

Mini Project 3

MiniChess AI

WSL Installation Tutorial

Outline

- Introduction
- Install WSL
- Setup WSL
- Install build-essential
- Compile in WSL
- Execute in WSL

Introduction

- We have noticed that executing the project on Windows is much slower than on Mac or Linux
- There also exists some serious problems like file remove error or baselines got killed too early and failed to output action
- For Windows users, we **recommend using WSL in this project** to avoid these problems

Windows Subsystem for Linux

- WSL enables your Windows computer to run Linux
- Since the final evaluation will take place on a machine with Linux environment, using WSL during development is a good choice
- Using WSL makes your code execute in normal speed
 - Without the lags in Windows
- Also prevents the remove file and baseline invalid move error

Outline

- Introduction
- **Install WSL**
- Setup WSL
- Install build-essential
- Compile in WSL
- Execute in WSL

Go to [WSL Installation Guide](https://docs.microsoft.com/en-us/windows/wsl/install-win10)

The screenshot shows a web browser displaying the Microsoft Docs page for 'Windows Subsystem for Linux Installation Guide for Windows 10'. The page is in dark mode and includes a left sidebar with navigation links like 'WSL Documentation', 'Overview', 'Quickstart', and 'Install WSL & update to WSL 2'. The main content area features the title 'Windows Subsystem for Linux Installation Guide for Windows 10' and a sub-header 'There are two options available for installing Windows Subsystem for Linux (WSL):'. It lists two options: 'Simplified install (preview release): wsl --install' and 'Manual install: Follow the six steps listed below.' A 'Note' box states: 'If you run into an issue during the install process, check the Troubleshooting installation section at the bottom of this page.' Below this, the 'Simplified Installation for Windows Insiders' section explains that the installation process has been improved in the latest Windows Insiders preview builds, replacing manual steps with a single command. It lists requirements: joining the Windows Insiders Program, installing a preview build of Windows 10 (OS build 20H2 or higher), and opening a command line window with Administrator privileges. The right sidebar contains a 'Is this page helpful?' section and a table of contents for the article.

Microsoft | Docs | Documentation | Learn | Q&A | Code Samples

Windows / Development environment / WSL / Quickstart / Install WSL & update to WSL 2

Filter by title

WSL Documentation

Overview

Quickstart

Install WSL & update to WSL 2

Install on Windows Server

Create a user account & password

Tutorials

How-to

Frequently Asked Questions

WSL 2 FAQ

Troubleshooting

Release Notes

Windows Subsystem for Linux Installation Guide for Windows 10

04/07/2021 • 10 minutes to read • 🗨️ 🗨️ 🗨️ 🗨️ 🗨️

There are two options available for installing Windows Subsystem for Linux (WSL):

- Simplified install (preview release):** `wsl --install`

The `wsl --install` simplified install command requires that you join the Windows Insiders Program¹ and install a preview build of Windows 10 (OS build 20H2 or higher), but eliminates the need to follow the manual install steps. All you need to do is open a command window with administrator privileges and run `wsl --install`, after a restart you will be ready to use WSL.

- Manual install:** Follow the six steps listed below.

The manual install steps for WSL are listed below and can be used to install Linux on any version of Windows 10.

Note

If you run into an issue during the install process, check the Troubleshooting installation section at the bottom of this page.

Simplified Installation for Windows Insiders

The installation process for Windows Subsystem for Linux has been significantly improved in the latest Windows Insiders preview builds of Windows 10, replacing the manual steps below with a single command.

In order to use the `wsl --install` simplified install command, you must:

- Join the Windows Insiders Program¹
- Install a preview build of Windows 10 (OS build 20H2 or higher).
- Open a command line window with Administrator privileges

Once those requirements are met, to install WSL:

Is this page helpful?

Yes No

In this article

- Simplified installation for Windows Insiders
- Manual installation steps
- Step 1 - Enable the Windows Subsystem for Linux
- Step 2 - Check requirements for running WSL 2
- Step 3 - Enable Virtual Machine features
- Step 4 - Download the Linux kernel update package
- Step 5 - Set WSL 2 as your default version
- Step 6 - Install your Linux distribution of choice
- Install Windows Terminal (optional)
- Set your distribution version to WSL 1 or WSL 2
- Troubleshooting installation

Download PDF

Copy the command in step 1

Install WSL on Windows 10 | Microsoft Docs

Filter by title

- WSL Documentation
 - Overview
 - Quickstart
 - Install WSL & update to WSL 2
 - Install on Windows Server
 - Create a user account & password
 - Tutorials
 - How-to
 - Frequently Asked Questions
 - WSL 2 FAQ
 - Troubleshooting
 - Release Notes

Downloads and installs a Linux distribution (reboot may be required)

By default, the installed Linux distribution will be Ubuntu. This can be changed using `wsl --install -d <distribution name>`, (replacing `<distribution name>` with the name of your desired distribution.) Additional Linux distributions may be added to your machine after the initial install using the `wsl --install -d <distribution name>` command.

To see a list of available Linux distributions, enter `wsl --list --online`.

Manual Installation Steps

If you are not on a Windows Insiders build, the features required for WSL will need to be enabled manually following the steps below.

Step 1 - Enable the Windows Subsystem for Linux

You must first enable the "Windows Subsystem for Linux" optional feature before installing any Linux distributions on Windows.

Open PowerShell as Administrator and run:

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

We recommend now moving on to step #2, updating to WSL 2, but if you wish to only install WSL 1, you can now restart your machine and move on to Step 6 - Install your Linux distribution of choice. To update to WSL 2, wait to restart your machine and move on to the next step.

Step 2 - Check requirements for running WSL 2

To update to WSL 2, you must be running Windows 10.

- For x64 systems: Version 1903 or higher, with Build 18362 or higher.
- For ARM64 systems: Version 2004 or higher, with Build 19041 or higher.
- Builds lower than 18362 do not support WSL 2. Use the [Windows Update Assistant](#) to update your version of Windows.

To check your version and build number, select Windows logo key + R, type `winver`, select **OK**. Update to the latest Windows version in the Settings menu.

Download PDF

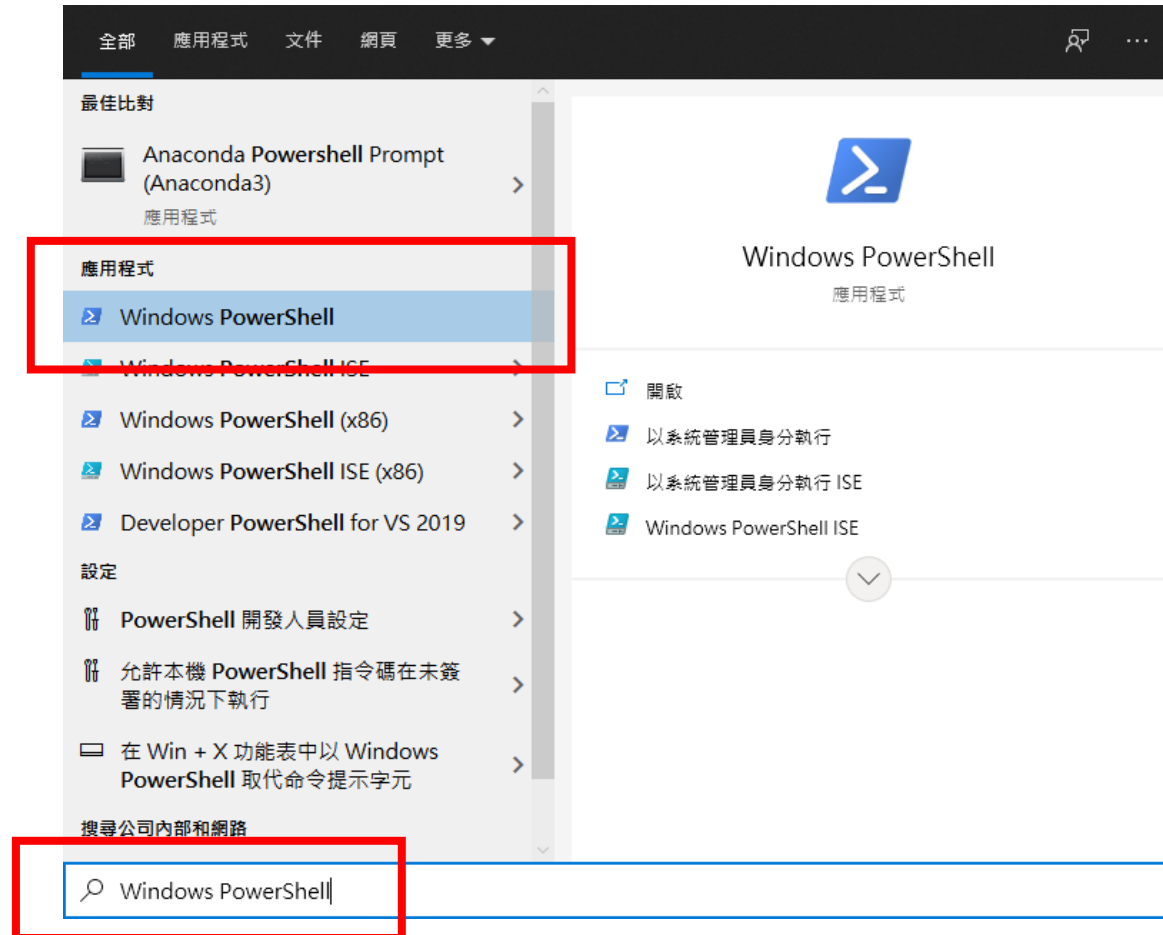
Is this page helpful?

