Visual Studio Code with WSL (Ubuntu linux terminal)

WSL is a Linux based kernel developed for running on a windows 10 operating system. To install this environment on a windows PC, and thereafter incorporate it into VSC, follow this guide.

The WSL kernel comes, after a couple of installations, with a C and C++ Compiler, which you can access directly from the Terminal in VSC.

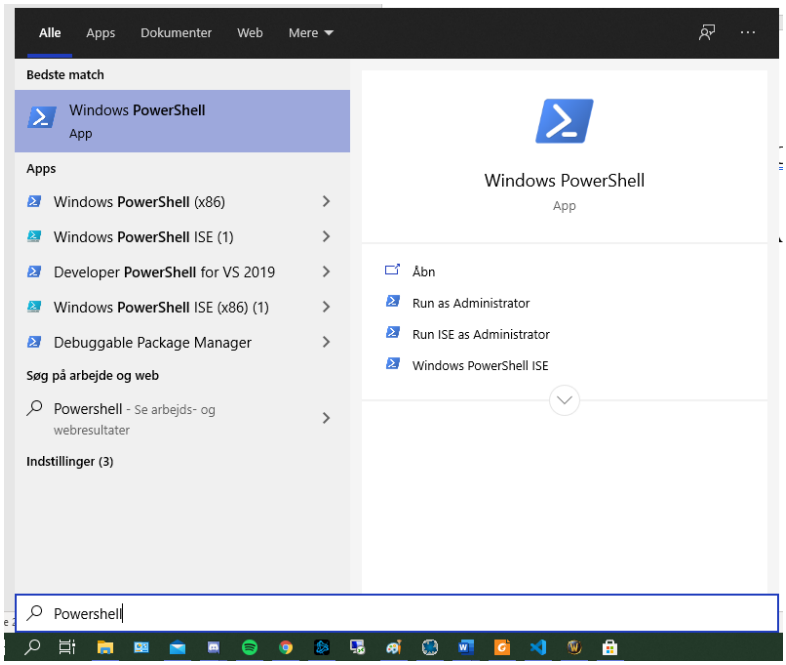
Like so:

```
lillevang@DESKTOP-CFVRJ4H:/mnt/c/Users/Philip/OneDrive/Skrivebord/Basic C programmering - Eksempler$ gcc 002_Printf_operators.c -o print
lillevang@DESKTOP-CFVRJ4H:/mnt/c/Users/Philip/OneDrive/Skrivebord/Basic C programmering - Eksempler$ ./print
This program is designed to show case print operators:
======================
Type 0 to terminate the program
Type 1 to get information about operators
Type 2 for a showcase
======================
```

Part 1: Install Ubuntu from the Microsoft store

Part 2: Enable WSL on your PC

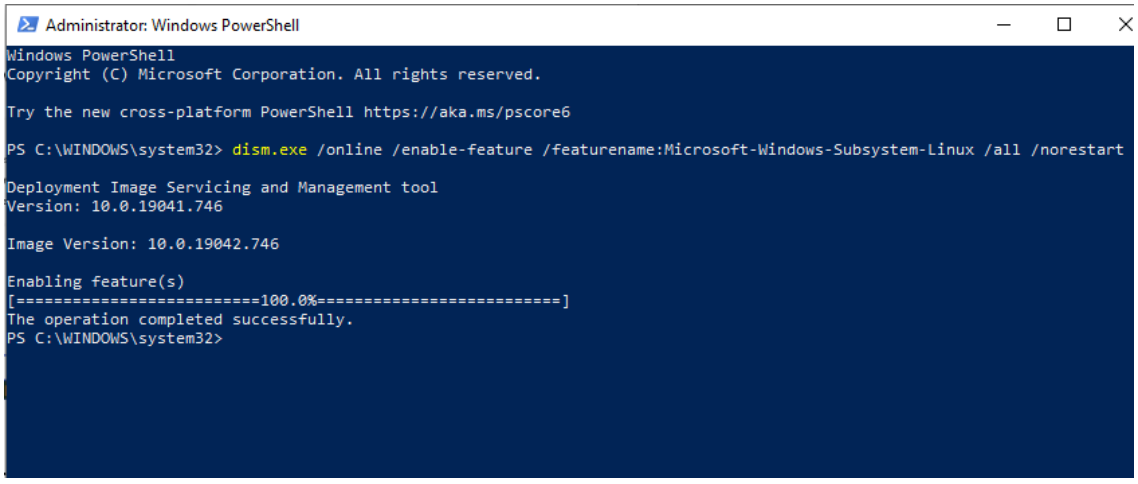
Step 1: Open your Powershell Terminal as Administrator, by right-clicking on it and pressing “Run as administrator”.



