

# GitBash Install

In this short tutorial, I will be demonstrating the installation of [GitBash](#). **GitBash** is packaged with many Unix/Linux utilities, as well as, the core Git commands for managing [GitHub](#) repositories.

Although I use git to manage my GitHub repositories, my main interest in **GitBash** is to have access to the many Unix/Linux utilities in my Windows environment. Please note that I will be accepting most of the defaults except for when we reach the **Adjusting your PATH environment** screen. Feel free to make any other setup selections based on your needs.

## Prerequisites

- Active internet connection
- GitBash can be downloaded [here](#)

Steps to complete tutorial:

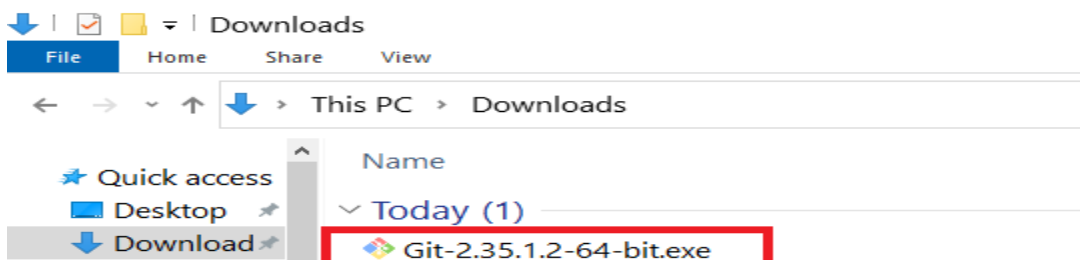
- [Download GitBash](#)
- [Install GitBash](#)
- [Test Unix/Linux Utilities](#)

## Download GitBash

First, go to the following link <https://git-scm.com/download/win> and download your system's version. I am on Windows 10, so I chose the 64-bit standalone installer.

## Install GitBash

From your system's download location, launch the GitBash installer.

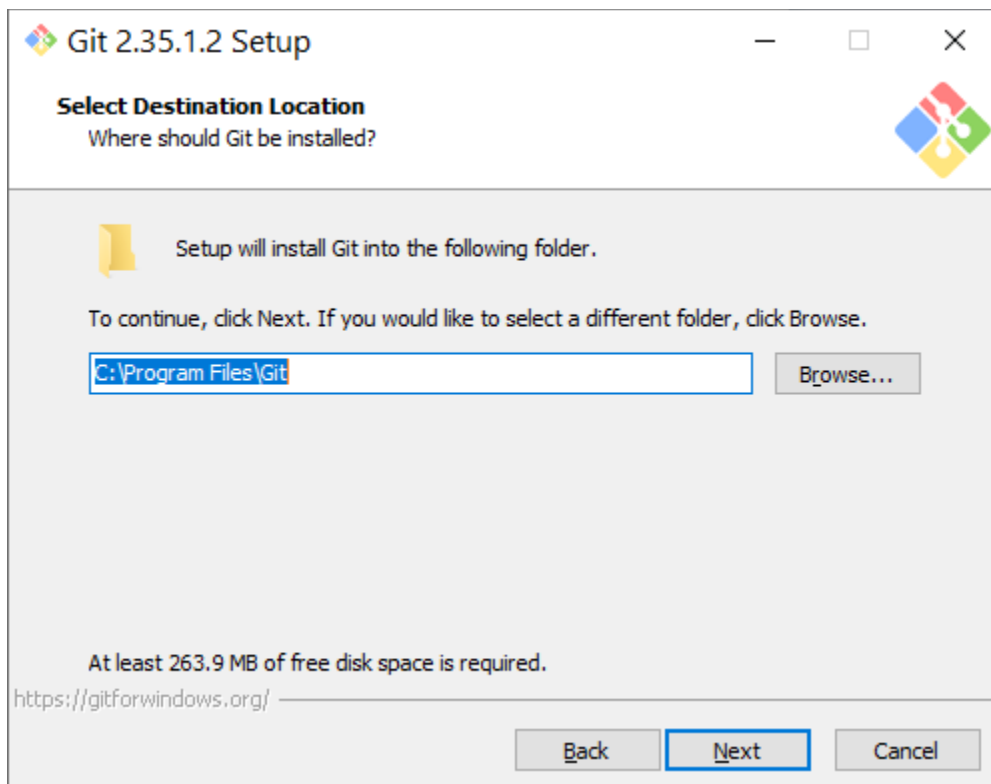


If prompted by User Account Control to proceed, click **Yes**.

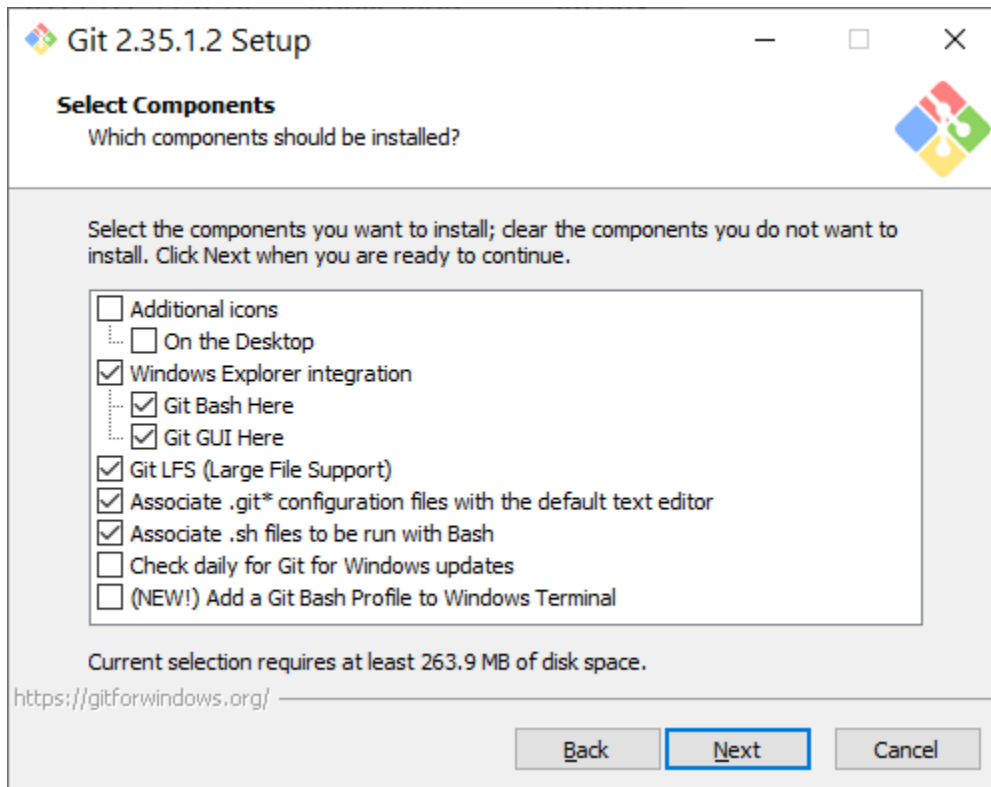
To begin the installation, click **Next**.