Yes No

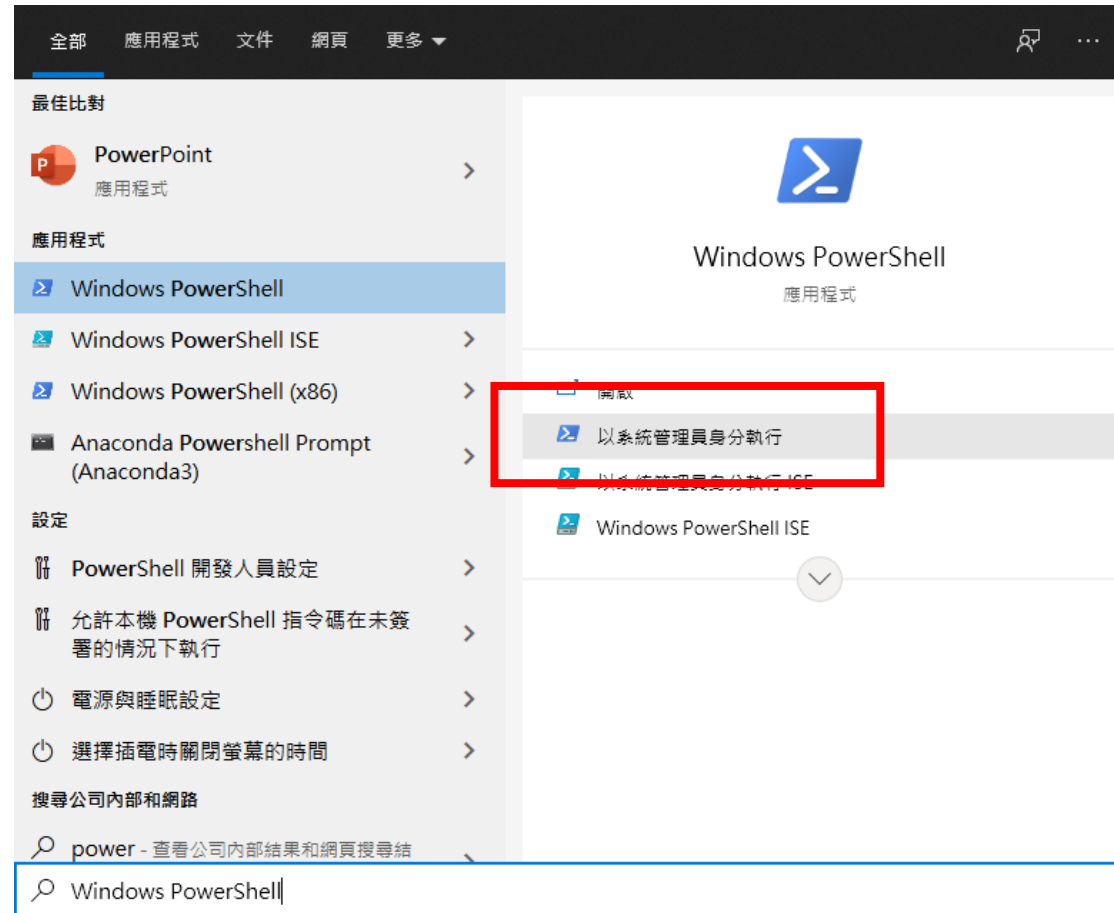
In this article

- Simplified installation for Windows Insiders
- Manual installation steps
 - Step 1 - Enable the Windows Subsystem for Linux
 - Step 2 - Check requirements for running WSL 2
 - Step 3 - Enable Virtual Machine feature
 - Step 4 - Download the Linux kernel update package
 - Step 5 - Set WSL 2 as your default version
 - Step 6 - Install your Linux distribution of choice
- Install Windows Terminal (optional)
- Set your distribution version to WSL 1 or WSL 2
- Troubleshooting installation

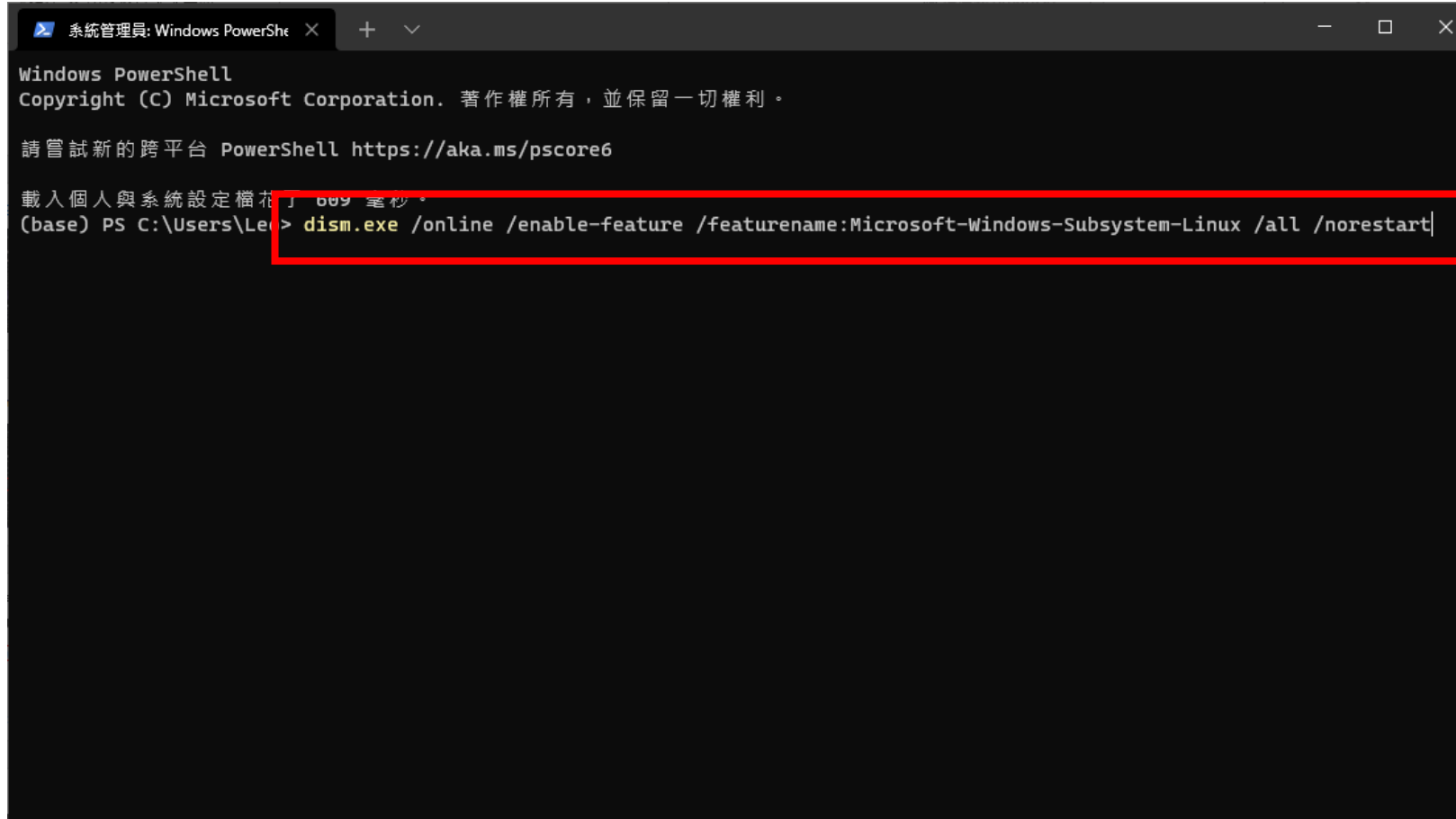
Search for PowerShell with search tool



Open PowerShell as Administrator



Paste the command and press enter



A screenshot of a Windows PowerShell terminal window. The window title bar shows '系統管理員: Windows PowerShell' and standard window controls. The terminal text includes: 'Windows PowerShell', 'Copyright (C) Microsoft Corporation. 著作權所有，並保留一切權利。', '請嘗試新的跨平台 PowerShell <https://aka.ms/pscore6>', and '載入個人與系統設定檔花了 009 毫秒。'. The command prompt shows '(base) PS C:\Users\Lee>'. The command 'dism.exe /online /enable-feature /featurename:Microsoft-Windows-Subsystem-Linux /all /norestart' is entered and highlighted with a red rectangular box.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. 著作權所有，並保留一切權利。

