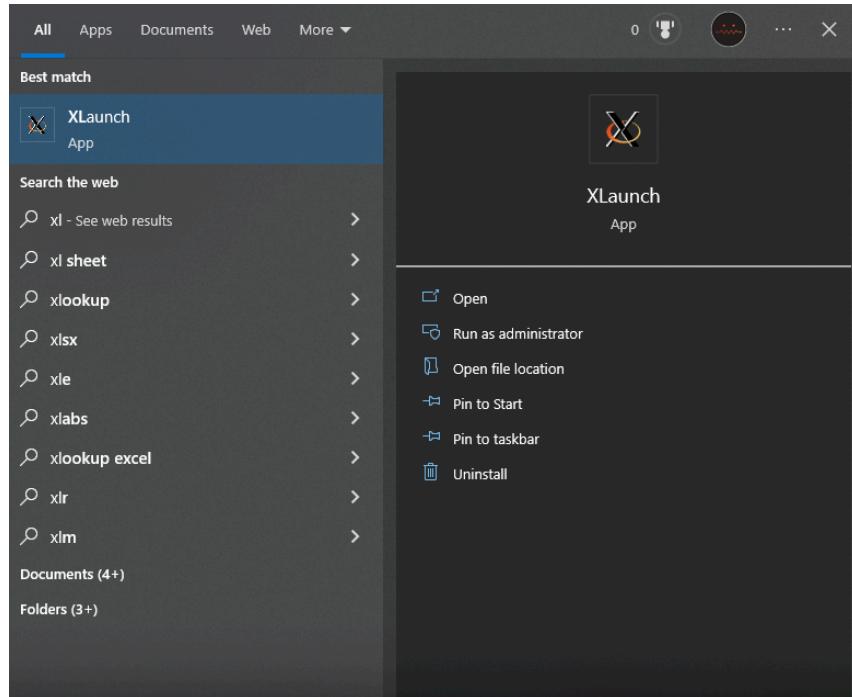


4. If you get a failure message after finishing this process, you may have to run the .exe installer as an administrator. **Right-click** on the .exe file and select “**Run as administrator**”. Input your password and let it run again.

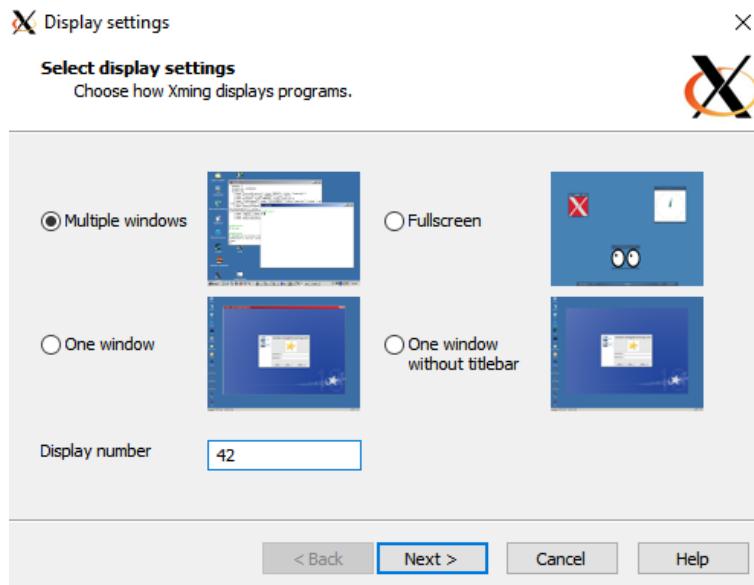


[!] Before running RACECAR Sim on a Windows machine, sometimes X Launch will not run automatically, providing you with a terminal error. To fix this issue, complete the following steps (Complete the following steps if this is your first time downloading X Launch as well):

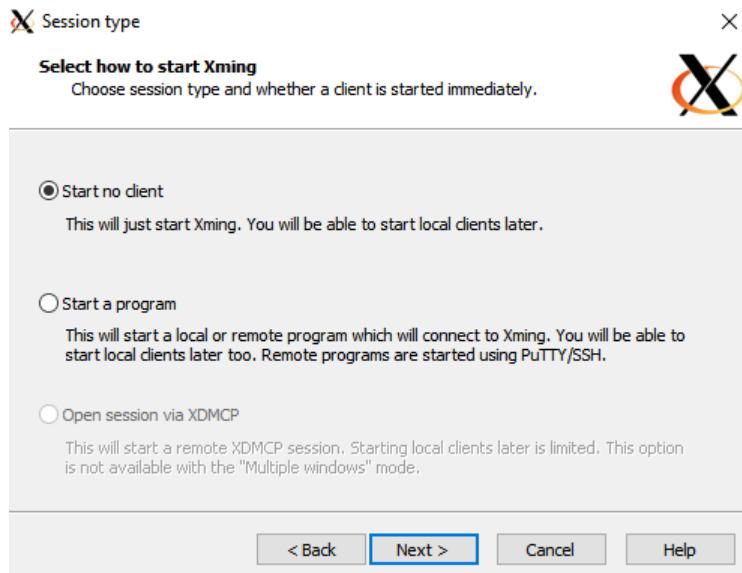
1. Run XLaunch on your computer by finding the application:



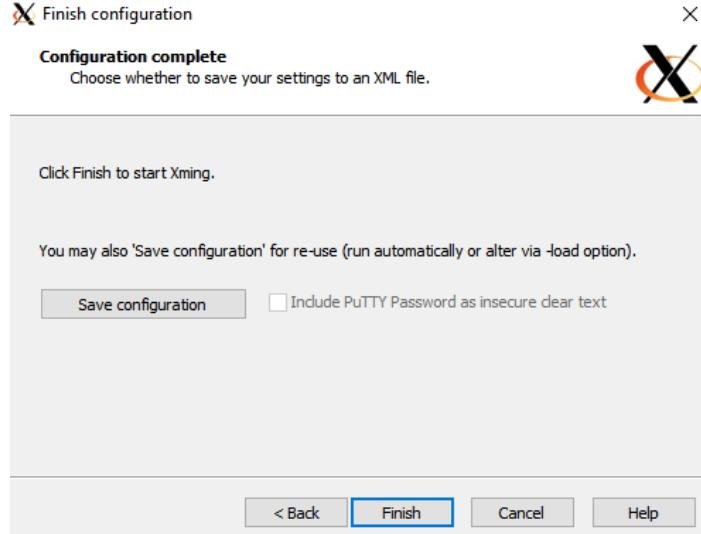
2. On the “**Display Settings**” window, make sure “**multiple windows**” are selected, and the display number is set to “**42**”. Click on “**Next**”.



3. On the “**Session Type**” window, make sure “**No Client**” is selected and click “**Next**”.



4. Continue to click "**Next**" until the "**Finish Configuration**" window is reached. Click on "**Finish**" to start Xming. No window should pop up after this. Xming will run in the background the next time you complete the assignment that requires it.



3) Run Installer Setup Script

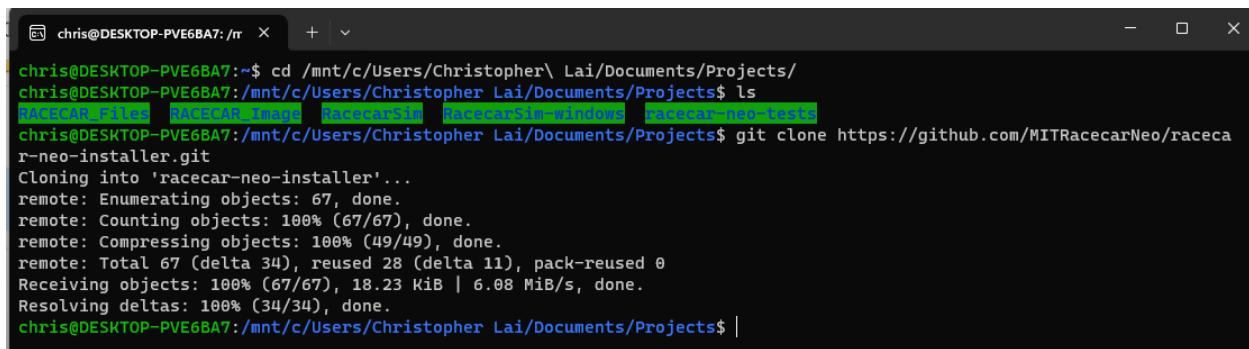
1. In a WSL Ubuntu window (Windows), or terminal window (Mac/Linux), navigate to a folder that you want to store your RACECAR projects in using the **cd** command.
 - a. If you are using WSL Ubuntu on Windows, run **cd ~** to create a folder in the root directory. Then, you can run **explorer.exe .** to open a File Explorer window in that directory.
 - i. **[!]** There is an intentional period at the end of the **explorer.exe .** command.



```
chris@DESKTOP-PVE6BA7:~$ cd ~
chris@DESKTOP-PVE6BA7:~$ |
```

2. Type in the following git clone command (or copy and past) and hit “**Enter**”:

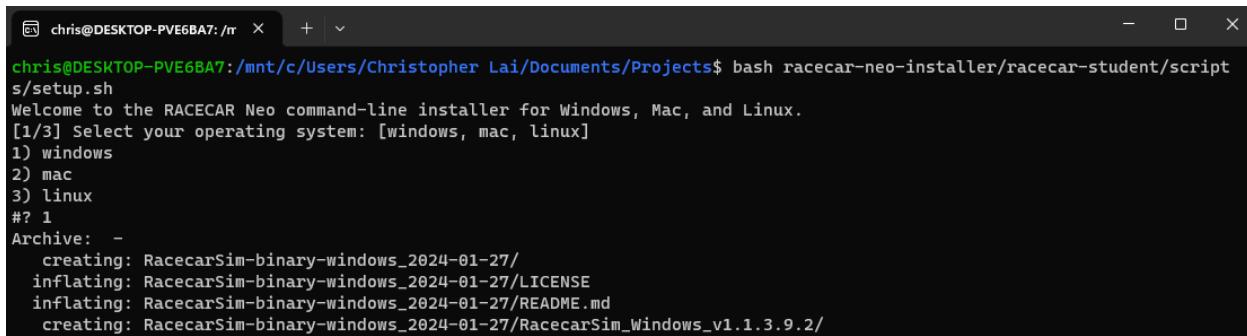
git clone https://github.com/MITRacecarNeo/racecar-neo-installer.git



```
chris@DESKTOP-PVE6BA7:~$ cd /mnt/c/Users/Christopher\ Lai/Documents/Projects/
chris@DESKTOP-PVE6BA7:/mnt/c/Users/Christopher Lai/Documents/Projects$ ls
RACECAR_files RACECAR_Image RacecarSim RacecarSim-windows Racecar-neo-tests
chris@DESKTOP-PVE6BA7:/mnt/c/Users/Christopher Lai/Documents/Projects$ git clone https://github.com/MITRacecarNeo/racecar-neo-installer.git
Cloning into 'racecar-neo-installer'...
remote: Enumerating objects: 67, done.
remote: Counting objects: 100% (67/67), done.
remote: Compressing objects: 100% (49/49), done.
remote: Total 67 (delta 34), reused 28 (delta 11), pack-reused 0
Receiving objects: 100% (67/67), 18.23 KiB | 6.08 MiB/s, done.
Resolving deltas: 100% (34/34), done.
chris@DESKTOP-PVE6BA7:/mnt/c/Users/Christopher Lai/Documents/Projects$ |
```