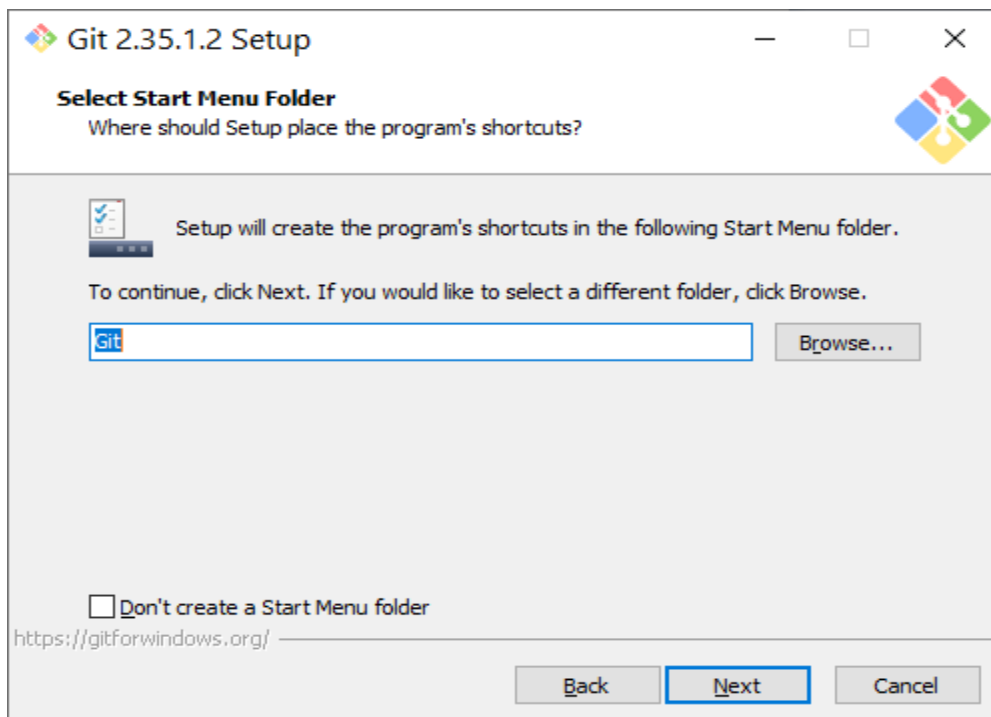
I have left the default installation destination and clicked **Next**.



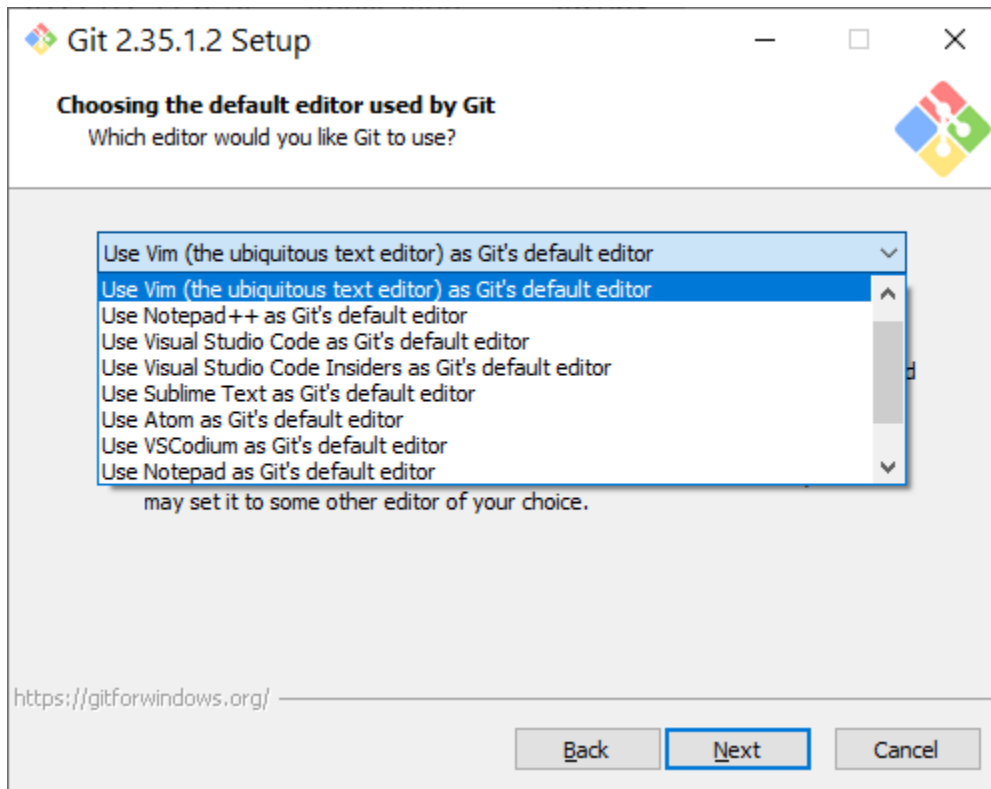
I went with the default components selected and clicked **Next**.



