

# WSL - Windows Subsystem for Linux

HRM

August 21, 2025

## Table of contents

|   |          |
|---|----------|
| <b>Install WSL in Windows</b>                     | <b>1</b> |
| Remove Previous Installed Distributions . . . . . | 1        |
| Update WSL -> Ubuntu Installation . . . . .       | 1        |
| Install JavaScript Tools . . . . .                | 2        |
| Install Python Tools . . . . .                    | 2        |
| Install C/C++ Tools . . . . .                     | 2        |
| Install Java Tools . . . . .                      | 2        |
| Use NPX . . . . .                                 | 3        |

## Install WSL in Windows

We will use windows powershell to install wsl.

### Remove Previous Installed Distributions

```
wsl --list --verbose
```

For Each Listed distribution - `wsl --unregister <DistributionName>` - Open Settings → Apps → Installed apps, Find each Linux distribution, click the three-dot menu, and select Uninstall

### Update WSL -> Ubuntu Installation

```
wsl --update
```

```
wsl --list --online
```

```
wsl --install Ubuntu-24.04 -> Install Ubuntu
```

```
sudo apt update && sudo apt upgrade -y -> Update Ubuntu
```

```
PS C:\Users\hrith> wsl --install Ubuntu-24.04
Downloading: Ubuntu 24.04 LTS
Installing: Ubuntu 24.04 LTS
Distribution successfully installed. It can be launched via 'wsl.exe -d Ubuntu-24.04'
Launching Ubuntu-24.04...
Provisioning the new WSL instance Ubuntu-24.04
This might take a while...
Create a default Unix user account: hrm
New password:
Retype new password:
passwd: password updated successfully
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
```

```
hrm@bitnd:/mnt/c/Users/hrith$ sudo apt update && sudo apt upgrade -y
[sudo] password for hrm:
Hit:1 http://archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
```

## Install JavaScript Tools

- nvm install visit and run bash script <https://github.com/nvm-sh/nvm>
- node install <https://nodejs.org/en/download>
  - nvm install 22
  - nvm list nvm use 22 nvm current
  - corepack enable yarn
  - corepack enable pnpm > Check
- nvm -v
- node -v
- npm -v npx -v
- pnpm -v
- yarn -v

## Install Python Tools

- 1) Python VENV
  - `python3 -m venv .venv -> Copy error code and run sudo apt install python3.12-venv`
- 2) PIPX
  - `sudo apt install pipx`
- 3) UV - Rust-based Python package installer
  - `pipx install uv` It will maintain isolation
- 4) LLM
  - `pipx install llm -> pipx ensurepath`
  - Configure it
    - `llm install llm-gemini` or `llm install llm-ollama`
    - `llm keys set gemini`
    - `llm -m gemini-2.0-flash 'Tell me fun facts about Mountain View'`
- 5) MiniConda
  - Download `.sh` <https://www.anaconda.com/download/success>
  - `bash <pathto .sh file>`
  - `conda config --set auto_activate_base false`

## Install C/C++ Tools

`sudo apt install build-essential - gcc -> The C compiler - g++ -> The C++ compiler`  
`Check - gcc -version - g++ -version`

## Install Java Tools

`sudo apt install default-jdk`

This command installs:

Java Development Kit (JDK) - Compiler, debugger, and development tools  
Java Runtime Environment (JRE) - Required to run Java applications  
Java Virtual Machine (JVM) - Core execution environment

- Configure `JAVA_HOME` Environment Variable

- echo 'export JAVA\_HOME="/usr/lib/jvm/default-java"' >> ~/.bashrc ## confirm the path first using below update-alternative... command
    - restart shell
  - Install other versions of java
    - sudo apt install openjdk-17-jdk
  - Set Default Java/Javac installed version
    - sudo update-alternatives --config java
    - sudo update-alternatives --config javac
- Check
- ```
- java --version - javac --version - echo $JAVA_HOME
```

## Use NPX

- npm install -g promptfoo then npx promptfoo view