請嘗試新的跨平台 PowerShell https://aka.ms/pscore6

載入個人與系統設定檔花了 009 毫秒。
(base) PS C:\Users\Lee> dism.exe /online /enable-feature /featurename:Microsoft-Windows-Subsystem-Linux /all /norestart
```

Restart your computer

- Since WSL 1 is enough for this project, this slide will not go through step 2 ~ step 5 of the installation guide
- Restart your computer before you proceed to next step
 - If there is “reboot” and “update and reboot”, click “update and reboot”
- If you want to install WSL 2, you can follow the installation guide to complete step 2 ~ step 5 then restart your computer

Install Ubuntu 22.04.2 LTS



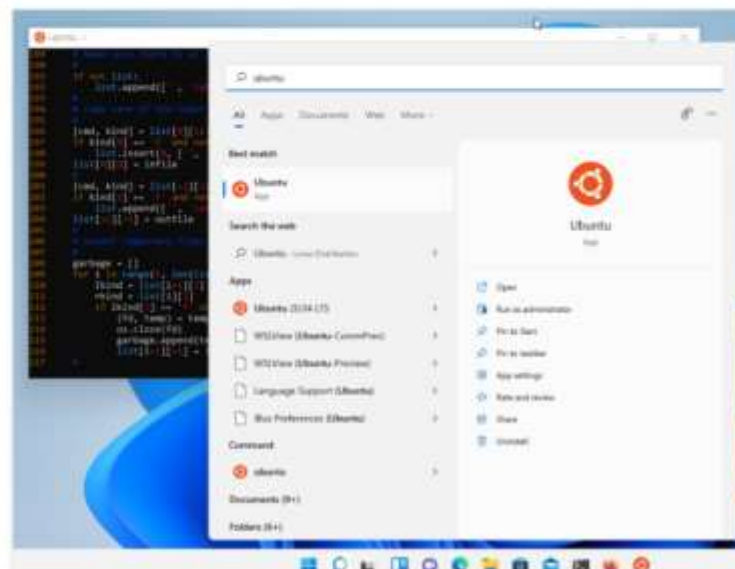
Ubuntu 22.04.2 LTS

Canonical Group Limited

★★★★☆ 18 | 開發人員工具

在 Store 應用程式中取得

免費



詳細資料

提供 1 種語言

發行者為

Canonical Group Limited

2021/10/25

Contact information

條款

隱私權原則

開發人員與 IT

App 徽章

端點管理員



一般

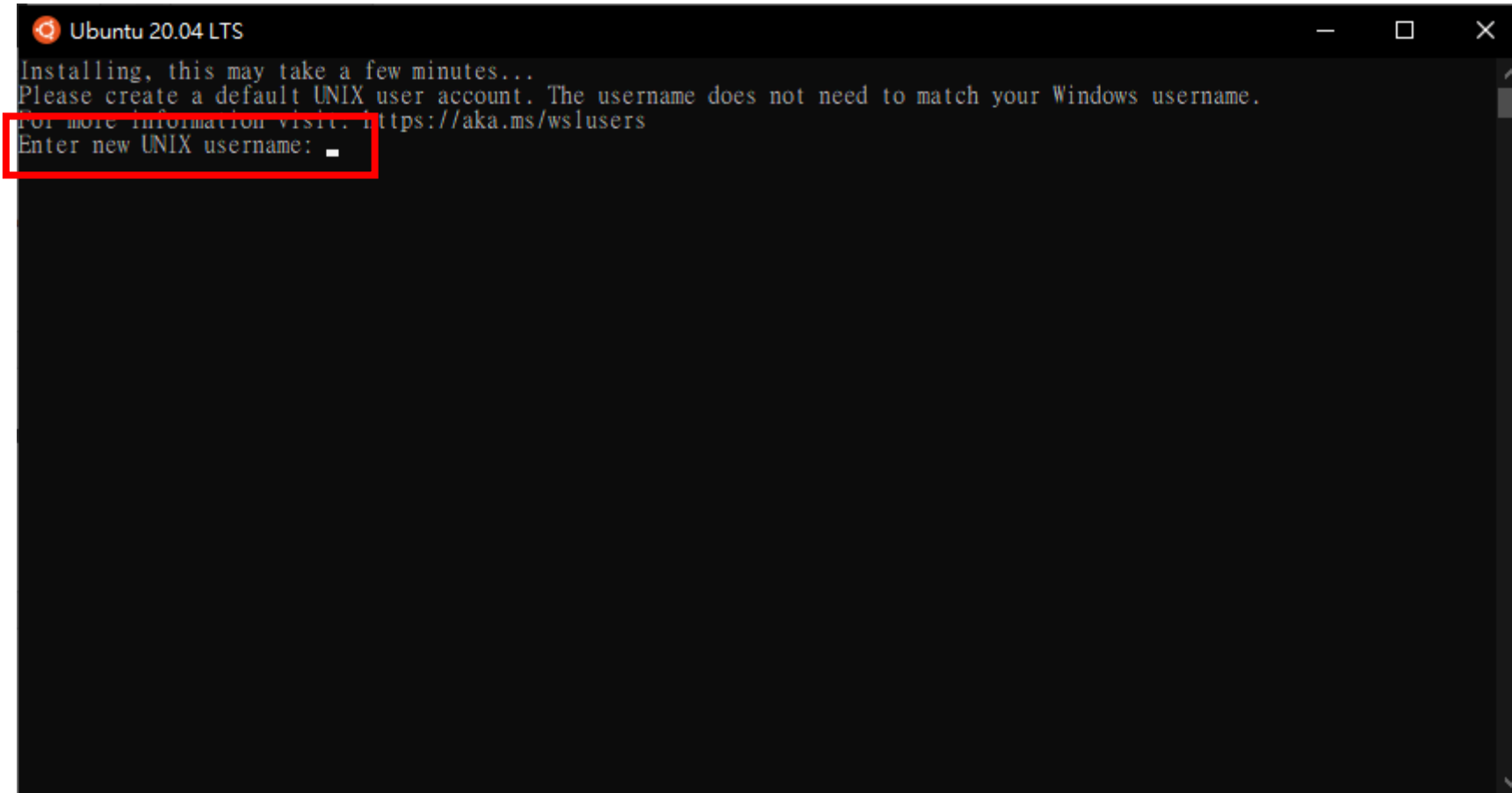
Outline

- Introduction
- Install WSL
- **Setup WSL**
- Install build-essential
- Compile in WSL
- Execute in WSL

Search for Ubuntu and start it

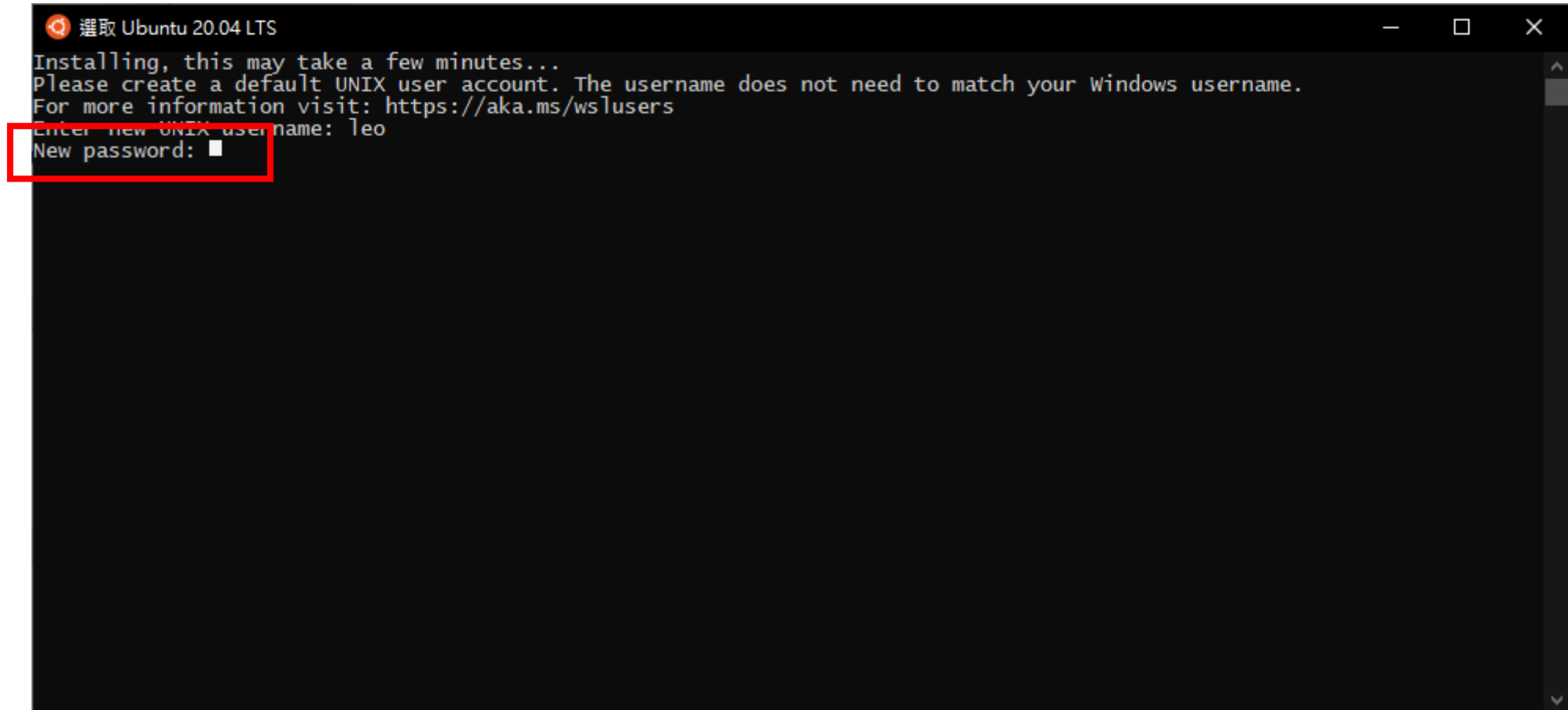


Enter your new username for Ubuntu

A screenshot of a terminal window titled "Ubuntu 20.04 LTS". The window has standard window controls (minimize, maximize, close) in the top right corner. The terminal text reads: "Installing, this may take a few minutes... Please create a default UNIX user account. The username does not need to match your Windows username. For more information visit: https://aka.ms/wslusers Enter new UNIX username: ". The prompt "Enter new UNIX username: " is highlighted with a red rectangular box. The cursor is positioned at the end of the prompt line.