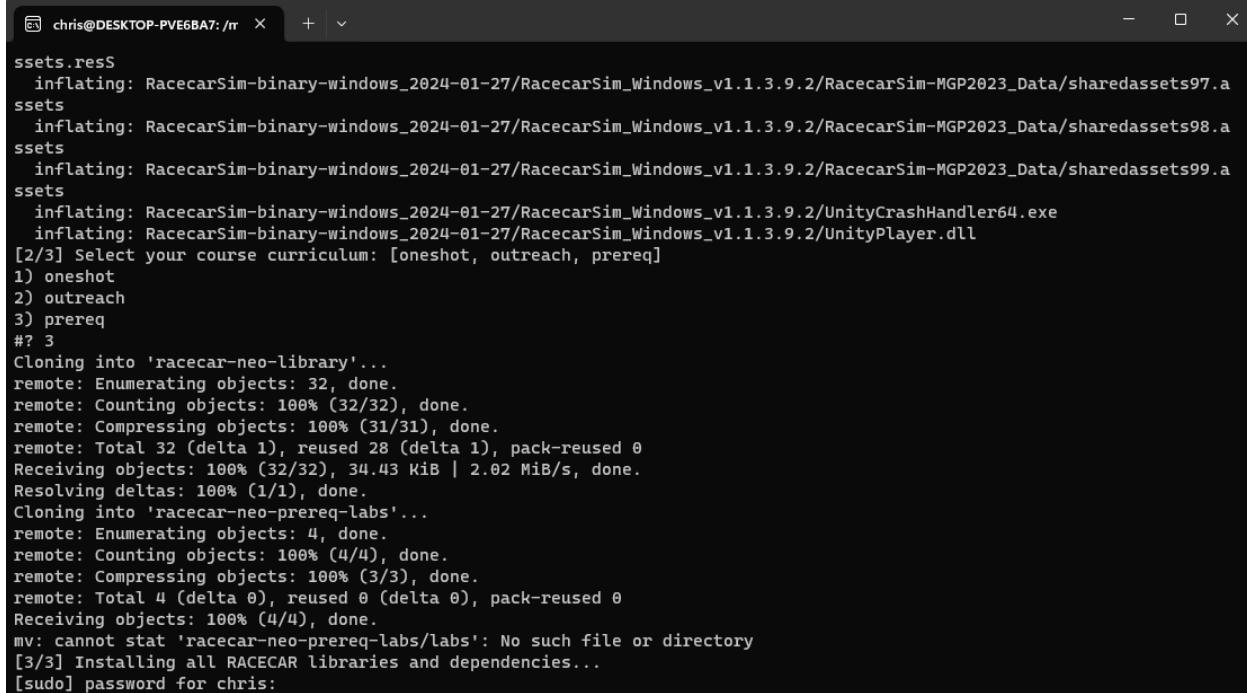
3. Run the following command to activate the setup script.

bash racecar-neo-installer/racecar-student/scripts/setup.sh



```
chris@DESKTOP-PVE6BA7:/mnt/c/Users/Christopher Lai/Documents/Projects$ bash racecar-neo-installer/racecar-student/scripts/setup.sh
Welcome to the RACECAR Neo command-line installer for Windows, Mac, and Linux.
[1/3] Select your operating system: [windows, mac, linux]
1) windows
2) mac
3) linux
#? 1
Archive: -
  creating: RacecarSim-binary-windows_2024-01-27/
  inflating: RacecarSim-binary-windows_2024-01-27/LICENSE
  inflating: RacecarSim-binary-windows_2024-01-27/README.md
  creating: RacecarSim-binary-windows_2024-01-27/RacecarSim_Windows_v1.1.3.9.2/
```

4. The file will ask you several questions, such as operating system used, and absolute path to RACECAR folder. (Type in your sudo password when prompted)
 - i. **Operating System:** (1: Windows, 2: Mac, 3: Linux)
 - ii. **Curriculum Type:** (1: oneshot, 2: outreach, 3: prereq)



```
ssets.resS
  inflating: RacecarSim-binary-windows_2024-01-27/RacecarSim_Windows_v1.1.3.9.2/RacecarSim-MGP2023_Data/sharedassets97.a
ssets
  inflating: RacecarSim-binary-windows_2024-01-27/RacecarSim_Windows_v1.1.3.9.2/RacecarSim-MGP2023_Data/sharedassets98.a
ssets
  inflating: RacecarSim-binary-windows_2024-01-27/RacecarSim_Windows_v1.1.3.9.2/RacecarSim-MGP2023_Data/sharedassets99.a
ssets
  inflating: RacecarSim-binary-windows_2024-01-27/RacecarSim_Windows_v1.1.3.9.2/UnityCrashHandler64.exe
  inflating: RacecarSim-binary-windows_2024-01-27/RacecarSim_Windows_v1.1.3.9.2/UnityPlayer.dll
[2/3] Select your course curriculum: [oneshot, outreach, prereq]
1) oneshot
2) outreach
3) prereq
#? 3
Cloning into 'racecar-neo-library'...
remote: Enumerating objects: 32, done.
remote: Counting objects: 100% (32/32), done.
remote: Compressing objects: 100% (31/31), done.
remote: Total 32 (delta 1), reused 28 (delta 1), pack-reused 0
Receiving objects: 100% (32/32), 34.43 KiB | 2.02 MiB/s, done.
Resolving deltas: 100% (1/1), done.
Cloning into 'racecar-neo-prereq-labs'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (4/4), done.
mv: cannot stat 'racecar-neo-prereq-labs/labs': No such file or directory
[3/3] Installing all RACECAR libraries and dependencies...
[sudo] password for chris:
```

