

WSL ASSIGNMENT

Download Linux

Step 1: To Display available list use command “wsl –list -online”. Instal the default ubuntu 20.04 distro.

Step 2: Install a specific distro by name, such as Debian.

```
root@8b5c7dd85f01583: ~  
NAME                                FRIENDLY NAME  
Ubuntu                              Ubuntu  
Debian                              Debian GNU/Linux  
kali-linux                          Kali Linux Rolling  
Ubuntu-18.04                        Ubuntu 18.04 LTS  
Ubuntu-20.04                        Ubuntu 20.04 LTS  
Ubuntu-22.04                        Ubuntu 22.04 LTS  
Ubuntu-24.04                        Ubuntu 24.04 LTS  
OracleLinux_7.9                     Oracle Linux 7.9  
OracleLinux_8.7                     Oracle Linux 8.7  
OracleLinux_9.1                     Oracle Linux 9.1  
openSUSE-leap-15.6                  openSUSE Leap 15.6  
SUSE-Linux-Enterprise-15-SP5        SUSE Linux Enterprise 15 SP5  
SUSE-Linux-Enterprise-15-SP6        SUSE Linux Enterprise 15 SP6  
openSUSE-Tumbleweed                 openSUSE Tumbleweed  
PS C:\Users\Administrator> wsl --install  
Ubuntu is already installed.  
Launching Ubuntu...  
root@8b5c7dd85f01583:~# wsl --install -d Debian
```

Launch & Update Linux

Step 1: “sudo apt update” is used to update the package list.

```
root@8b5c7dd85f01583:~# sudo apt update  
Get:1 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]  
Hit:2 http://archive.ubuntu.com/ubuntu noble InRelease  
Get:3 http://archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]  
Get:4 http://security.ubuntu.com/ubuntu noble-security/main amd64 Components [8972 B]  
Get:5 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Components [52.0 kB]  
Get:6 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Components [208 B]  
Get:7 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Components [208 B]  
Get:8 http://archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]  
Get:9 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [838 kB]  
Get:10 http://archive.ubuntu.com/ubuntu noble-updates/main Translation-en [191 kB]  
Get:11 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 Components [15.1 kB]  
Get:12 http://archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [1003 kB]  
Get:13 http://archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [250 kB]  
Get:14 http://archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [315 kB]  
Get:15 http://archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Components [212 B]  
Get:16 http://archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Components [948 B]  
Get:17 http://archive.ubuntu.com/ubuntu noble-backports/main amd64 Components [208 B]  
Get:18 http://archive.ubuntu.com/ubuntu noble-backports/universe amd64 Components [17.7 kB]  
Get:19 http://archive.ubuntu.com/ubuntu noble-backports/restricted amd64 Components [212 B]  
Get:20 http://archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 Components [212 B]  
Fetched 3207 kB in 5s (618 kB/s)  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
19 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

Step 2: “sudo apt upgrade” is used to upgrade the installed packages.

```
19 packages can be upgraded. Run 'apt list --upgradable' to see them.  
root@8b5c7dd85f01583:~# sudo apt upgrade  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
Calculating upgrade... Done  
The following package was automatically installed and is no longer required:  
  libllvm17t64  
Use 'sudo apt autoremove' to remove it.  
The following NEW packages will be installed:  
  libllvm19 mesa-libgallium  
The following upgrades have been deferred due to phasing:  
  libunwind8 wsl-setup  
The following packages will be upgraded:  
  libegl-mesa0 libgbm1 libgl1-mesa-dri libglapi-mesa libglx-mesa0 libnss-systemd libpam-systemd libsystemd-shared  
  libsystemd0 libudev1 mesa-vulkan-drivers systemd systemd-dev systemd-resolved systemd-sysv systemd-timesyncd udev
```

Step 3: Install the WSL Kernel .

```
PS C:\Users\Administrator> wsl --install  
Ubuntu is already installed.  
Launching Ubuntu...
```

Step 4: Update the WSL Kernel .