When you have opened your Terminal, Write this line, and Restart your Computer on Success

```
dism.exe /online /enable-feature /featurename:Microsoft-Windows-Subsystem-Linux /all /norestart
```

Copy paste it in, like this:



Now, open the Ubuntu Terminal, Setup a Custom Username and Password, (the password is hidden), you have to write it twice and press enter. This is only needed if you do not open it with admin rights.

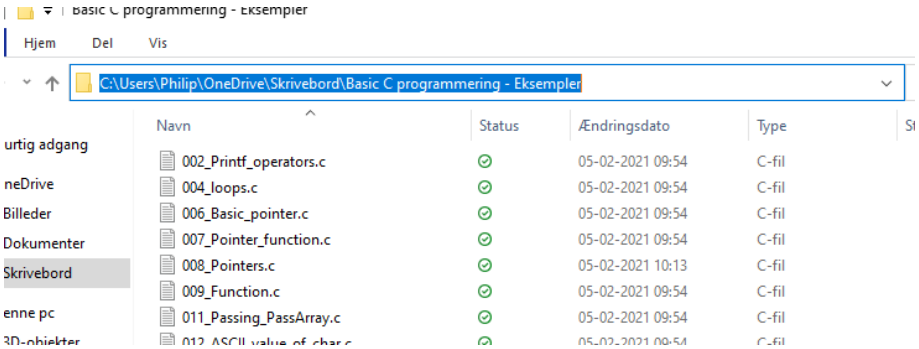
Now you write these 3 Commands in this order

sudo apt-get update

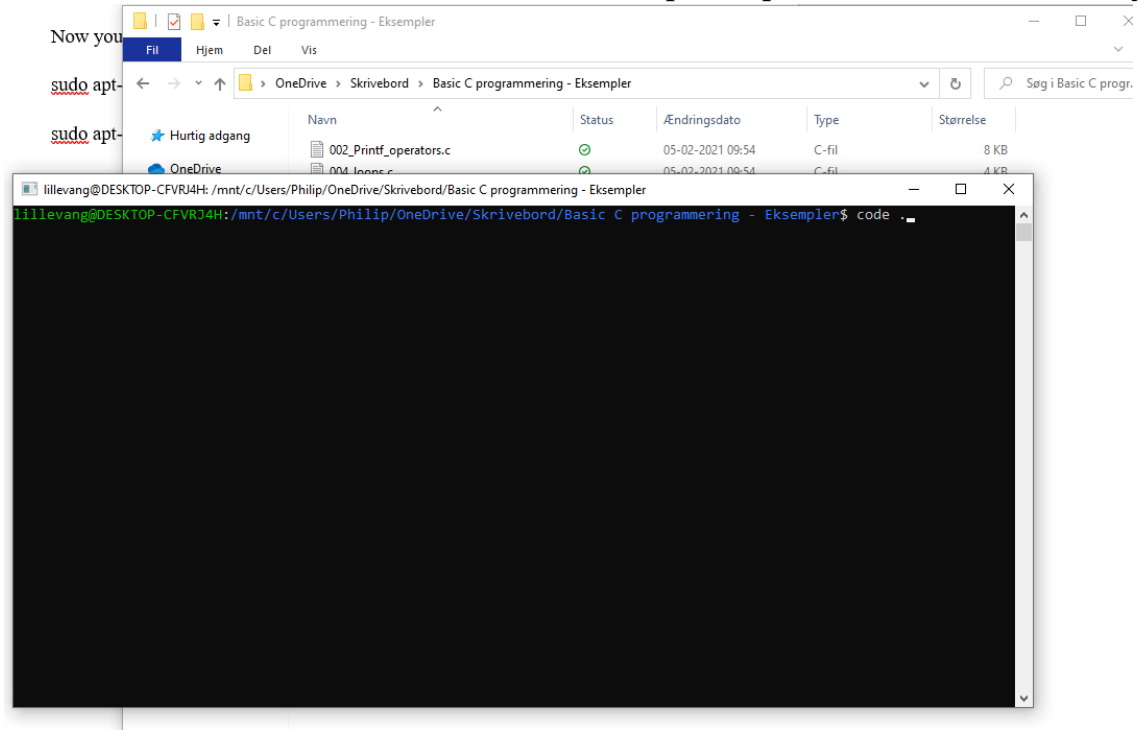
sudo apt-get upgrade

sudo apt-get install build-essential

Then, after you finished the installations, you can open your filepath, and find your Workspace of choice:



Delete the blue text, and write “bash”, this opens up ubuntu in that Directory.



Now you can write “code .” to open any Directory like this, in VSC

THINGS TO REPEAT:

- Opening VSC from the filepath, in the appropriate Filepath
- 2 commands!
- Gcc YourFile.c -o YourFileName (-o give you a chance to rename the file, else it’s just called “a.out)
- “./YourFileName” or “./a.out” to run the program