5. Wait as the installer script completes all the library and dependency installations. This may take a while depending on your computer and internet speed. (~15 minutes)

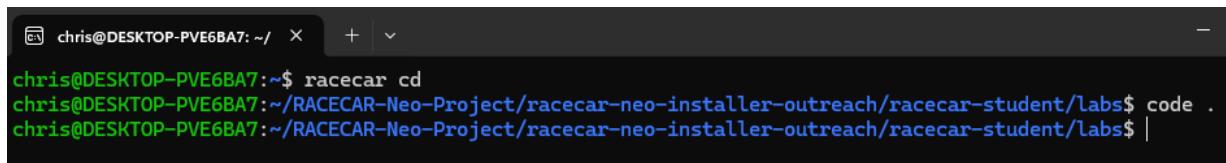
```
chris@DESKTOP-PVE6BA7:~/r   X  +  v
ssets.resS
  inflating: RacecarSim-binary-windows_2024-01-27/RacecarSim_Windows_v1.1.3.9.2/RacecarSim-MGP2023_Data/sharedassets97.a
sssets
  inflating: RacecarSim-binary-windows_2024-01-27/RacecarSim_Windows_v1.1.3.9.2/RacecarSim-MGP2023_Data/sharedassets98.a
sssets
  inflating: RacecarSim-binary-windows_2024-01-27/RacecarSim_Windows_v1.1.3.9.2/RacecarSim-MGP2023_Data/sharedassets99.a
sssets
  inflating: RacecarSim-binary-windows_2024-01-27/RacecarSim_Windows_v1.1.3.9.2/UnityCrashHandler64.exe
  inflating: RacecarSim-binary-windows_2024-01-27/RacecarSim_Windows_v1.1.3.9.2/UnityPlayer.dll
[2/3] Select your course curriculum: [oneshot, outreach, prereq]
1) oneshot
2) outreach
3) prereq
#? 3
Cloning into 'racecar-neo-library'...
remote: Enumerating objects: 32, done.
remote: Counting objects: 100% (32/32), done.
remote: Compressing objects: 100% (31/31), done.
remote: Total 32 (delta 1), reused 28 (delta 1), pack-reused 0
Receiving objects: 100% (32/32), 34.43 KiB | 2.02 MiB/s, done.
Resolving deltas: 100% (1/1), done.
Cloning into 'racecar-neo-prereq-labs'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (4/4), done.
mv: cannot stat 'racecar-neo-prereq-labs/labs': No such file or directory
[3/3] Installing all RACECAR libraries and dependencies...
[sudo] password for chris:
```

- a. After the script is finished, run the command **source ~/.bashrc** to activate the new scripts.
 - b. Alternatively, **close and re-open** your terminal window.
6. Run **racecar test** and verify the output looks correct.

```
chris@DESKTOP-PVE6BA7:~/r   X  +  v
chris@DESKTOP-PVE6BA7:~$ racecar cd
chris@DESKTOP-PVE6BA7:~/racecar-neo-installer/racecar-student/labs$ racecar test
racecar tool set up successfully!
  RACECAR_ABSOLUTE_PATH: /home/chris/racecar-neo-installer/racecar-student
  RACECAR_IP: 127.0.0.1
  RACECAR_TEAM: student
chris@DESKTOP-PVE6BA7:~/racecar-neo-installer/racecar-student/labs$ |
```