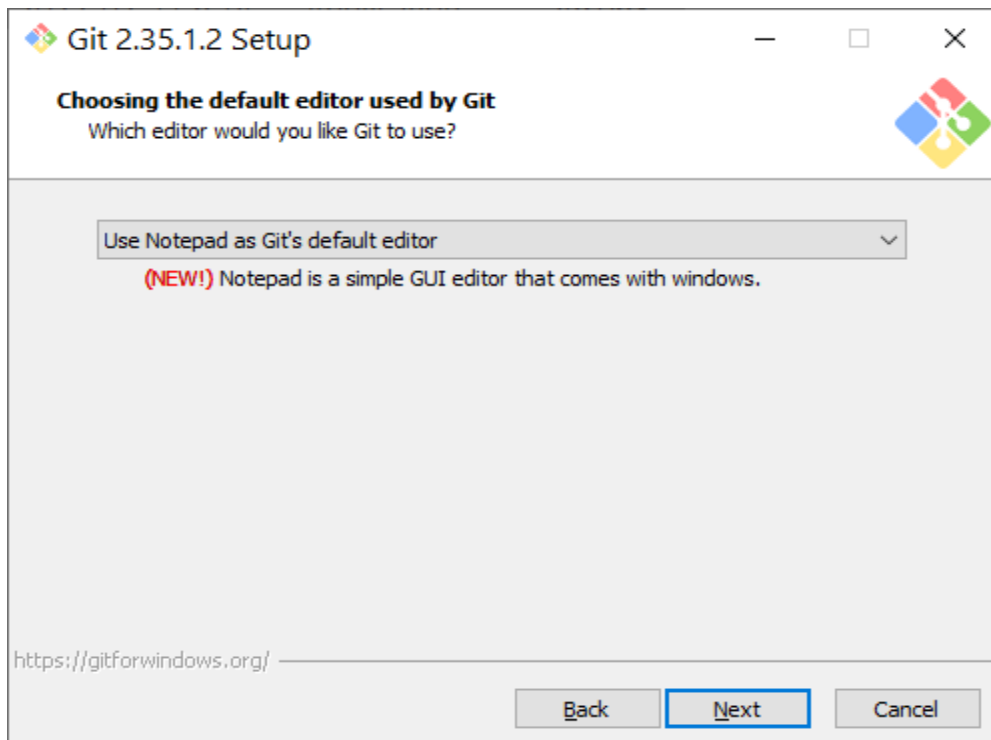
Accept the Start Menu Folder and click **Next**.



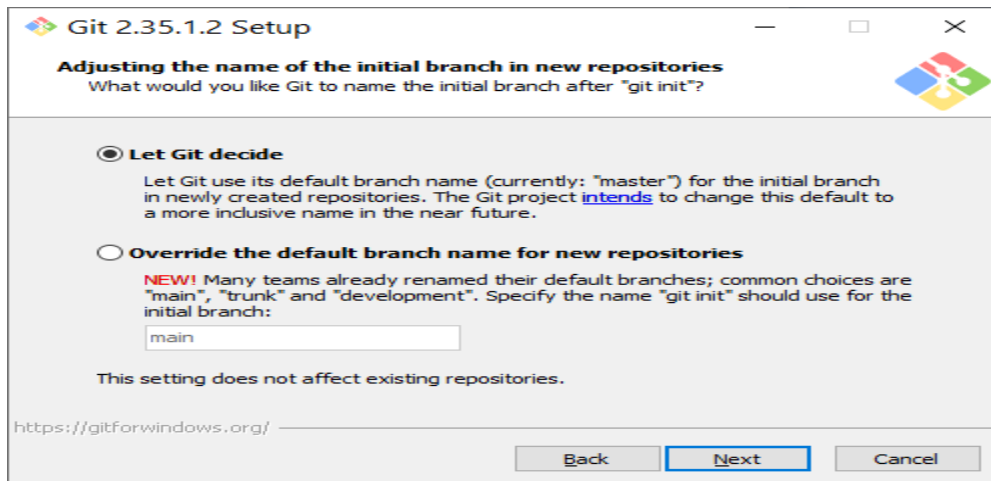
The next step is to choose the default editor used by Git. Note that the default is **Vim**. If you are not comfortable using **Vim**, select another default editor.



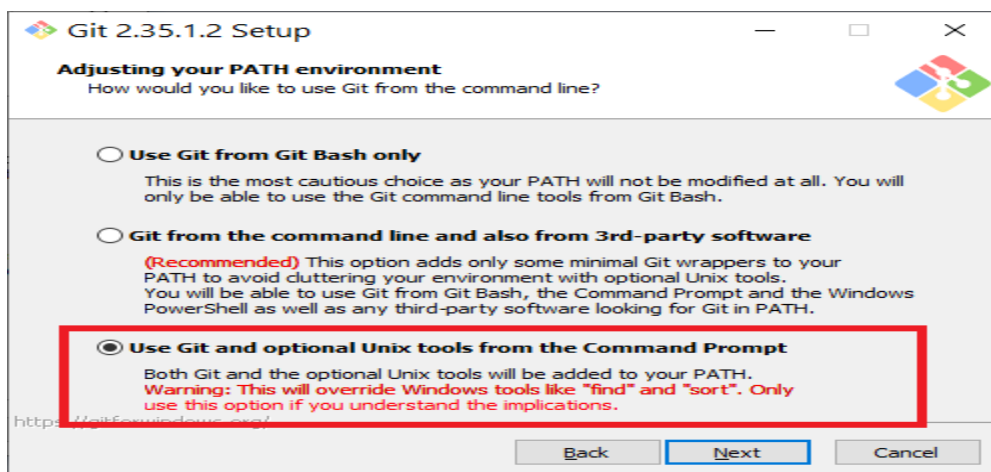
Although I use Vim, I will choose a different editor. I have selected **Notepad** and clicked **Next**.



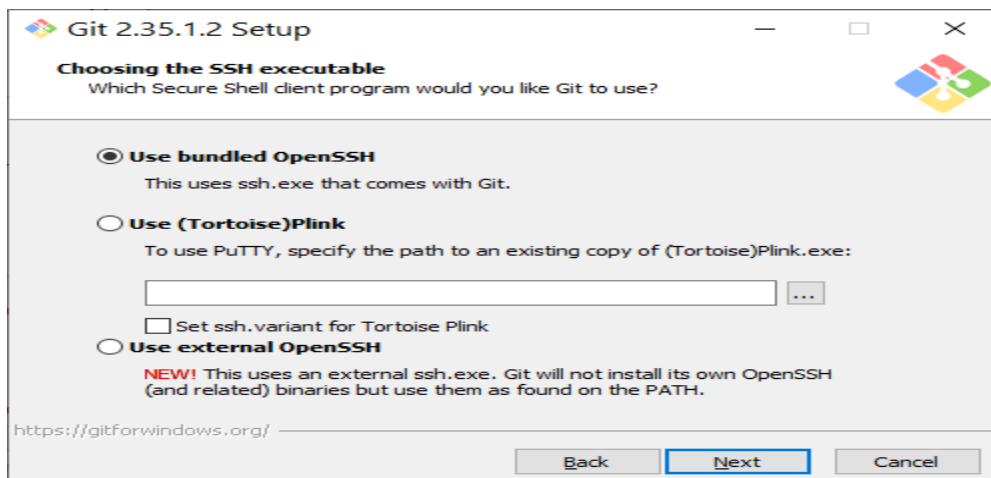
Again, I went with the default and clicked **Next**.