```
Ubuntu 20.04 LTS
Installing, this may take a few minutes...
Please create a default UNIX user account. The username does not need to match your Windows username.
For more information visit: https://aka.ms/wslusers
Enter new UNIX username: 
```

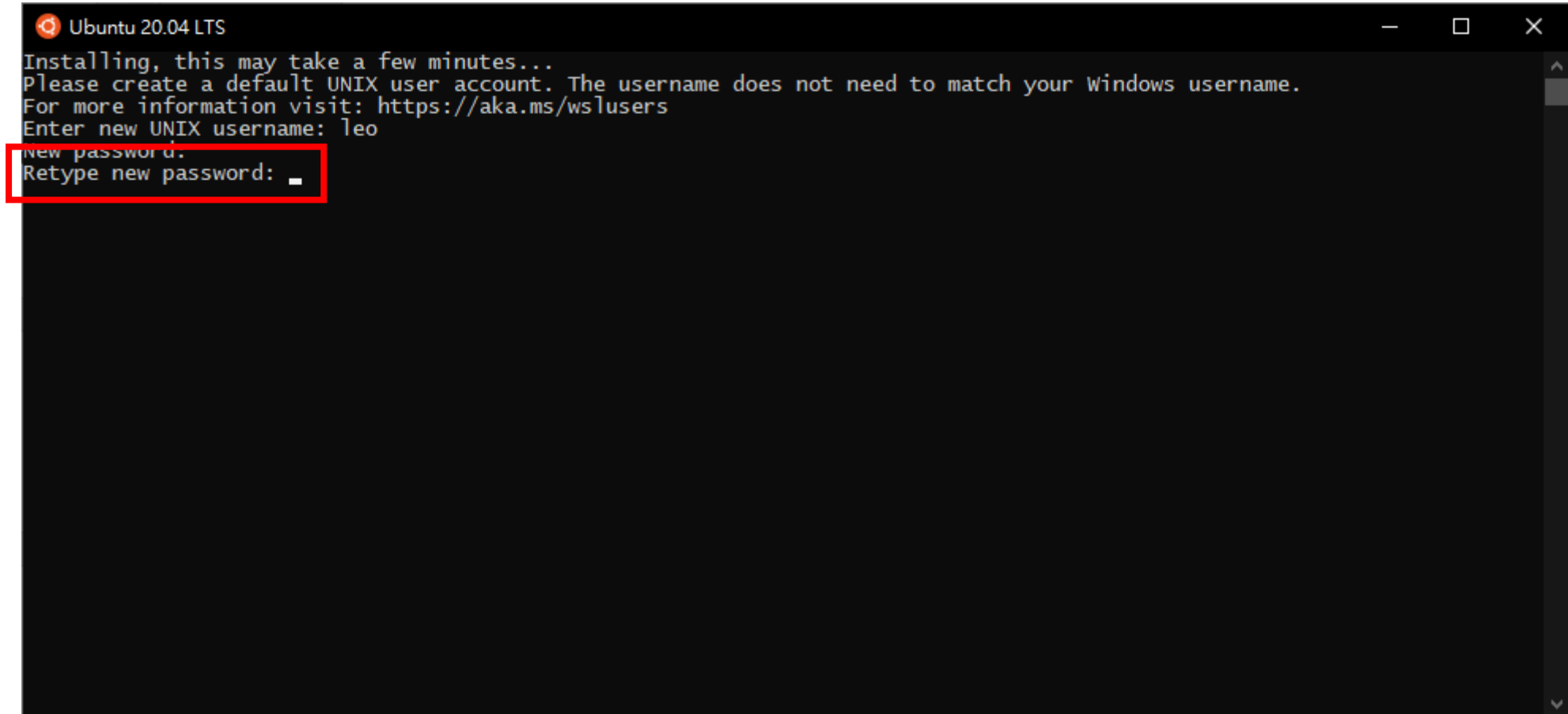
Enter your new password for Ubuntu



The screenshot shows a terminal window titled "選取 Ubuntu 20.04 LTS". The text inside the terminal reads: "Installing, this may take a few minutes... Please create a default UNIX user account. The username does not need to match your Windows username. For more information visit: <https://aka.ms/wslusers> Enter new UNIX username: leo New password: [mask]". A red rectangular box highlights the "New password:" prompt and its corresponding masked input field.

```
選取 Ubuntu 20.04 LTS
Installing, this may take a few minutes...
Please create a default UNIX user account. The username does not need to match your Windows username.
For more information visit: https://aka.ms/wslusers
Enter new UNIX username: leo
New password: [mask]
```


Validate your new password



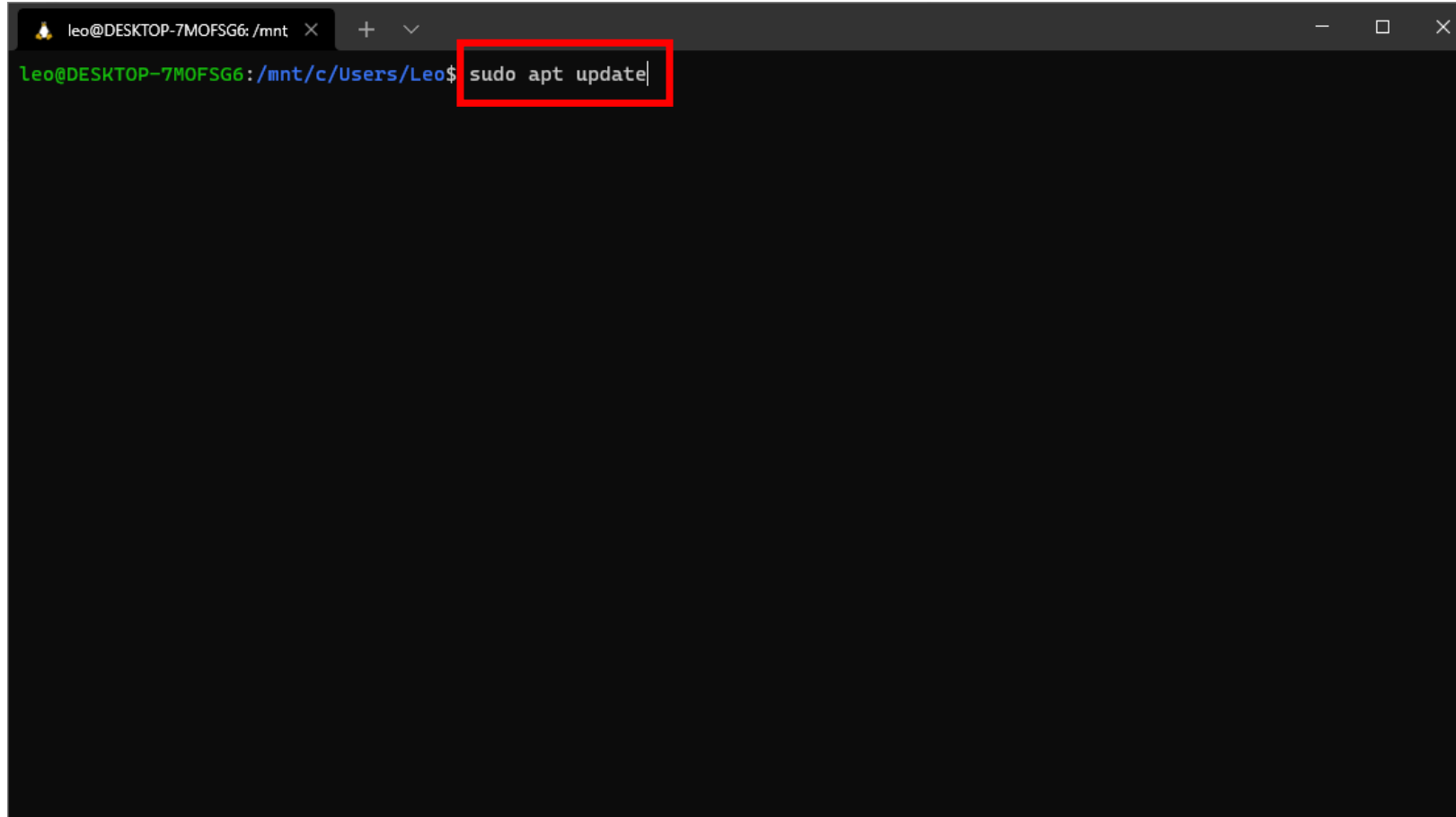
```
Ubuntu 20.04 LTS
Installing, this may take a few minutes...
Please create a default UNIX user account. The username does not need to match your Windows username.
For more information visit: https://aka.ms/wslusers
Enter new UNIX username: leo
new password:
Retype new password: _
```

The image shows a terminal window titled "Ubuntu 20.04 LTS" with standard window controls. The text inside the terminal indicates the installation of Ubuntu on Windows. It prompts the user to create a default UNIX user account, noting that the username does not need to match the Windows username. It provides a link for more information: <https://aka.ms/wslusers>. The user has entered "leo" as the new UNIX username. The terminal then prompts for a new password, and the user has entered a password (represented by a single character in the image). The prompt "Retype new password:" is shown, indicating the next step is to confirm the password.

Outline

- Introduction
- Install WSL
- Setup WSL
- **Install build-essential**
- Compile in WSL
- Execute in WSL

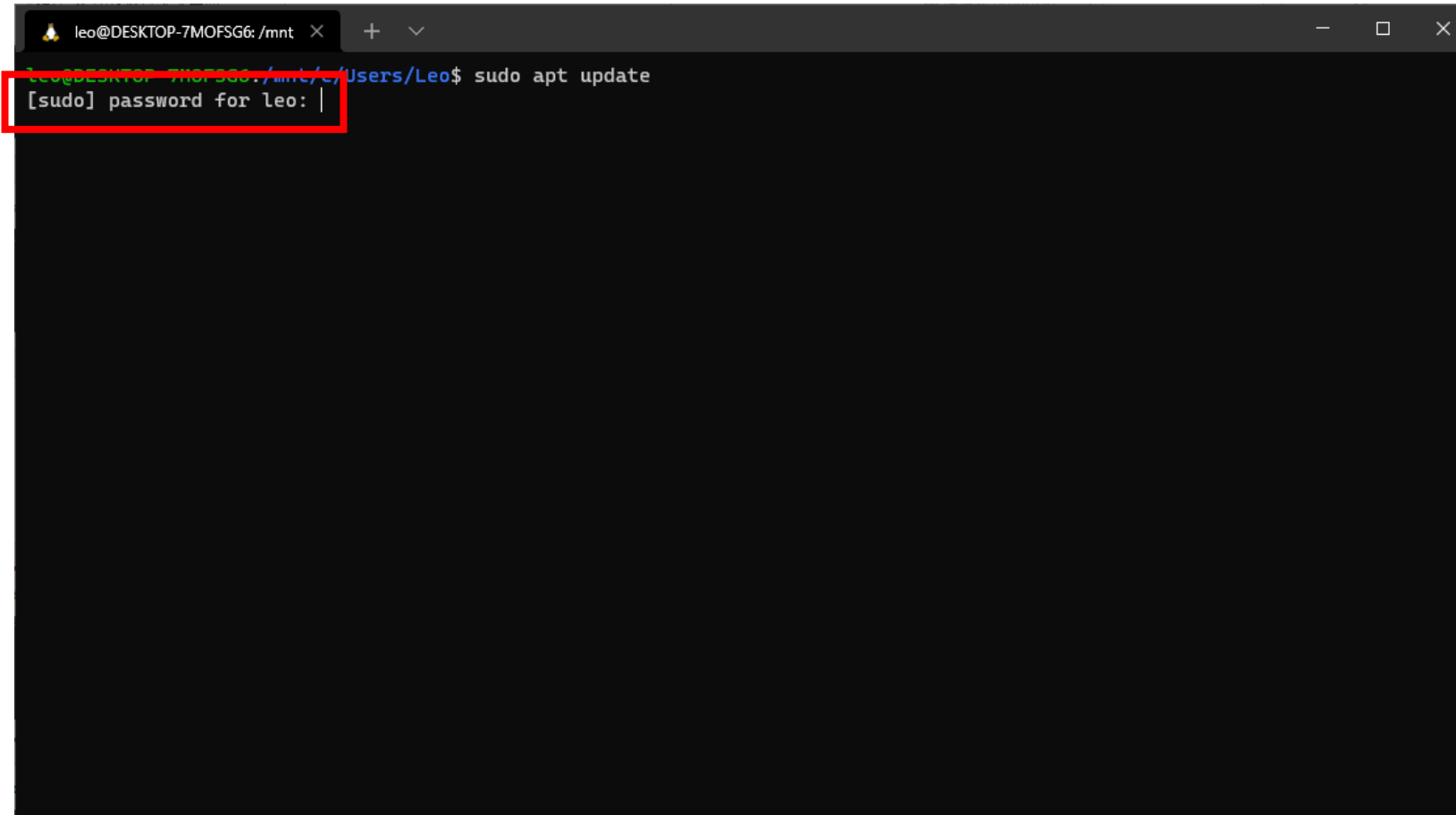
Enter “**sudo apt update**”



A terminal window with a dark background. The title bar shows 'leo@DESKTOP-7MOFSG6: /mnt' and standard window controls. The prompt is 'leo@DESKTOP-7MOFSG6:/mnt/c/Users/Leo\$'. The command 'sudo apt update' is entered and highlighted with a red rectangle.