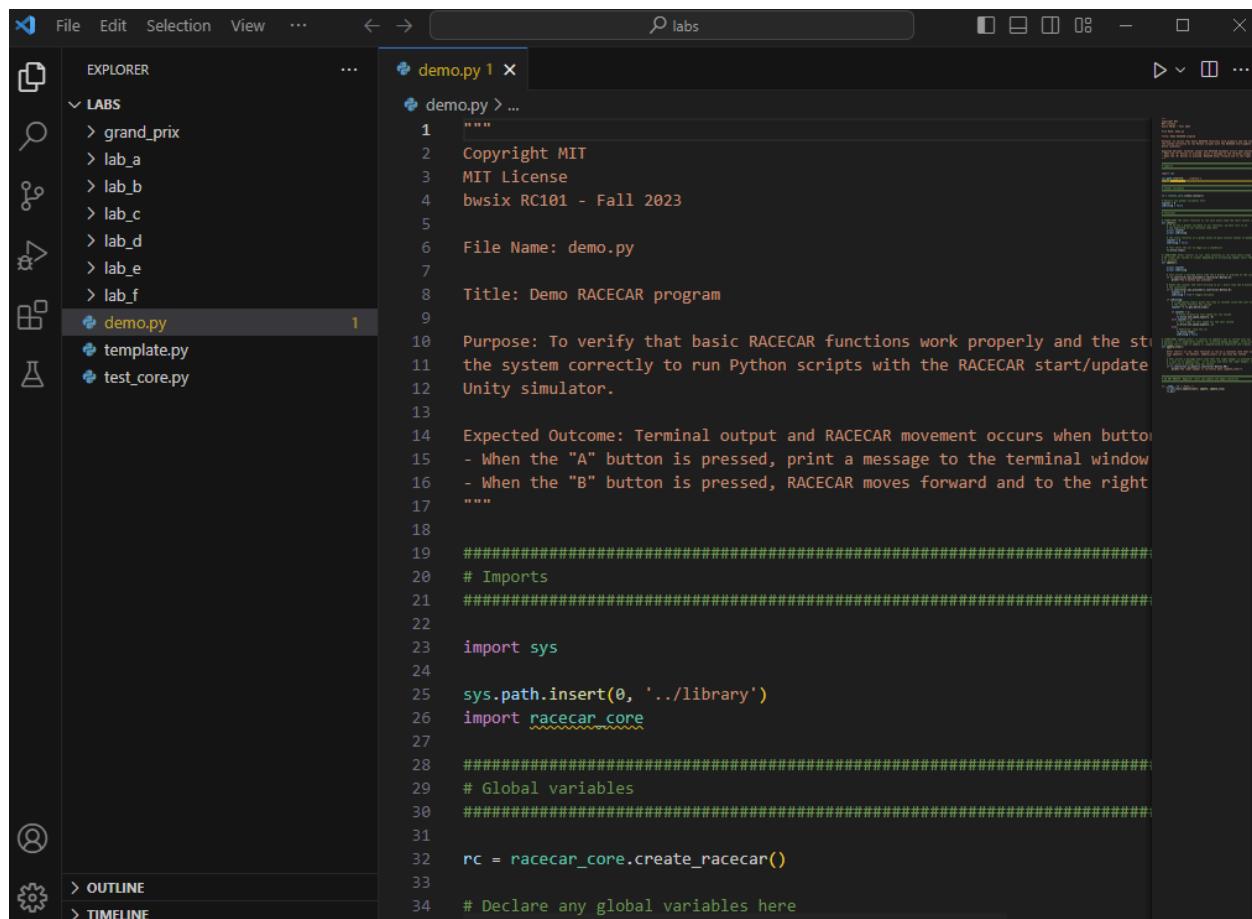
4) Test RACECAR Setup

1. Open a new terminal window and type in the following commands
 - a. **racecar-cd**: Navigates to the labs folder
 - b. **code .**: Opens lab files in Visual Studio Code as a workspace
 - i. **[!]** There is an intentional period in the command code .



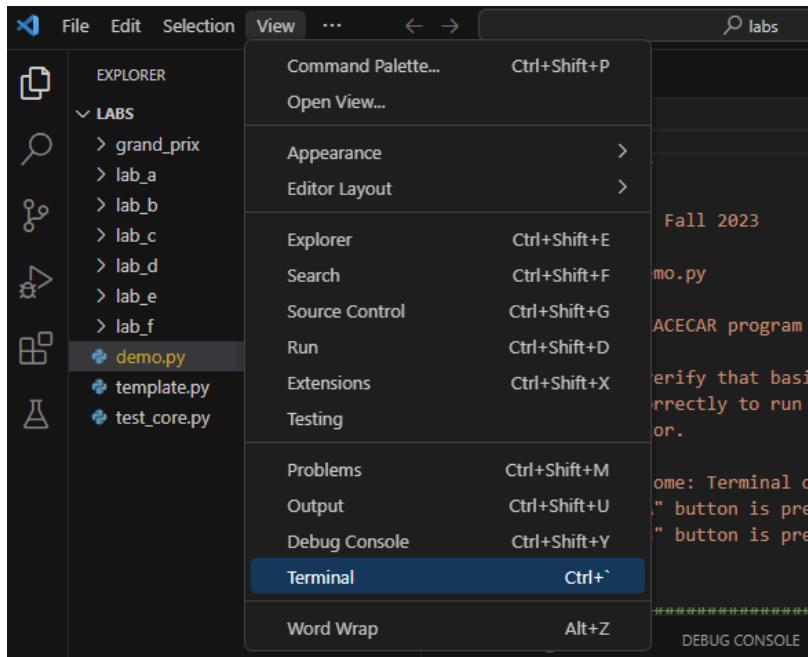
```
chris@DESKTOP-PVE6BA7:~/...$ racecar cd
chris@DESKTOP-PVE6BA7:~/RACECAR-Neo-Project/racecar-neo-installer-outreach/racecar-student/Labs$ code .
chris@DESKTOP-PVE6BA7:~/RACECAR-Neo-Project/racecar-neo-installer-outreach/racecar-student/Labs$ |
```

2. Navigate to the file "demo.py" and read through the commands as to what the expected behavior of the file is.



```
demo.py 1 x
demo.py > ...
1 """
2 Copyright MIT
3 MIT License
4 bwsix RC101 - Fall 2023
5
6 File Name: demo.py
7
8 Title: Demo RACECAR program
9
10 Purpose: To verify that basic RACECAR functions work properly and the system correctly to run Python scripts with the RACECAR start/update Unity simulator.
11
12 Expected Outcome: Terminal output and RACECAR movement occurs when button
13 - When the "A" button is pressed, print a message to the terminal window
14 - When the "B" button is pressed, RACECAR moves forward and to the right
15 """
16
17 #####
18 # Imports
19 #####
20
21
22 import sys
23
24
25 sys.path.insert(0, '../library')
26 import racecar_core
27
28 #####
29 # Global variables
30 #####
31
32 rc = racecar_core.create_racecar()
33
34 # Declare any global variables here
```

3. Navigate to **View > Terminal**, which opens up a terminal in Visual Studio Code.



4. In the terminal window, make sure that you are in the correct directory, then run the command "**racecar sim demo.py**".
- a. The directory should be a variation of:
".../racecar-neo-installer/racecar-student/labs"
 - b. **For Windows:** If you are on Windows (WSL Ubuntu), make sure that the VSC terminal is "bash" instead of "powershell", else this step will fail! Check the top right corner of the terminal window to see which type of terminal you are running your script in, and make sure it's the right one!
 - c. **For Mac:** Ensure that the terminal selection is "bash" or "zsh"