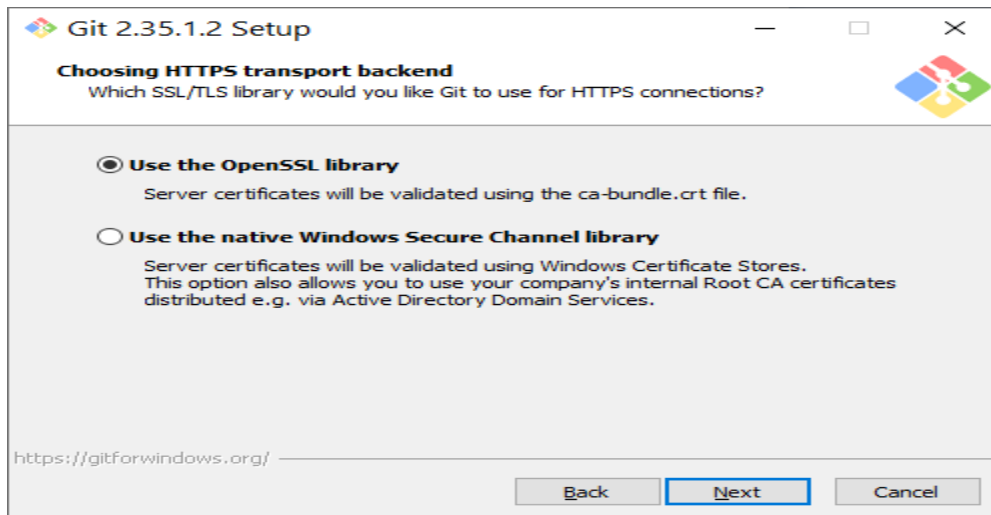
In the next step, this setup selection will allow us to use Unix/Linux utilities in a Windows environment. Ensure that option is selected and click **Next**.



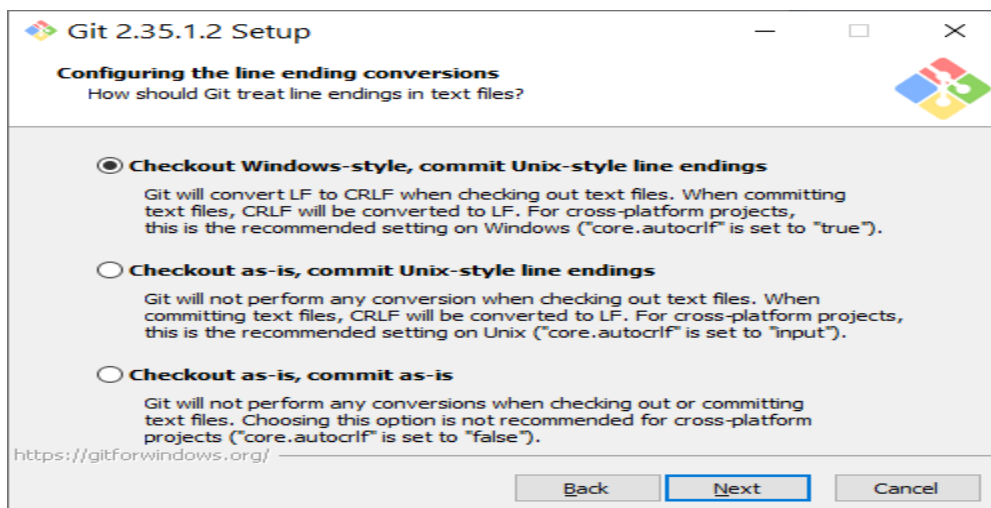
I will be leaving the default for SSH and clicking **Next**.



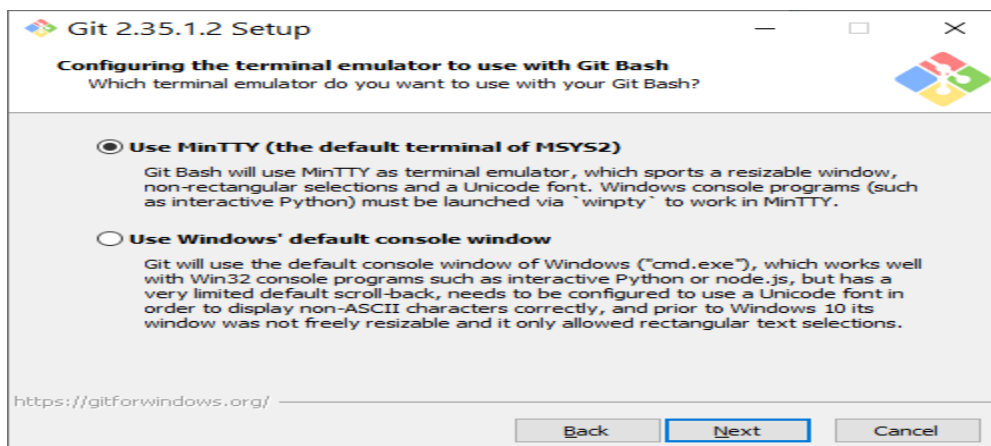
Again, leaving the default selection and clicking **Next**.



