

WSL2 on Windows

Install Guide

# 

# Table of Contents

[Table of Contents 2](#_Toc126393108)

[Introduction 3](#_Toc126393109)

[Install WSL 3](#_Toc126393110)

# 

# Introduction

Windows 10 Pro now provides a VERY functional Ubuntu command window (In Linux we call this a “Shell” window) that provides almost 100% compatibility with Ubuntu Linux. In fact, it leverages the Windows HyperV technology but does not require building a full blown VM. Microsoft calls this the “Windows Subsystem for Linux Version 2” or “WSL2”.

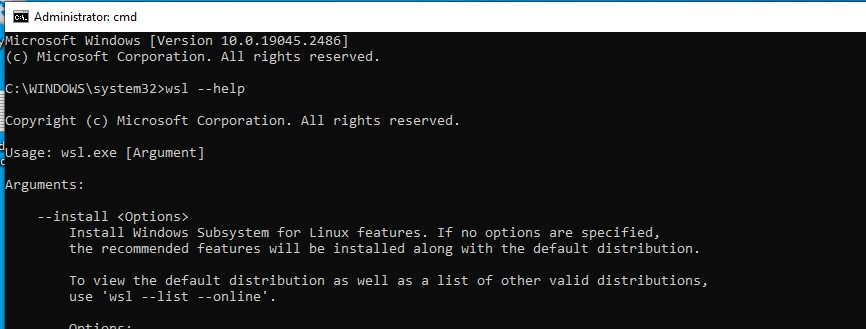
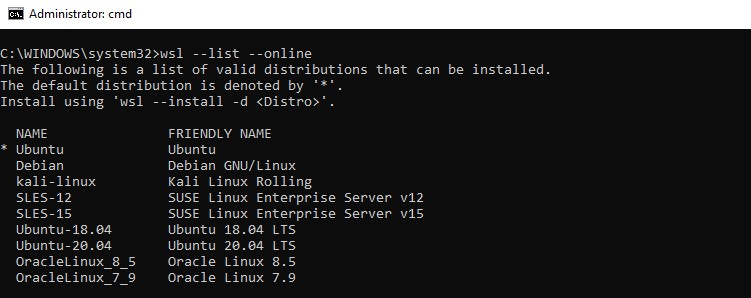
We will be walking through setting up WSL2.0 on your Windows 10 laptop/desktop. WSL2.0 is, for all intents and purposes, a native implementation on Windows without actually running a Linux VM. WSL2.0 is fast and 100% compatible with a standard Ubuntu distribution (one may even install linux packages in WSL2.0).

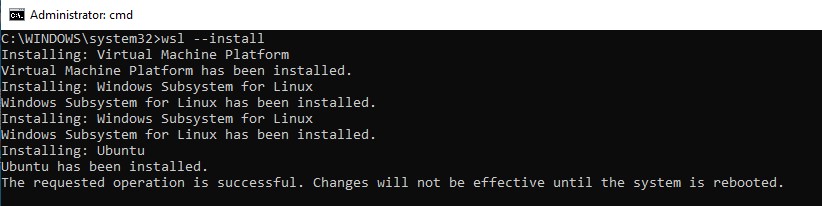
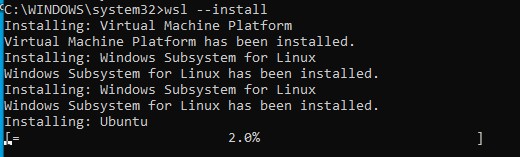
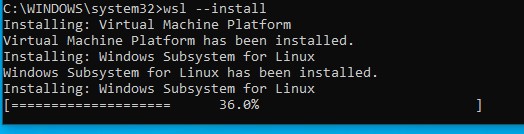
# Install WSL

* On your Windows 10 desktop, open a command window as "Adminstrator"

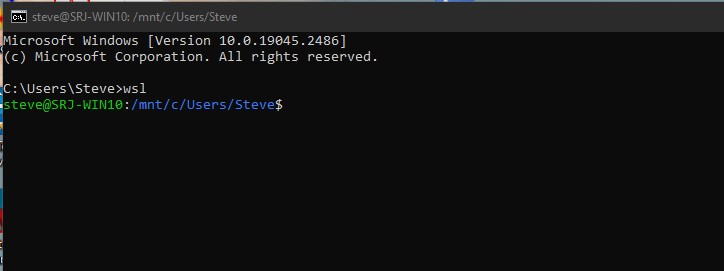
Text

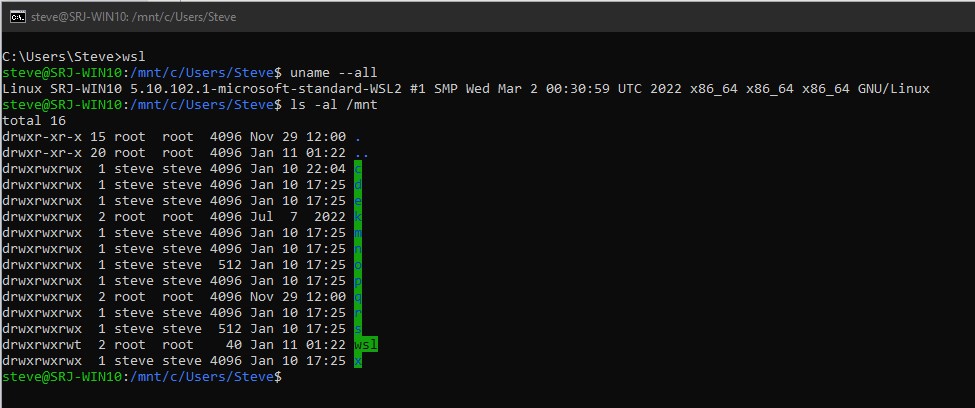
Description automatically generated

* Type in the command "wsl --help" and read about the option for installing WSL:
* Now take a look at the available distributions that we can install in WSL:
* In our class we will be using Ubuntu as the learning platform, so go ahead and install the Ubuntu distribution into WSL:



* And our final product -- a full featured Linux command window.
* Reboot your system per the above instructions
* Note below a few points.
  + There are some issues with running the WSL system and VMware Workstation (VMW). Most are related to running multi-nested hypervisors under VMW.
  + Start the Linux window by opening a Windows Command Prompt and issuing the command "wsl" (see screenshot).



* Note that all Windows drive letters are accessible via the "/mnt/<drive\_letter>" path.
* Note that "uname" shows a real Linux distribution.
* At this point you have a full featured Linux system running on your Windows system. One is able to install packages (via 'apt'), write BASH scripts, even use things such as 'git' in order to create a version-controlled environment (in fact, the screenshots used in this presentation are all sourced from GITHUB that is a remote repository on a WSL installation.