The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The top menu bar includes File, Edit, Selection, View, and a search bar labeled 'labs'. On the left, there's a sidebar with icons for Explorer, LABS, Outline, Timeline, and Help. The Explorer view shows a folder structure under 'LABS' containing 'grand_prix', 'lab_a', 'lab_b', 'lab_c', 'lab_d', 'lab_e', 'lab_f', and files 'demo.py', 'template.py', and 'test_core.py'. The 'demo.py' file is currently selected and open in the main editor area. The code in 'demo.py' is as follows:

```
1 """
2 Copyright MIT
3 MIT License
4 bwsix RC101 - Fall 2023
5
6 File Name: demo.py
7
8 Title: Demo RACERCAR program
9
10 Purpose: To verify that basic RACERCAR functions work properly and the system correctly to run Python scripts with the RACERCAR start/update Unity simulator.
11
12 Expected Outcome: Terminal output and RACERCAR movement occurs when button
13 - When the "A" button is pressed, print a message to the terminal window
14 - When the "B" button is pressed, RACERCAR moves forward and to the right
15 """
16
17 """
18
19 #####
20 #####
```

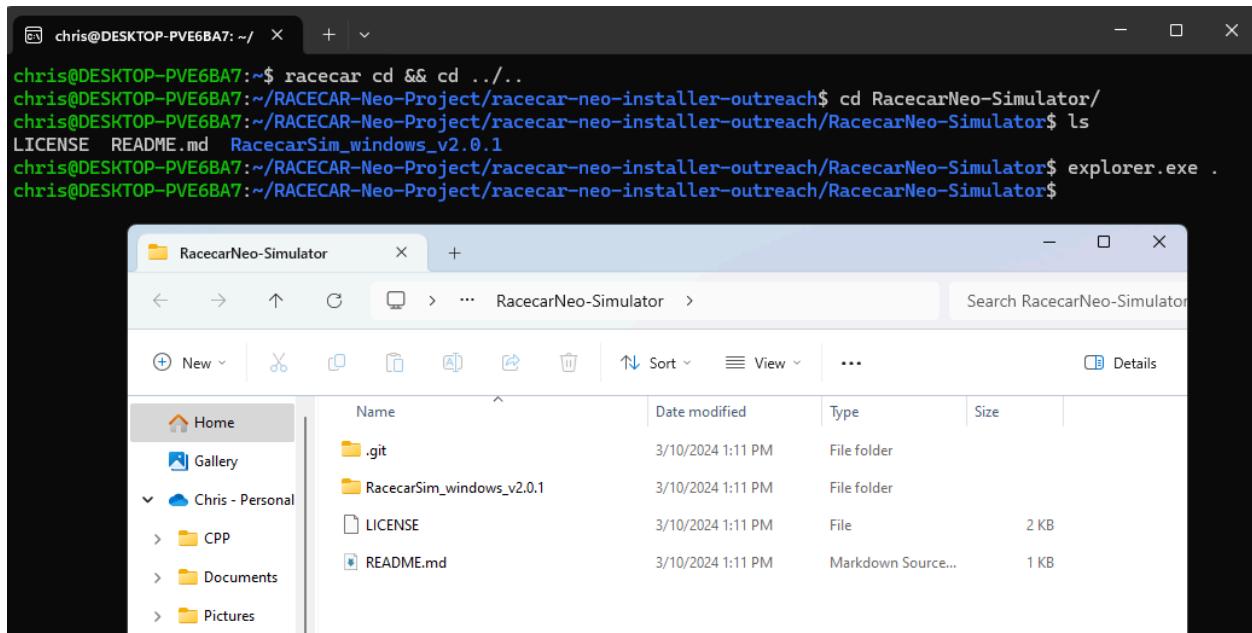
Below the editor, tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS are visible. The TERMINAL tab is active, showing a PowerShell prompt. A dropdown menu for the terminal shows options: PowerShell, Git Bash, Command Prompt, and Ubuntu (WSL), with 'Ubuntu (WSL)' currently selected. The status bar at the bottom indicates the file is 'Ln 1' and the version is '3.9.7 64-bit'.

5. Navigate to the RACECAR Sim application. The application is pulled into the **racecar-neo-installer** folder during the setup procedure and is located under the relative path: ".../racecar-neo-installer/RacecarNeo-Simulator". Type the following commands to navigate to the right folder.
 - a. **racecar cd && cd ..**
 - b. **cd RacecarNeo-Simulator**



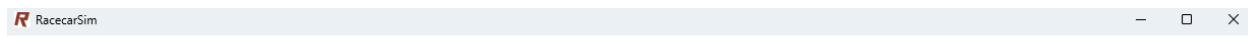
```
chrish@DESKTOP-PVE6BA7:~$ racecar cd && cd ../..
chrish@DESKTOP-PVE6BA7:~/RACECAR-Neo-Project/racecar-neo-installer-outreach$ cd RacecarNeo-Simulator/
chrish@DESKTOP-PVE6BA7:~/RACECAR-Neo-Project/racecar-neo-installer-outreach/RacecarNeo-Simulator$ ls
LICENSE README.md RacecarSim_windows_v2.0.1
chrish@DESKTOP-PVE6BA7:~/RACECAR-Neo-Project/racecar-neo-installer-outreach/RacecarNeo-Simulator$ |
```

6. Use the respective command based on your OS to open the folder in the file explorer. **Each command has an intentional period after it. Don't forget it!!!**
 - a. **Windows:** Type "**explorer.exe .**" and press "**Enter**".
 - b. **Mac:** Type "**open .**" and press "**Enter**".
 - c. **Linux:** Type "**nautilus .**" and press "**Enter**".