```
PS C:\Users\Administrator> wsl --update  
Checking for updates.  
The most recent version of Windows Subsystem for Linux is already installed.
```

Step 5: Lists all installed WSL distributions with their status and version details.

```
PS C:\Users\Administrator> wsl --list --verbose  
NAME                STATE              VERSION  
* docker-desktop    Stopped            2  
Ubuntu              Stopped            2  
PS C:\Users\Administrator>
```

Switch Between WSL1 and WSL2

```
PS C:\Users\Administrator> wsl --set-version Ubuntu 1
Conversion in progress, this may take a few minutes.
The distribution is already the requested version.
Error code: Wsl\Service\WSL_E_VH_MODE_INVALID_STATE
PS C:\Users\Administrator> wsl --set-version Ubuntu 2
For information on key differences with WSL 2 please visit https://aka.ms/wsl2
Conversion in progress, this may take a few minutes.
The operation completed successfully.
```

Set a Default Linux Distribution

Step 1: It shows all the installed WSL distributions

```
PS C:\Users\Administrator> wsl --list
Windows Subsystem for Linux Distributions:
docker-desktop (Default)
Ubuntu
```

Step 2: Sets the specified WSL distribution as the default for future wsl commands.

```
PS C:\Users\Administrator> wsl --setdefault Ubuntu
The operation completed successfully.
```

Run Linux as a Specific User

Step 1: Run the WSL distribution with specified user “pradhi”.

```
PS C:\Users\Administrator> wsl --user pradhi
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 5.15.167.4-microsoft-standard-WSL2 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Tue Feb  4 09:22:00 UTC 2025

System load:  0.09          Processes:            53
Usage of /:   0.2% of 1006.85GB Users logged in:        0
Memory usage: 4%           IPv4 address for eth0: 172.28.20.19
Swap usage:   0%           IPv4 address for eth0: 192.168.1.100

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.
   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

This message is shown once a day. To disable it please create the
/home/pradhi/.hushlogin file.
pradhi@88b5c7d985f01583: ~$
```

Step 2: Creates a new directory named “backup” on the D: drive

```
PS C:\Users\Administrator> mkdir D:\backup

Directory: D:\

Mode                LastWriteTime         Length Name
----                -
d-----          04-02-2025         15:22         backup
```

Step 3: Exports the Ubuntu WSL distribution to a tar file named ubuntu.tar and saves it to the D:\backup directory.

```
PS C:\Users\Administrator> wsl --export Ubuntu D:\backup\ubuntu.tar
Export in progress, this may take a few minutes.
The operation completed successfully.
```

Step 4: The command wsl --unregister ubuntu removes the Ubuntu WSL distribution and deletes its data from the system.

```
PS C:\Users\Administrator> wsl --unregister Ubuntu
Unregistering.
The operation completed successfully.
```

Step 5: Creates a new directory named "wsl" on the D: drive

```
PS C:\Users\Administrator> mkdir D:\wsl

Directory: D:\

Mode                LastWriteTime         Length Name
----                -
d-----          04-02-2025         15:10         wsl
```

Step 6: Imports the Ubuntu WSL distribution from the ubuntu.tar file located in the D:\backup directory and installs it to the D:\wsl directory.

```
PS C:\Users\Administrator> wsl --import Ubuntu D:\wsl\ D:\backup\ubuntu.tar
Import in progress, this may take a few minutes.
The operation completed successfully.
PS C:\Users\Administrator>
```

Step 7: Sets the default user for the Ubuntu WSL distribution to <yourname>.

```
PS C:\Users\Administrator> ubuntu config --default-user pradhi
```

Step 8: The command `sudo vi /etc/wsl.conf` opens the WSL configuration file for editing.

