Installing and Using the Windows Subsystem for Linux

Installing the Windows Subsystem for Linux (WSL) on your computer allows you to run Linux command line utilities directly on Windows. WSL only works on Windows 10 64-bit Version 1709 or later. This guide shows you how to install and use the Ubuntu WSL. You can use any Linux distribution in the Microsoft Store, but the steps for another distribution are likely different. The Ubuntu terminal will allow you to use command-line tools such as bash, git, ssh, and sftp. You can install other tools using apt.

1 Installing WSL

You can install WSL by doing the following:

- 1. Search "Turn Windows features on or off" in the Windows search bar and open it.
- 2. Look for "Windows Subsystem for Linux", make sure the box is checked, and hit OK.
- 3. Restart your computer.
- 4. Open the Microsoft Store application and search "Ubuntu".
- 5. Click on Ubuntu, hit get, and wait for it to install. Note: You can use any version, but the one without the version number will give you the latest version of Ubuntu.
- 6. Launch the Ubuntu application and allow first time setup.
- 7. Create a username and password. Note: When entering a password, no characters will appear, but you are typing your password.
- 8. Update Ubuntu with the command:

```
sudo apt update && sudo apt upgrade
```

You will have to enter your password. You should run this command periodically as Ubuntu will not update on its own.

You have successfully installed and setup WSL and can now use the terminal as you would on Ubuntu.

2 Installing g++

g++ is a C++ compiler. You can install g++ in Ubuntu by doing the following:

- 1. Open the Ubuntu terminal
- 2. Install g++ with the command:

sudo apt install g++

You have successfully installed g++ and can now use it to compile your C++ programs in WSL.

3 Installing make

In order to use makefiles, you must install make. You can install make in Ubuntu by doing the following:

- 1. Open the Ubuntu terminal
- 2. Install make with the command:

```
sudo apt install make
```

You have successfully installed make and can now use makefiles in WSL.

4 Accessing Your Windows Files

Accessing where your Window files are stored through WSL may be a little tricky. To access your files in Windows, you must first change into the directory /mnt, which you can do with the command:

cd /mnt

You can now use cd (change directory) to access your files in Windows. If any file or folder contains a space in it, you must put it in quotation marks e.g. "CS Projects". You can list the contents of your directory with the command:

ls

For example, if your files are stored in your documents folder, you can access them by typing the command:

cd /mnt/c/Users/username/Documents

You want to enter the username set for your account after Users. This is likely different than the username you created for Ubuntu.

If you don't know your username, search cmd in the Windows search bar and open it. Then type the command:

echo %username%

This should be the username you want to put for username in the example command. If your username contains a space, you must put it in quotation marks e.g. "Alan Turing".

5 Writing Scripts

Writing a script can make doing a repetitive task quicker to do. For example, a script make it quicker to execute a long change directory command. You can write a simple script with the following steps:

- 1. Open the Ubuntu terminal
- 2. Open a new file with the command:

```
vi script_name.sh
```

where script_name is what you want your new script to be named or the name of an existing script you want to edit.

- 3. Press "i" for insert to write to the file.
- 4. Type:

#!/bin/bash

This is how bash scripts must start, to signify that we are using bash commands.

5. Type what commands you want the script to run. For example, if you wanted it to your documents folder, you would type:

```
cd /mnt/c/Users/username/Documents
```

- 6. When finished, press "ESC" to leave insert mode.
- 7. Save the file and quit vi by typing:

:wq

8. Give the script permission to run with the command:

```
chmod +x script_name.sh
```

9. Execute your script by typing:

```
./script_name.sh
```

If your script contains the command cd, you must run it with the following:

. ./script_name.sh

Without the extra dot, changing directories will not take effect in your terminal.

6 Troubleshooting

If you are having issues with WSL please consult the following document: https://docs.microsoft.com/en-us/windows/wsl/troubleshooting

7 Questions and Feedback

If you have questions about or feedback for this guide, please email them to projects AT owenkunhardt DOT com.