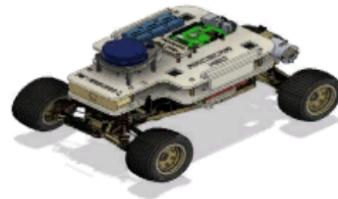


7. Navigate to the folder titled "**RacecarSim_[your OS version]_v####**". In the folder there should be one of the following files. Double click the file to open the simulator:
 - a. **Windows:** RacecarSim.exe

- b. **Mac:** RacecarSim.app
- c. **Linux:** RacecarSim.x86_64

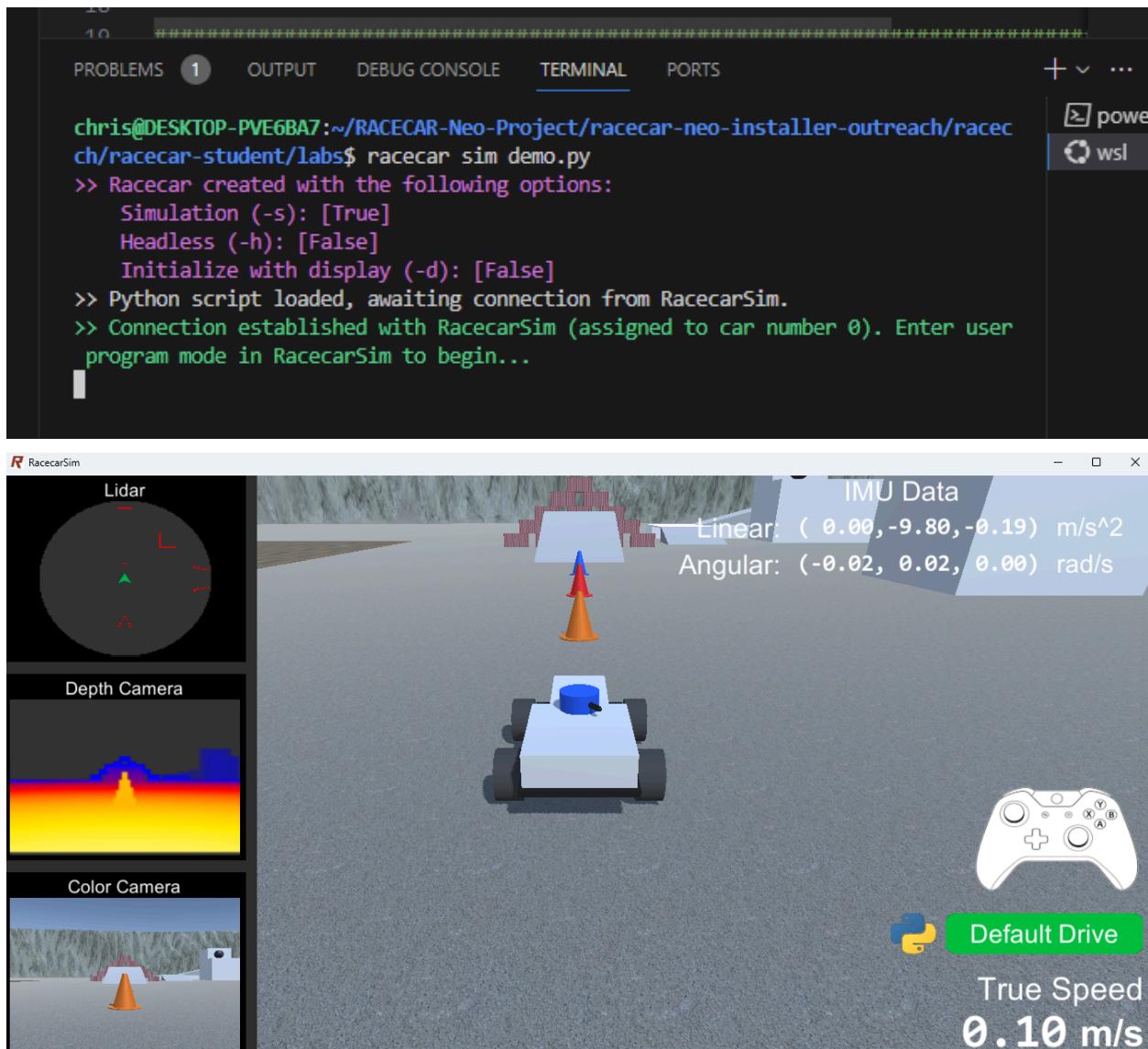


RACECAR Neo Simulation



8. For convenience, you may choose to pin this file to your desktop or taskbar for easy access. Next, click “**Begin Simulation**”. Observe the **solid Python logo** on the bottom right corner, which indicates a successful connection. In addition, the terminal should have successfully connected as well.
 - a. **[!!!]**If the Python logo is not lit up and you are running **Windows 11**, you must downgrade from WSL2 to WSL1 and restart this step. Click on this link to be redirected to the troubleshooting guide.

 [BWSI RACECAR Neo PC Software Installation Guide \[Part 1/2\]](#)



9. Press “**Enter**” while clicked inside of the simulation and verify that the following behavior occurs as expected:
 - a. When the “A” button is pressed (press the 1 key on the keyboard), the terminal states that “The A button was pressed”
 - b. When the “B” button is pressed (press the 2 key on the keyboard), the car drives forward and to the right.

```

18
19 #####
20
PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS + v ... ^
21
22 chris@DESKTOP-PVE6BA7:~/RACECAR-Neo-Project/racecar-neo-installer-outreach/racecar-student/labs$ racecar sim demo.py
23 >> Racecar created with the following options:
24   Simulation (-s): [True]
25   Headless (-h): [False]
26   Initialize with display (-d): [False]
27 >> Python script loaded, awaiting connection from RacecarSim.
28 >> Connection established with RacecarSim (assigned to car number 0). Enter user
29   program mode in RacecarSim to begin...
30 The A button was pressed

```



10. Press the “**ESC**” button to exit the simulation and “**Ctrl+C**” while clicked inside of the terminal to exit the script. You have now successfully set up the RACECAR simulator!

*) Troubleshooting Guide

Sometimes, things go wrong! Here are some of the most common issues we have found and the methods to fix them.