```
leo@DESKTOP-7MOFSG6:/mnt/c/Users/Leo$ sudo apt update
```

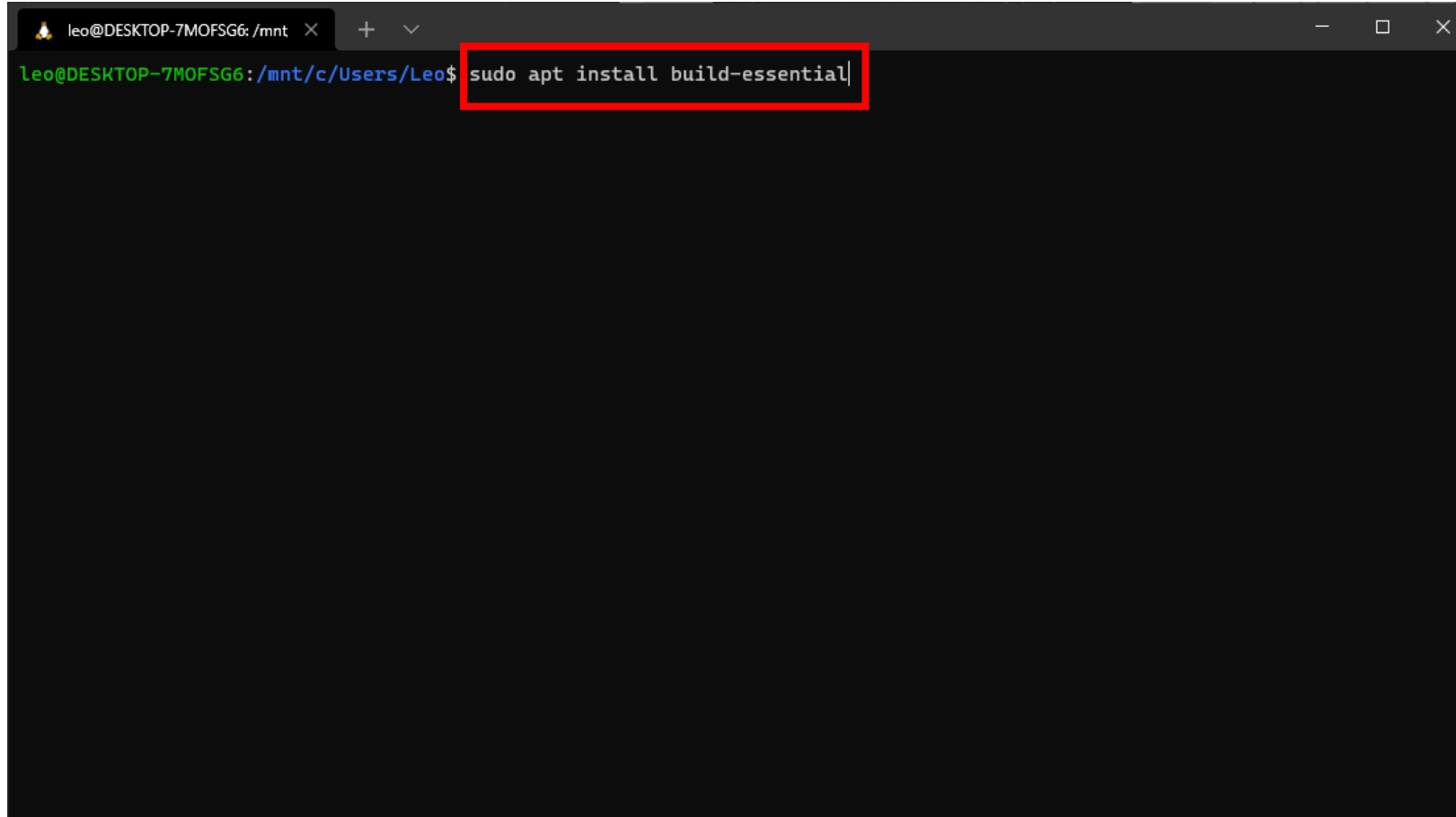
Enter your password for Ubuntu



A terminal window with a dark background. The title bar shows 'leo@DESKTOP-7MOFSG6: /mnt'. The prompt is 'leo@DESKTOP-7MOFSG6:/mnt/c/Users/Leo\$'. The command 'sudo apt update' has been entered. Below it, the prompt '[sudo] password for leo: |' is shown, with a red rectangle highlighting the text and the cursor. The rest of the terminal is empty.

```
leo@DESKTOP-7MOFSG6: /mnt
leo@DESKTOP-7MOFSG6:/mnt/c/Users/Leo$ sudo apt update
[sudo] password for leo: |
```

Enter “**sudo apt install build-essential**”

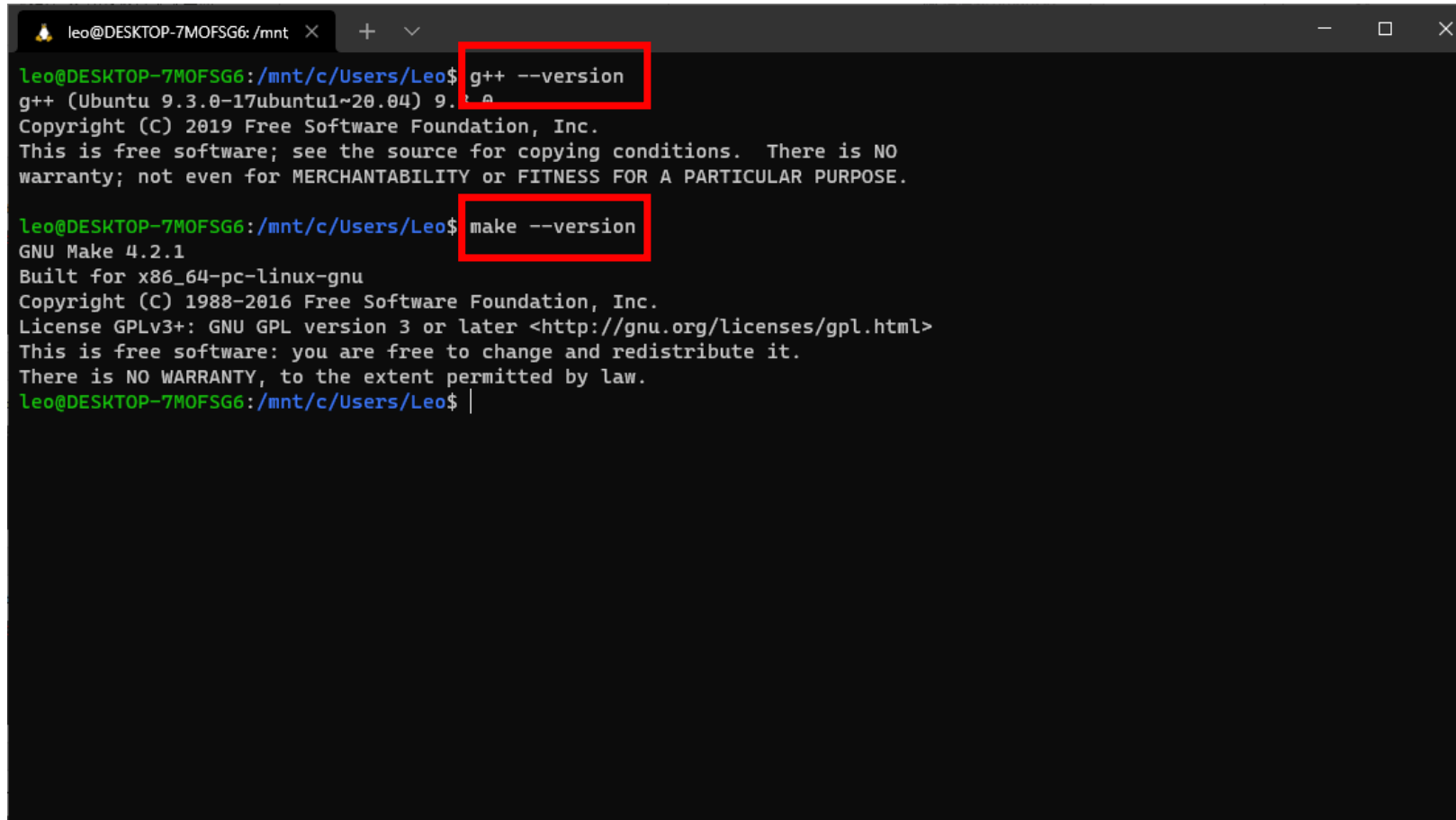
A terminal window with a dark background. The title bar shows a single tab with the text 'leo@DESKTOP-7MOFSG6: /mnt'. The terminal content shows the prompt 'leo@DESKTOP-7MOFSG6:/mnt/c/Users/Leo\$' followed by the command 'sudo apt install build-essential'. The command is enclosed in a red rectangular box. The cursor is at the end of the command.