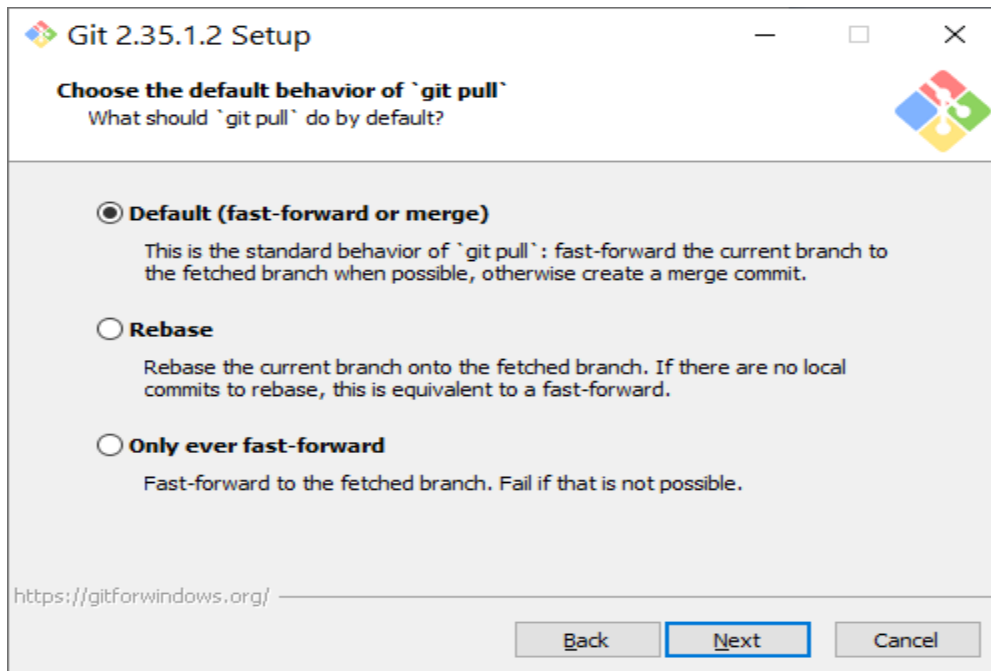
Leaving the default and clicking **Next**.



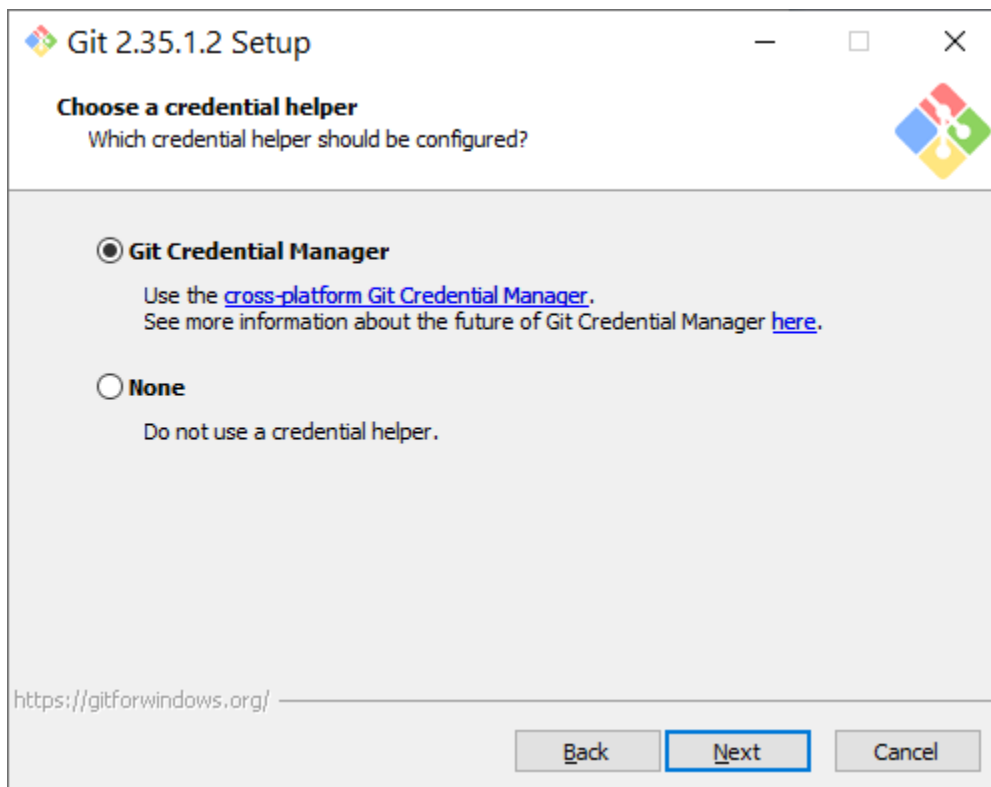
Leaving the default and clicking **Next**.



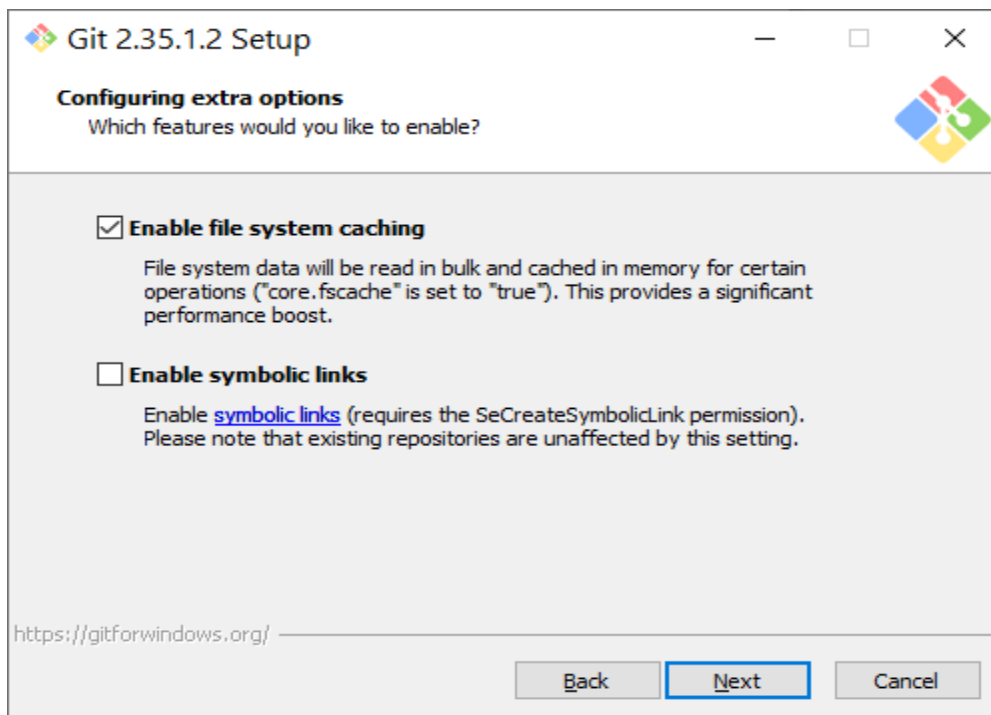
Leaving the default and clicking **Next**.