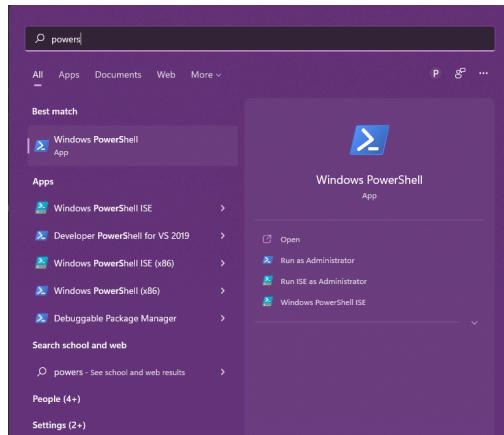
Problems with WSL: Windows 11

If the simulation is not connecting to the Python script through Ubuntu, it may be a Windows Subsystem Linux (WSL) issue. Since Windows 10, Microsoft has migrated systems to WSL 2 from WSL 1. Since RACECAR was developed before Windows 11, some computers running 11 may run into this version compatibility issue.

To fix this, we must downgrade from WSL2 back to WSL1.

[!] DO NOT DO THIS IN POWERSHELL RUNNING AS ADMINISTRATOR [!]

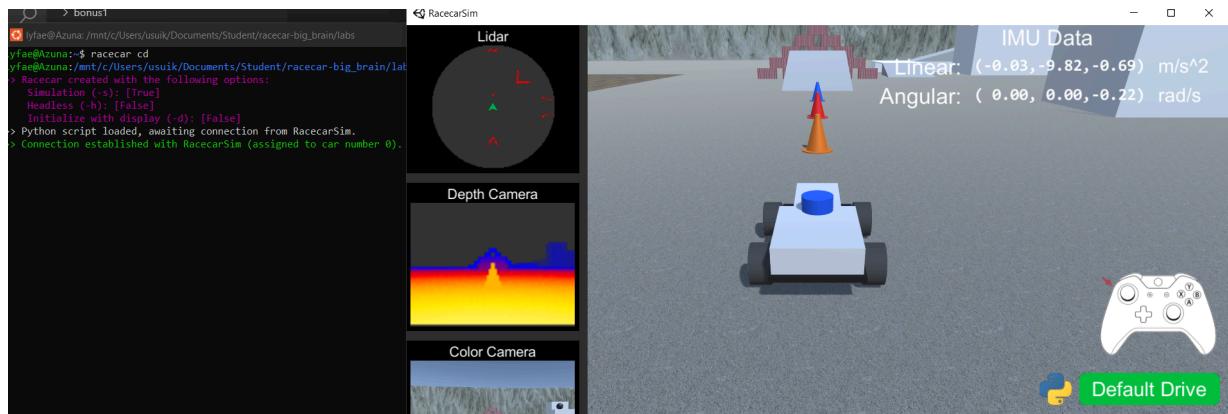
1. Open a Windows Powershell terminal on your computer:



2. Once in powershell, check what version of WSL you are on by using this command:
"wsl --list --verbose" (two dashes on both list and verbose)

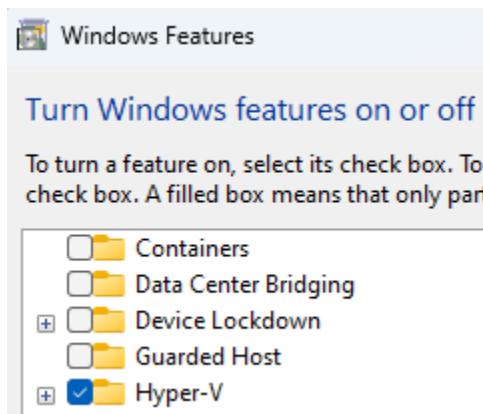
```
PS C:\Users\usuik> wsl --list --verbose
  NAME      STATE      VERSION
* Ubuntu    Running     1
PS C:\Users\usuik>
```

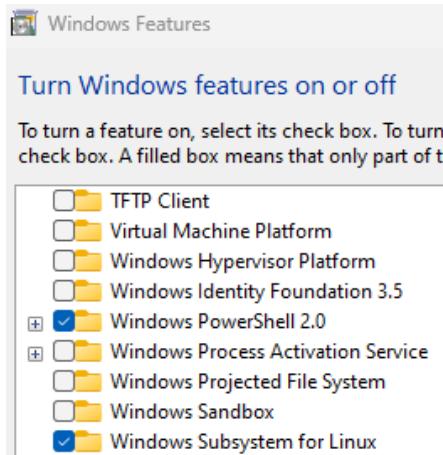
3. If the version states 2, run the command “**wsl --set-version Ubuntu 1**” to change the version from WSL 2 back to WSL 1. (two dashes on set, one dash on version)



Error: 0x800701bc WSL 2 requires an update to its kernel component

Ensure that the following is checked in your “Windows Feature” tab:





Windows Subsystem for Linux

Use this command as a last resort

```
dism.exe /online /enable-feature mode /featurename:VirtualMachinePlatform /all  
/norestart
```

Ubuntu account creation error

Error Message: Please enter a username matching the regular expression configured via the NAME_REGEX[_SYSTEM] configuration variable. Use the '--force-badname' option to relax this check or reconfigure NAME_REGEX.

Solution: Make sure the username does not have bad characters. All lowercase letters work the best.

Ubuntu: The system cannot find the path specified.

```
Run with cmd administrator  
wsl –shutdown  
wslconfig /unregister Ubuntu  
wsl –install –distribution Ubuntu
```

Help, I forgot my Ubuntu password!

<https://winaero.com/blog/reset-password-wsl-linux-distro-windows-10/>

Or, you can uninstall and reinstall Ubuntu, which puts you through the setup procedure again.

Mac: code . command does not exist!

<https://code.visualstudio.com/docs/setup/mac#:~:text=Launch%20VS%20Code.,code%20command%20in%20PATH%20command.>