```
leo@DESKTOP-7MOFSG6: /mnt
leo@DESKTOP-7MOFSG6:/mnt/c/Users/Leo$ sudo apt install build-essential
```

Press enter to continue

```
leo@DESKTOP-7MOFSG6: /mnt  ×  +  ▾  
leo@DESKTOP-7MOFSG6:/mnt/c/Users/Leo$ sudo apt install build-essential  
[sudo] password for leo:  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following additional packages will be installed:  
  binutils binutils-common binutils-x86-64-linux-gnu cpp cpp-9 dpkg-dev fakeroot g++ g++-9 gcc gcc-9 gcc-9-base  
  libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl libasan5 libatomic1 libbinutils  
  libc-dev-bin libc6-dev libcc1-0 libcrypt-dev libctf-nobfd0 libctf0 libdpkg-perl libfakeroot libfile-fcntllock-perl  
  libgcc-9-dev libgomp1 libisl22 libitm1 liblsan0 libmpc3 libquadmath0 libstdc++-9-dev libtsan0 libubsan1  
  linux-libc-dev make manpages-dev  
Suggested packages:  
  binutils-doc cpp-doc gcc-9-locales debian-keyring g++-multilib g++-9-multilib gcc-9-doc gcc-multilib autoconf  
  automake libtool flex bison gdb gcc-doc gcc-9-multilib glibc-doc bzip libstdc++-9-doc make-doc  
The following NEW packages will be installed:  
  binutils binutils-common binutils-x86-64-linux-gnu build-essential cpp cpp-9 dpkg-dev fakeroot g++ g++-9 gcc gcc-9  
  gcc-9-base libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl libasan5 libatomic1 libbinutils  
  libc-dev-bin libc6-dev libcc1-0 libcrypt-dev libctf-nobfd0 libctf0 libdpkg-perl libfakeroot libfile-fcntllock-perl  
  libgcc-9-dev libgomp1 libisl22 libitm1 liblsan0 libmpc3 libquadmath0 libstdc++-9-dev libtsan0 libubsan1  
  linux-libc-dev make manpages-dev  
0 upgraded, 41 newly installed, 0 to remove and 0 not upgraded.  
Need to get 40.0 MB of archives.  
After this operation, 175 MB of additional disk space will be used.  
Do you want to continue? [Y/n] |
```

Type “**g++ --version**” and “**make --version**”
to validate installation



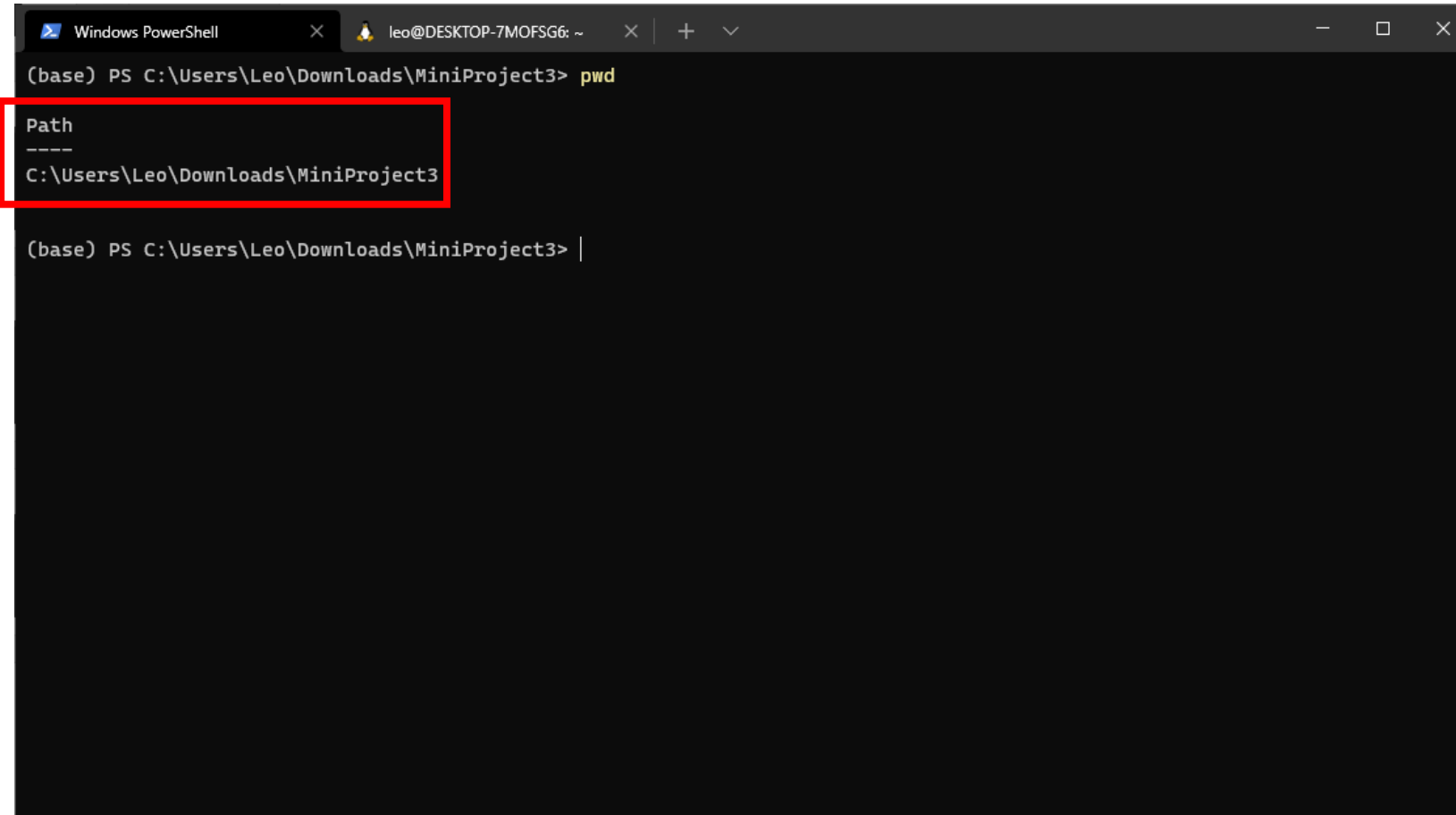
```
leo@DESKTOP-7MOFSG6: /mnt
leo@DESKTOP-7MOFSG6:/mnt/c/Users/Leo$ g++ --version
g++ (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0
Copyright (C) 2019 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