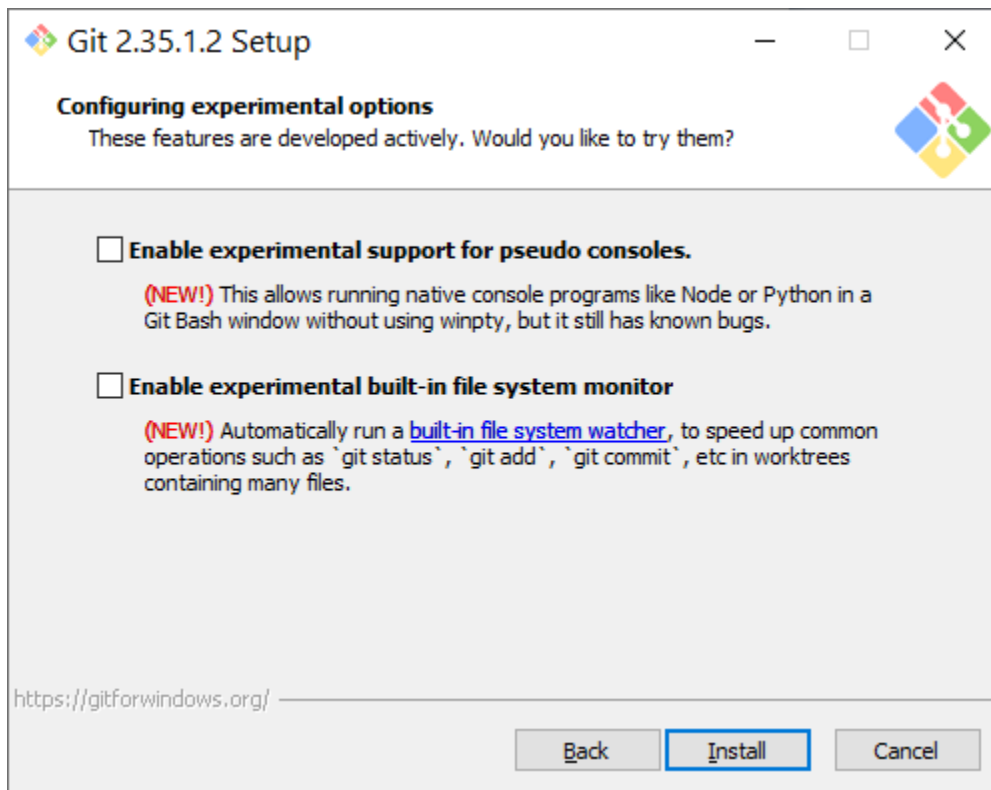
Leaving the default and clicking **Next**.



Leaving the default and clicking **Next**.

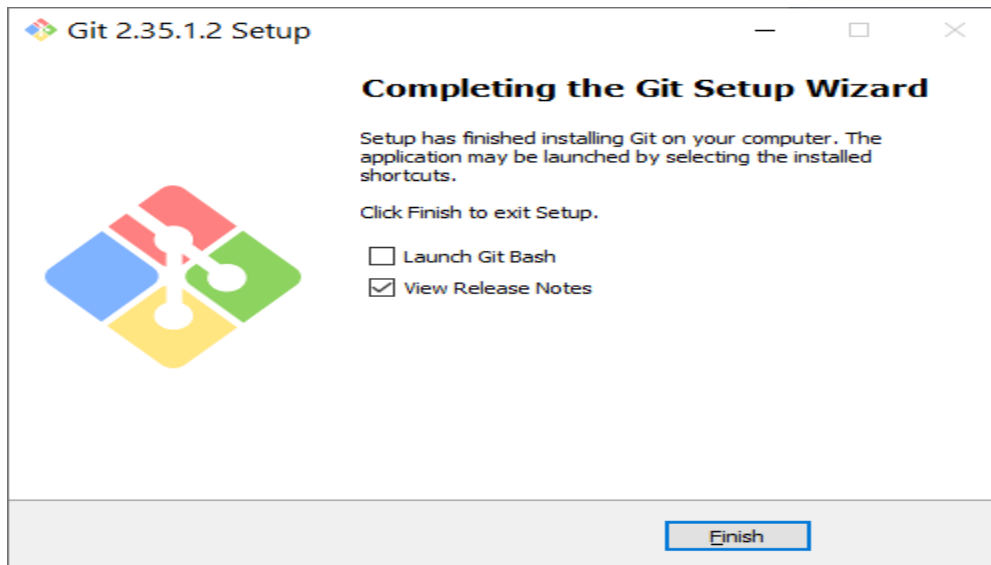


Leaving the default and clicking **Next**.