```
pradhi@8b5c7dd85f01583:~$ sudo vi /etc/wsl.conf
[sudo] password for pradhi:
pradhi@8b5c7dd85f01583:~$
```

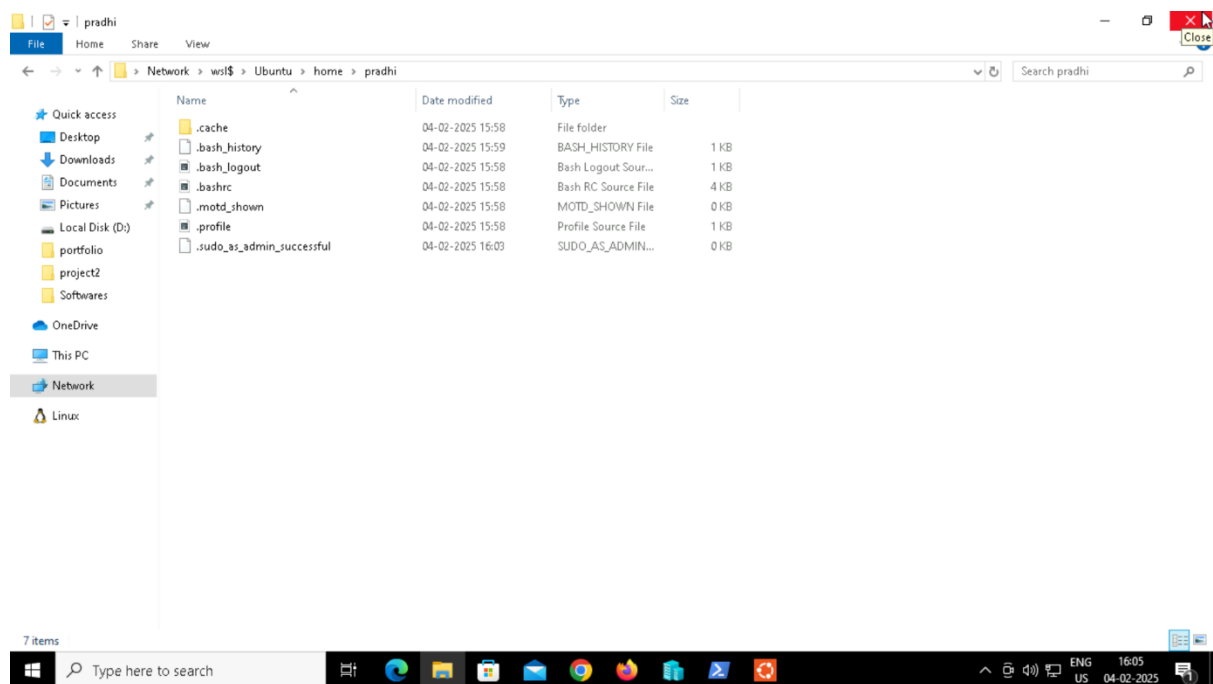
Edit file:



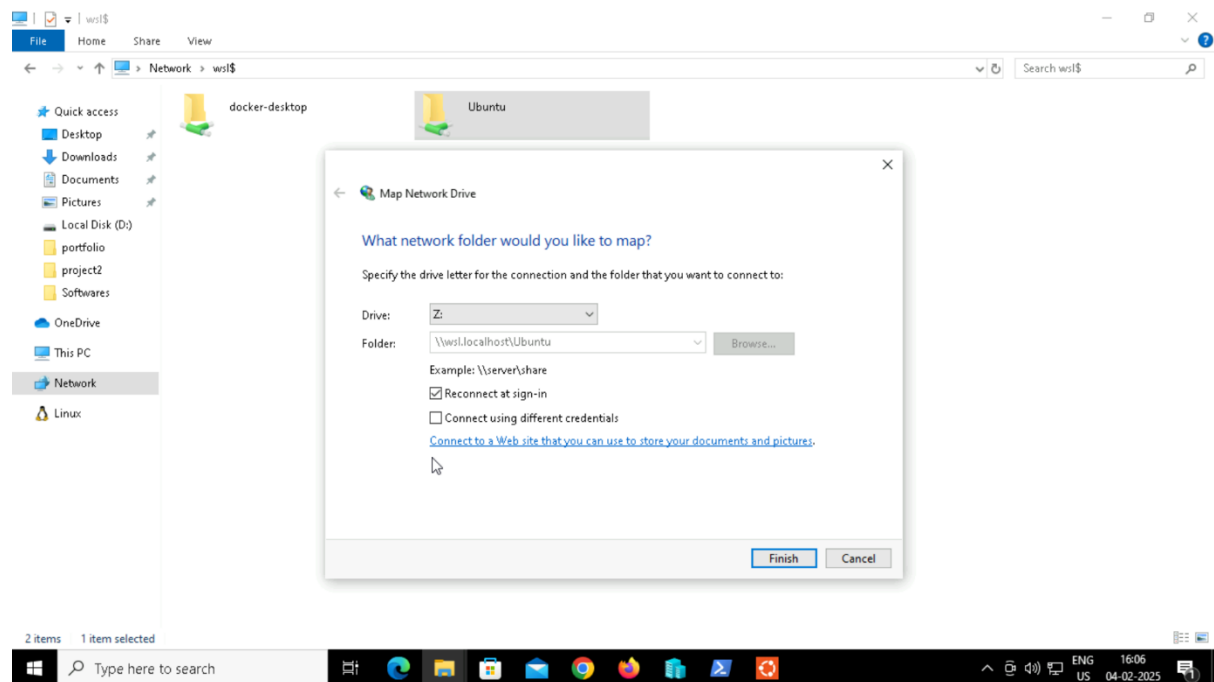
Step 9: Stops the running Ubuntu WSL distribution