leo@DESKTOP-7MOFSG6:/mnt/c/Users/Leo$ make --version
GNU Make 4.2.1
Built for x86_64-pc-linux-gnu
Copyright (C) 1988-2016 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
leo@DESKTOP-7MOFSG6:/mnt/c/Users/Leo$
```

Outline

- Introduction
- Install WSL
- Setup WSL
- Install build-essential
- **Compile in WSL**
- Execute in WSL

In Windows, your project path in C drive would be like “**C:\[project_path]**”



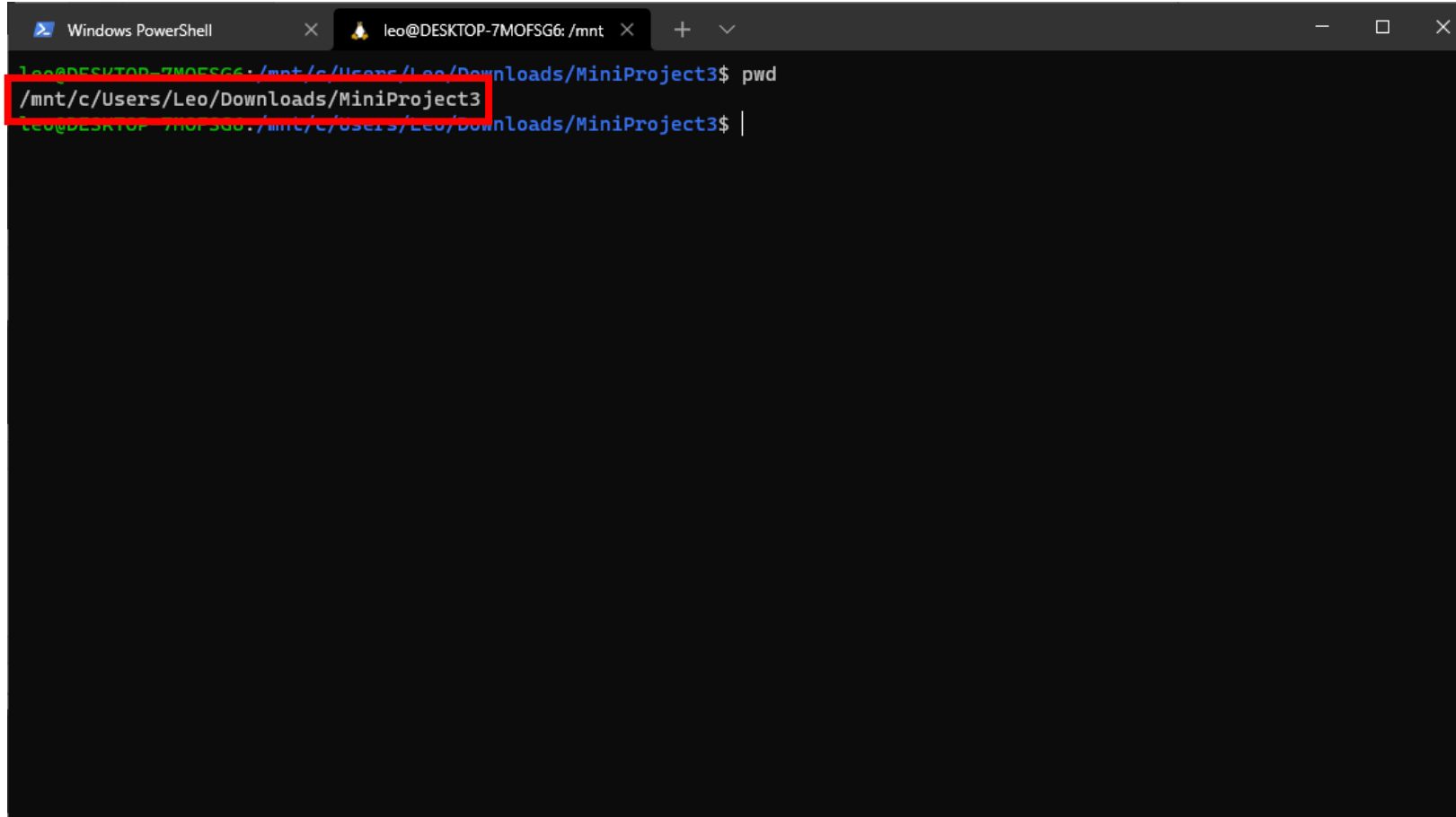
```
Windows PowerShell
leo@DESKTOP-7MOFSG6: ~
(base) PS C:\Users\Leo\Downloads\MiniProject3> pwd

Path
----
C:\Users\Leo\Downloads\MiniProject3

(base) PS C:\Users\Leo\Downloads\MiniProject3> |
```

The screenshot shows a Windows PowerShell terminal window. The title bar indicates the user is 'leo' on a desktop named 'DESKTOP-7MOFSG6'. The prompt is '(base) PS C:\Users\Leo\Downloads\MiniProject3>'. The user has entered the command 'pwd'. The output shows the current path as 'C:\Users\Leo\Downloads\MiniProject3'. The output is formatted with 'Path' and a separator '----'. The path is highlighted with a red box.

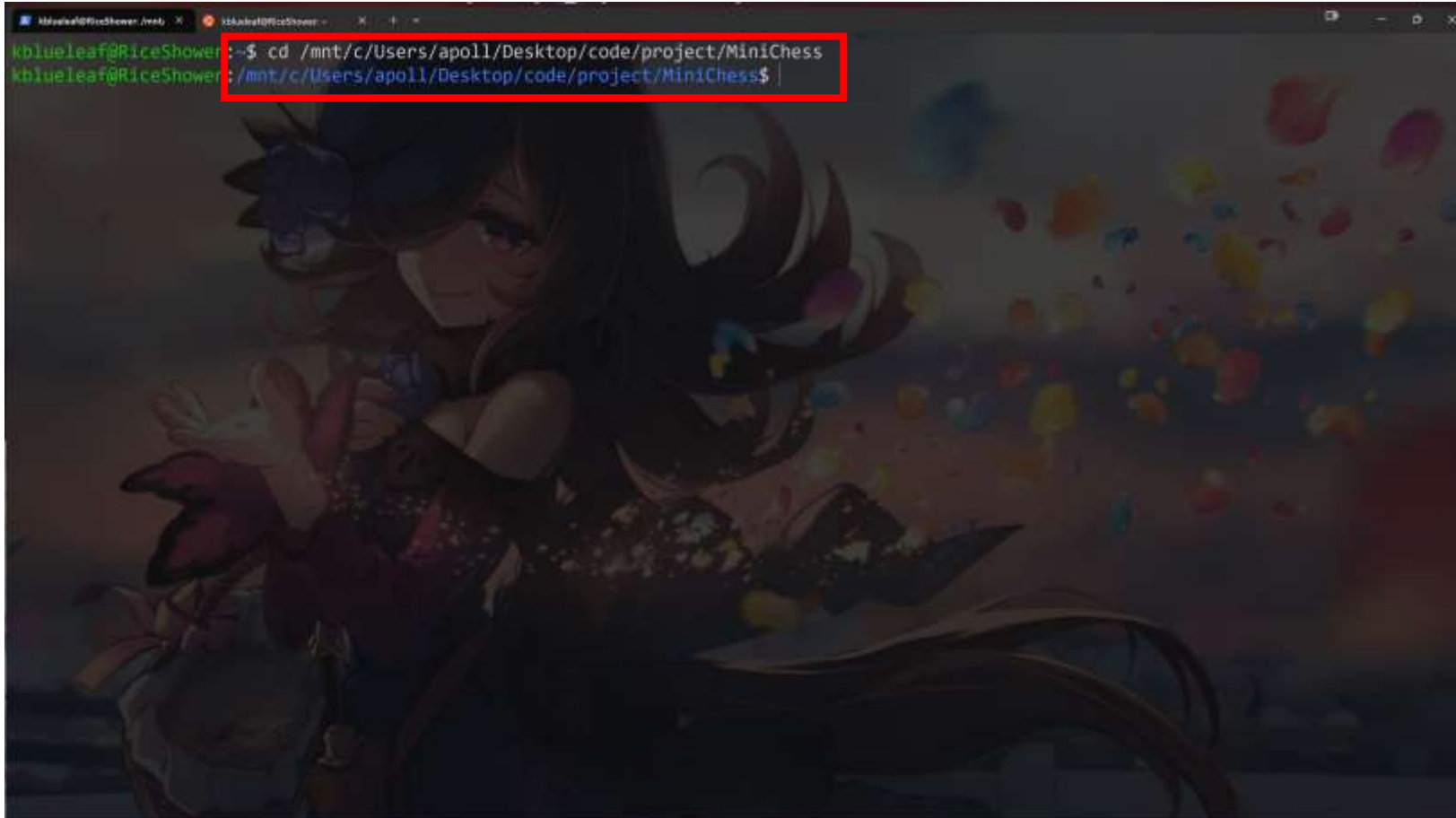
However, in WSL, your project path in **C drive** would be like **“/mnt/c/[project_path]”**