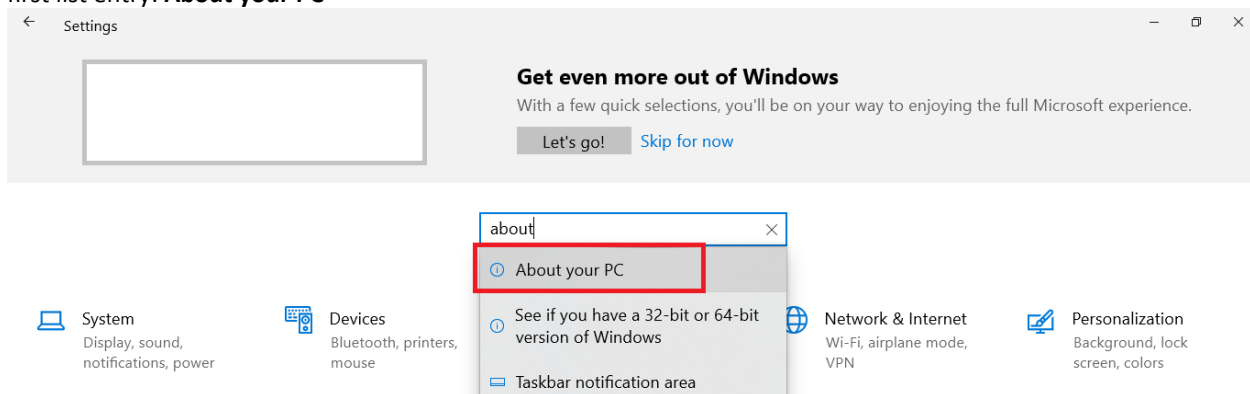




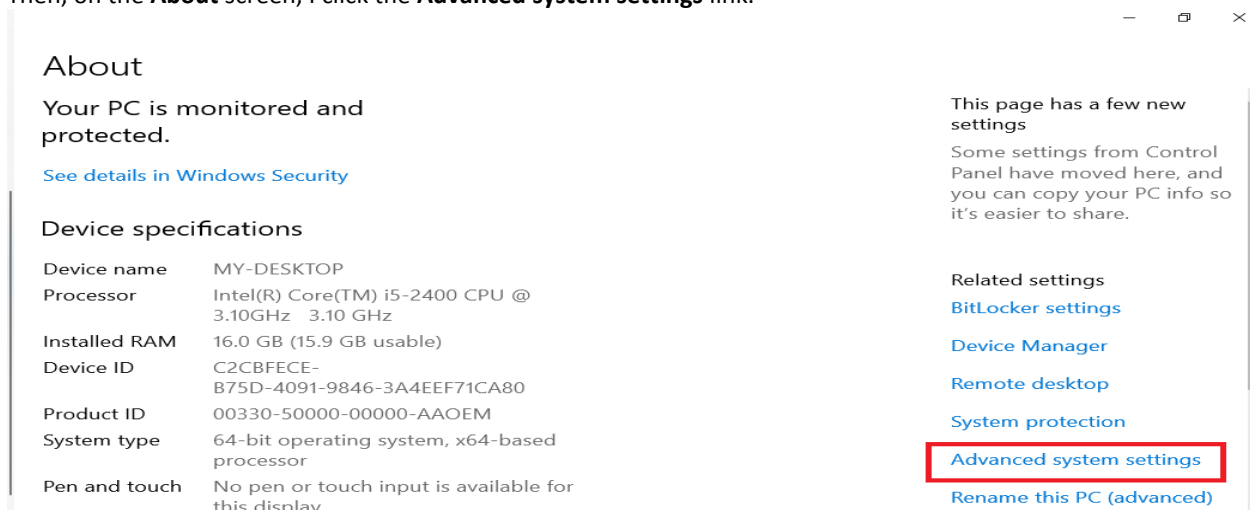
Installation has completed successfully.



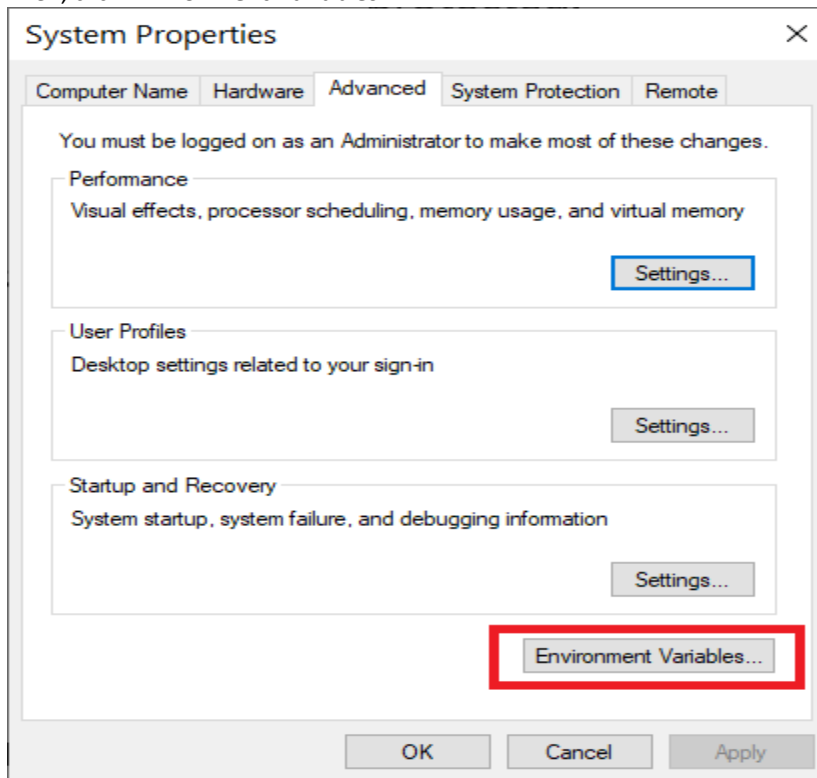
We will now confirm that our **PATH** environment variable has been modified to include Git. I am on Windows 10, so I first open my **Settings** window, then I enter **about** in the search field. Next, I choose the first list entry: **About your PC**



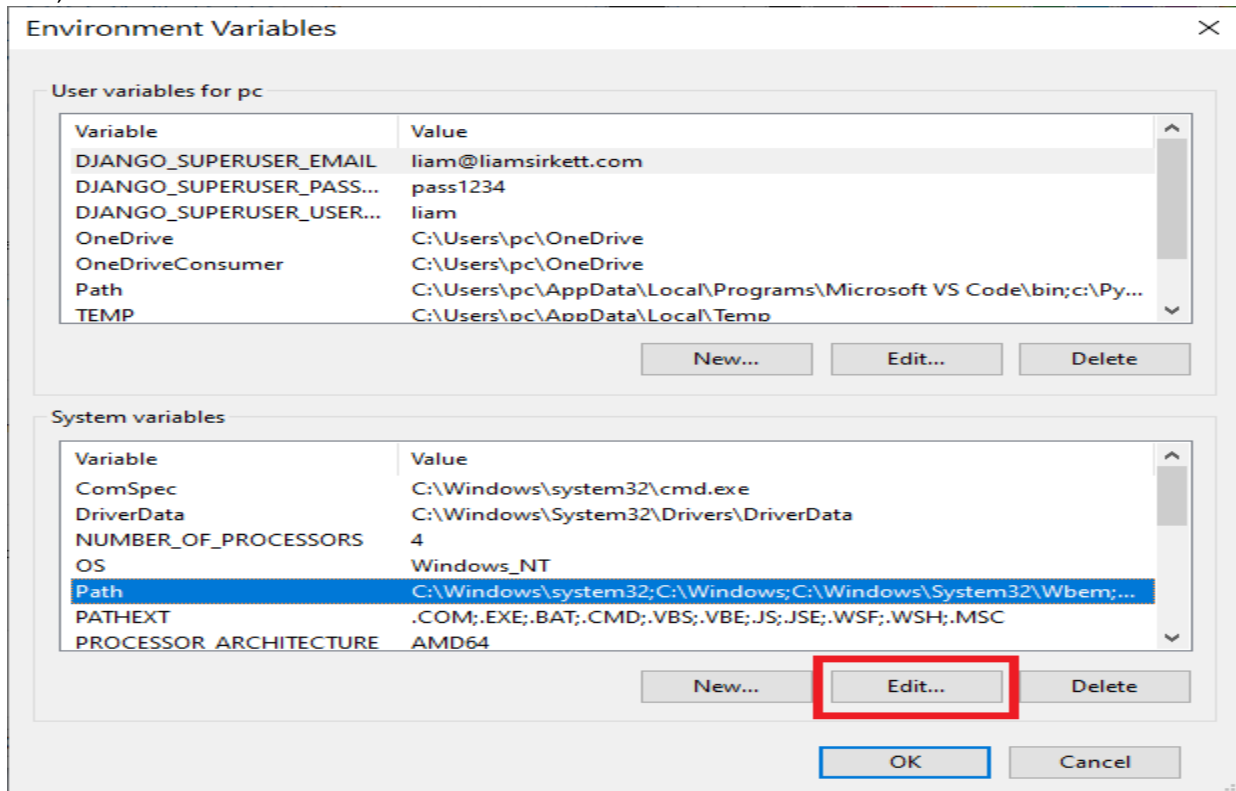
Then, on the **About** screen, I click the **Advanced system settings** link.



