

WSL (Windows Subsystem for Linux) Installation & Usage Guide (Ultimate Edition)

What is WSL?

WSL (Windows Subsystem for Linux) allows you to run a full Linux environment (e.g., Ubuntu, Debian) directly on Windows without needing a separate dual-boot setup or heavy virtual machines. It's ideal for developers who want Linux tools while staying in Windows.

Step-by-Step: How to Install WSL

① Prerequisites

- Windows 10 version **2004** (build 19041) or higher **OR** any version of Windows 11
- **Virtualization enabled** in BIOS:
 - Reboot and press **F2**, **Del**, or **Esc** to enter BIOS
 - Look for **Intel VT-x / AMD-V / Virtualization Technology** → Enable it

② Open PowerShell as Administrator

- Press **Win + S** → Type **"PowerShell"**
- Right-click **Windows PowerShell** → **Run as administrator**

③ Automatic Install (Recommended)

```
wsl --install
```

This will:


- Enable required Windows features
- Download and install **WSL 2**
- Install **Ubuntu** by default

 **Reboot when prompted** to finish setup.

④ Manual Installation (If `wsl --install` Fails or You Want Control)

Enable necessary Windows features manually:

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

Then install WSL kernel manually from:  <https://aka.ms/wsl2kernel>

Set WSL 2 as default:

```
wsl --set-default-version 2
```

Install Ubuntu (or any distro):

```
wsl --install -d Ubuntu
```

How to Check WSL Status

```
wsl --status
```

Shows:


- Default version (1 or 2)
 - Kernel version
 - Installed distributions
-

How to Open WSL

Option 1: Using Start Menu

- Press Win
- Search for **Ubuntu** or installed distro
- Press Enter

Option 2: Using Windows Terminal

- Open **Windows Terminal**
- Click  icon → Select Ubuntu (or any WSL distro)
- Shortcut: Ctrl + Shift + 3 (if mapped)

Option 3: From CMD/PowerShell

```
wsl
```

or

```
wsl -d Ubuntu
```

How to Close/Exit WSL

Exit WSL Session

Inside WSL terminal:

```
exit
```

Shutdown All WSL Instances

From CMD or PowerShell:

```
wsl --shutdown
```

List Running WSL Instances

```
wsl --list --running
```

Common Commands

- List installed distros:

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

- Install a specific distro:

```
wsl --install -d Debian
```

- Set default distro:

```
wsl --set-default <DistroName>
```

- Access Linux files from Windows:

```
\\wsl$\Ubuntu\home\your-username
```

- Access Windows files from WSL:

```
cd /mnt/c/Users
```

Common WSL Installation Errors & Fixes


 **Error:** `Wsl/InstallDistro/Service/RegisterDistro/CreateVm/HCS/HCS_E_SERVICE_NOT_AVAILABLE`

Fix: Enable features manually:

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

Also ensure:


- Virtualization is enabled in BIOS
- Reboot system after enabling features

 **Error:** `snap: command not found` **inside WSL**

Fix: WSL doesn't support `snapd` by default. Use `apt` instead:

```
sudo apt install <package-name>
```

Or enable systemd support manually (advanced setup).

 **Error: Command points to** `/snap/bin/...` **even after uninstall**

Fix:

```
hash -r      # Reset command cache in Bash
```

 **Ubuntu won't open or install**

Fix: Try resetting the distro:

```
wsl --unregister Ubuntu
wsl --install -d Ubuntu
```

Pro Tips

- Keep system updated:

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

- Install useful tools:

```
sudo apt install git curl tree build-essential
```

- Use `Windows Terminal` for the best WSL experience
- Use `code .` inside WSL to open projects with VS Code (with WSL extension)

Let me know if you'd like a version with screenshots, advanced features (like Docker, GUI apps), or auto-fix scripts!