```
Windows PowerShell
leo@DESKTOP-7MOFSG6: /mnt
leo@DESKTOP-7MOFSG6: /mnt/c/Users/Leo/Downloads/MiniProject3$ pwd
/mnt/c/Users/Leo/Downloads/MiniProject3
leo@DESKTOP-7MOFSG6: /mnt/c/Users/Leo/Downloads/MiniProject3$ |
```

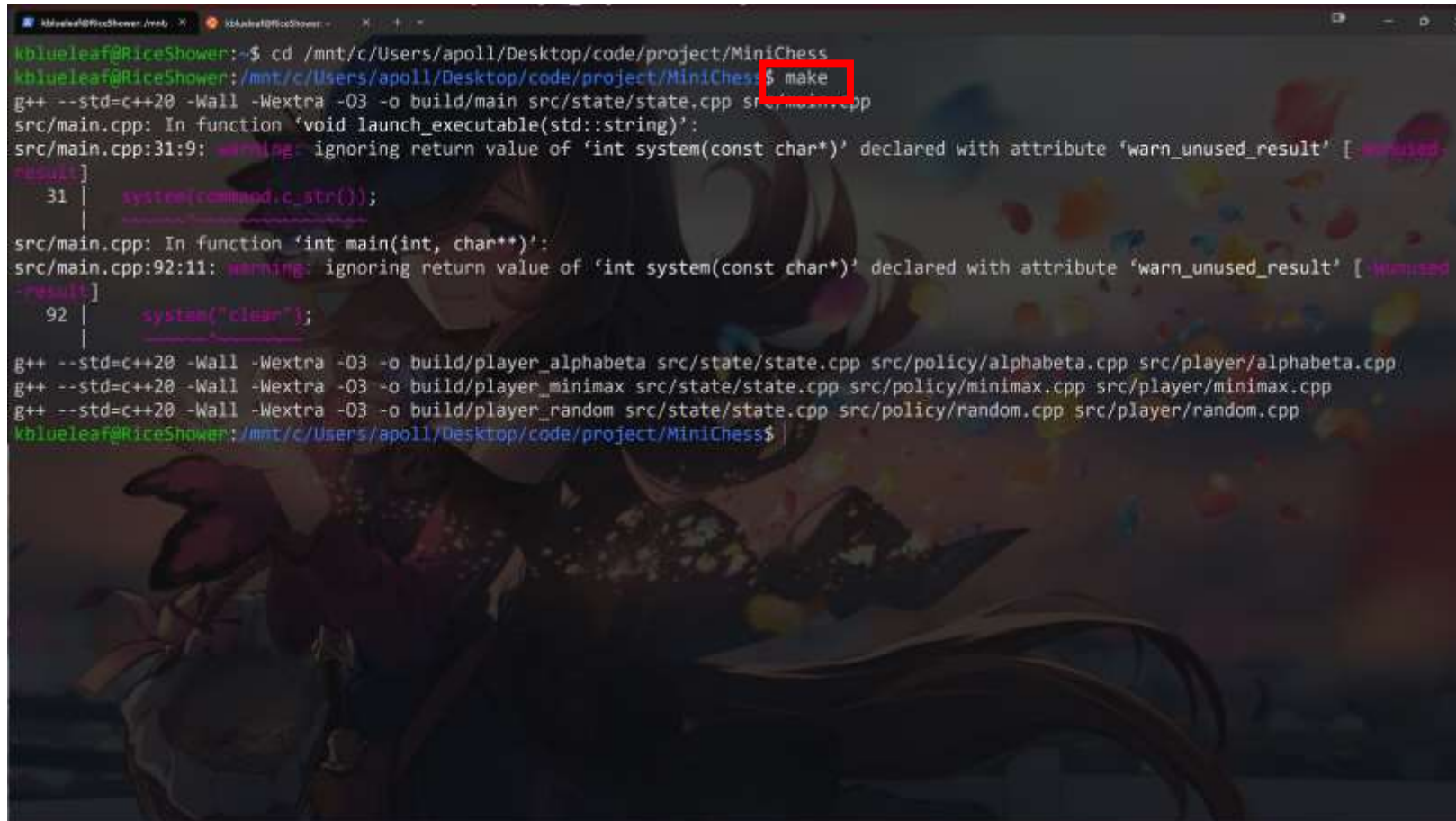
The screenshot shows a Windows PowerShell terminal window with two tabs. The active tab is titled 'leo@DESKTOP-7MOFSG6: /mnt'. The terminal content shows the user 'leo' at the prompt 'leo@DESKTOP-7MOFSG6:' in a directory '/mnt/c/Users/Leo/Downloads/MiniProject3'. The user has entered the command 'pwd', and the output is '/mnt/c/Users/Leo/Downloads/MiniProject3'. This output line is highlighted with a red rectangular box. The prompt for the next command is 'leo@DESKTOP-7MOFSG6: /mnt/c/Users/Leo/Downloads/MiniProject3\$ |'.

Change directory to the “**project**” folder of your project by “**cd [project_path]**”

A screenshot of a terminal window with a dark background and a colorful anime-style illustration of a character with long dark hair and a flower in her hair. The terminal shows a user named 'kblueleaf@RiceShower' at a prompt. The command being entered is 'cd /mnt/c/Users/apoll/Desktop/code/project/MiniChess', which is highlighted with a red rectangular box. The prompt changes from '~\$' to '/mnt/c/Users/apoll/Desktop/code/project/MiniChess\$' after the command is entered.

```
kblueleaf@RiceShower:~$ cd /mnt/c/Users/apoll/Desktop/code/project/MiniChess
kblueleaf@RiceShower:/mnt/c/Users/apoll/Desktop/code/project/MiniChess$
```

Type “**make**” to compile your code
warning for main.cpp can be ignored

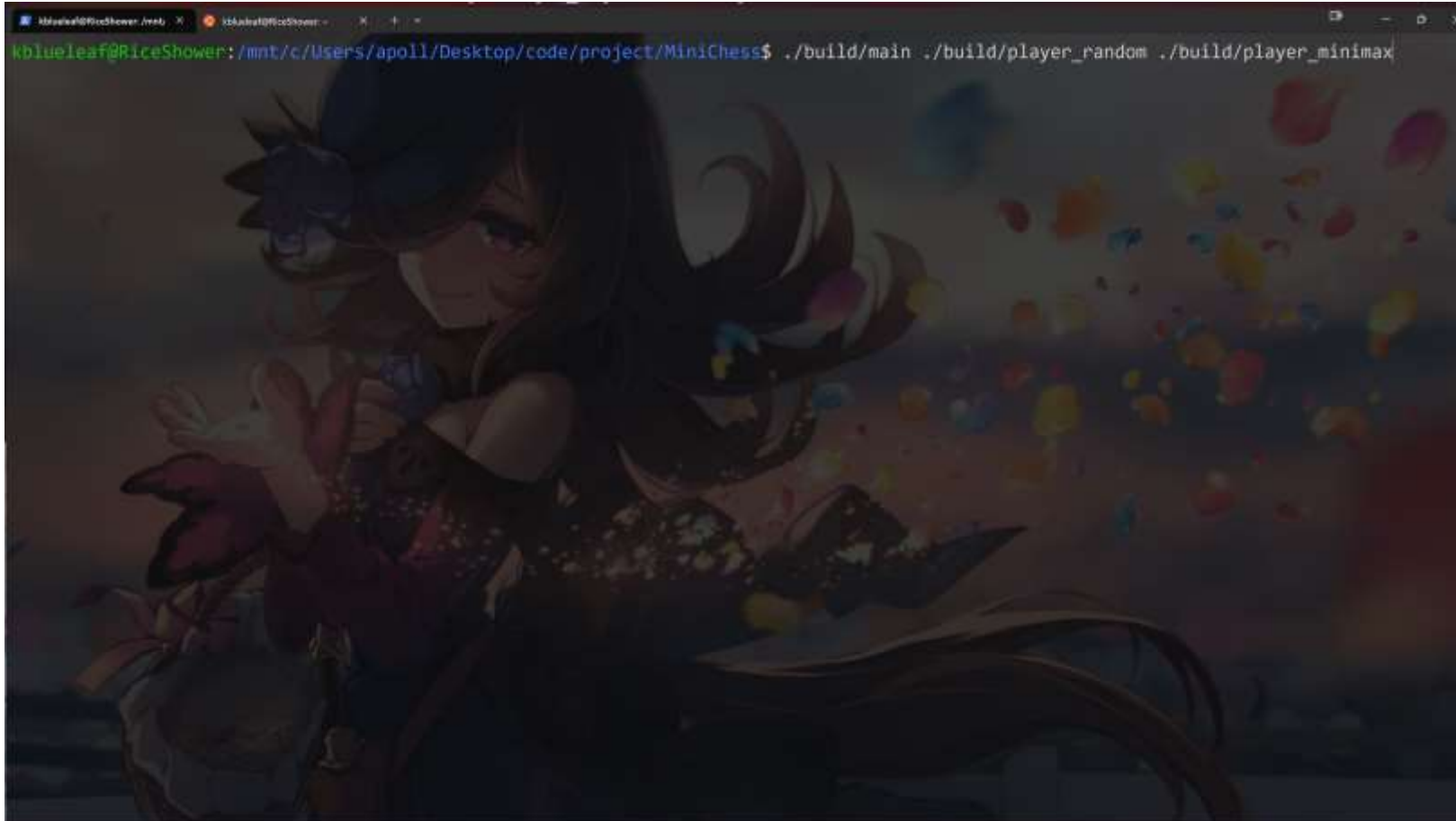
A terminal window with a dark background and a colorful anime-style illustration of a character with long hair and a headband. The terminal shows the execution of the 'make' command in a directory. The output includes the compilation of several C++ files using g++ with various flags. Two warnings are displayed, both related to ignoring the return value of the 'system' function, which are marked as 'unused-result'. The 'make' command is highlighted with a red box in the input line.

```
kblueleaf@RiceShower:~$ cd /mnt/c/Users/apoll/Desktop/code/project/MiniChess
kblueleaf@RiceShower:/mnt/c/Users/apoll/Desktop/code/project/MiniChess$ make
g++ --std=c++20 -Wall -Wextra -O3 -o build/main src/state/state.cpp src/main.cpp
src/main.cpp: In function 'void launch_executable(std::string)':
src/main.cpp:31:9: warning: ignoring return value of 'int system(const char*)' declared with attribute 'warn_unused_result' [-Wunused-result]
   31 |     system(command.c_str());
      |     ~~~~~^
src/main.cpp: In function 'int main(int, char**)':
src/main.cpp:92:11: warning: ignoring return value of 'int system(const char*)' declared with attribute 'warn_unused_result' [-Wunused-result]
   92 |     system("clear");
      |     ~~~~~^
g++ --std=c++20 -Wall -Wextra -O3 -o build/player_alphabeta src/state/state.cpp src/policy/alphabeta.cpp src/player/alphabeta.cpp
g++ --std=c++20 -Wall -Wextra -O3 -o build/player_minimax src/state/state.cpp src/policy/minimax.cpp src/player/minimax.cpp
g++ --std=c++20 -Wall -Wextra -O3 -o build/player_random src/state/state.cpp src/policy/random.cpp src/player/random.cpp
kblueleaf@RiceShower:/mnt/c/Users/apoll/Desktop/code/project/MiniChess$
```

Outline

- Introduction
- Install WSL
- Setup WSL
- Install build-essential
- Compile in WSL
- **Execute in WSL**

Type “**./build/main ./build/[AI1] ./build/[AI2]**” in terminal to execute while in the “[project_path]” folder of your project
You can also choose AI file from other folder (like baseline)



Happy Coding!