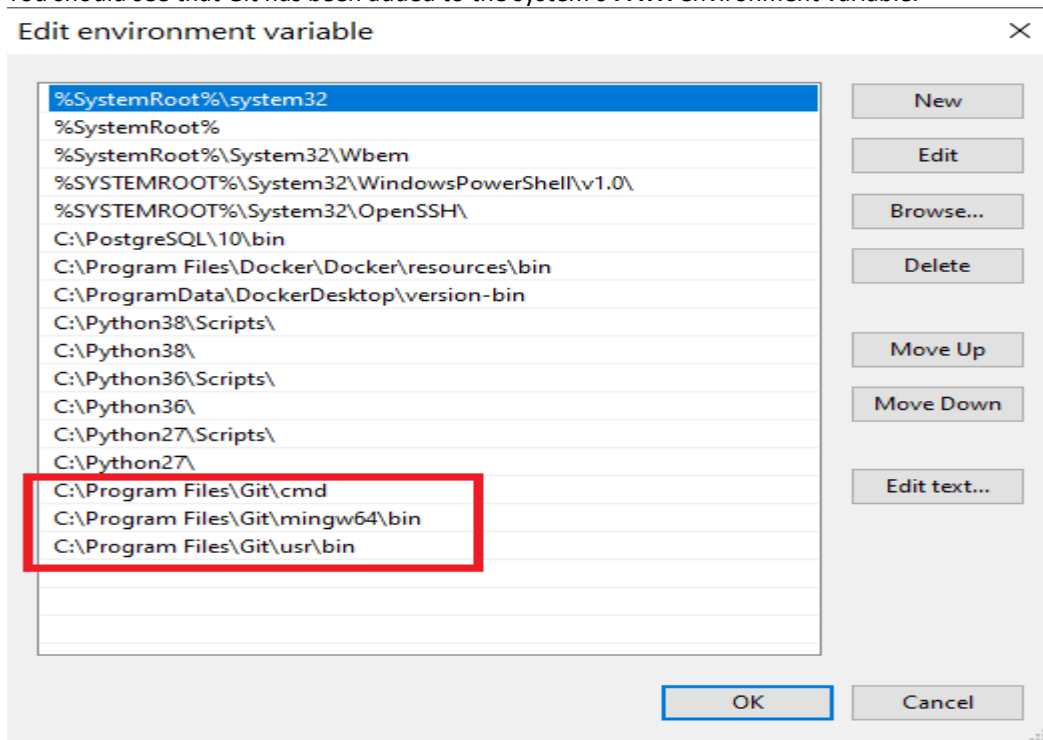
Then, click **Environment Variables**



Now, ensure **Path** is selected and click **Edit**



You should see that Git has been added to the system's **PATH** environment variable.



## Test Unix/Linux Utilities

I have opened a PowerShell window and executed a few commands

```
cd D:\0-FINAL # cd is the aliased PowerShell cmdlet Set-Location
ls # ls is the aliased PowerShell cmdlet Get-ChildItem
ls | grep -e "_start_*.ps1" # grep & wc are both Unix/Linux utilities
ls | grep -e "_start_*.ps1" | wc -l # made available by GitBash
```

```
PS C:\Users\pc> cd D:\0-FINAL\
PS D:\0-FINAL> ls

Directory: D:\0-FINAL

Mode                LastWriteTime         Length Name
----                -
d-----          2022-04-08    8:30 AM             LOGS
d-----          2022-04-08    8:27 AM          OLD-LOGS
d-----          2022-04-07    8:19 AM             ORIG
-a-----          2022-04-07    8:31 AM        1141 auto_start_vm_1_final.ps1
-a-----          2022-04-09    6:32 AM        3381 auto_start_vm_2.ps1
-a-----          2022-04-08    6:48 AM        3651 auto_start_vm_2_wip.ps1
-a-----          2020-12-17    4:44 AM        3735 auto_stop_VM-min.ps1
-a-----          2022-04-07    8:31 AM        1845 auto_stop_vm_1_final.ps1
-a-----          2022-04-07    6:43 AM        2782 auto_stop_vm_2_wip.ps1
-a-----          2022-04-09    5:04 AM         713 get_vm.ps1

PS D:\0-FINAL> ls | grep -e "*_start_*.ps1"
-a-----          2022-04-09    6:32 AM        3381 auto_start_vm_2.ps1
PS D:\0-FINAL> ls | grep -e "*_start_*.ps1" | wc -l
1
PS D:\0-FINAL>
```

Hopefully, you've enjoyed completing this tutorial and found it helpful.

Completing this tutorial is a prerequisite for my **Auto-Start Virtual Machine Part 2** tutorial accessible [here](#).

My main tutorials page can be accessed [here](#).

[Back to top](#)