```
PS C:\Users\Administrator> ubuntu config --default-user pradi
PS C:\Users\Administrator> wsl --terminate Ubuntu
The operation completed successfully.
PS C:\Users\Administrator>
```

Access Linux Files from Windows



Map a Network Drive



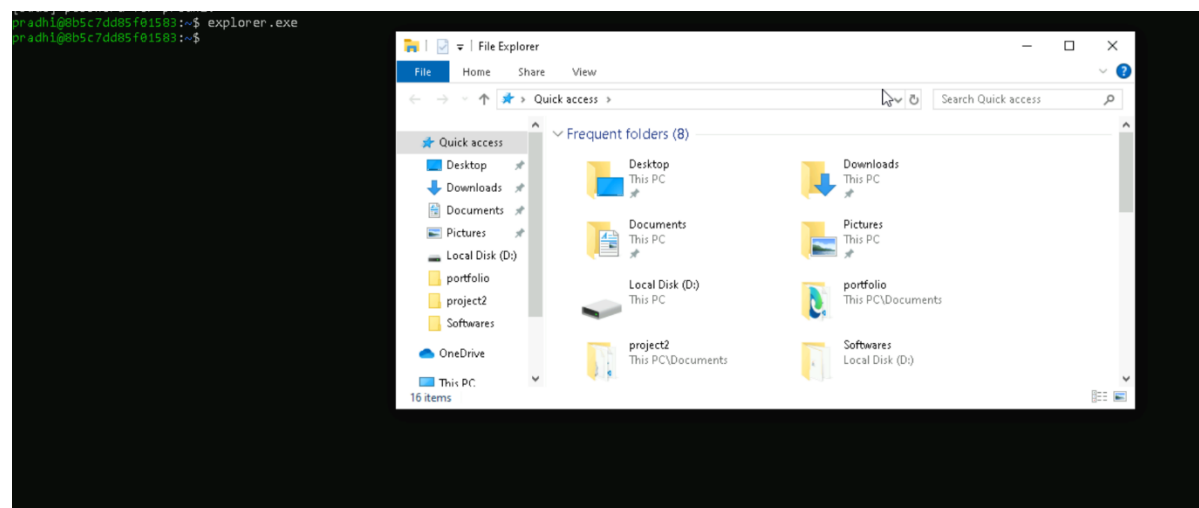
Accessing Windows Files from Linux

```
pradhig@b5c7dd85f01583:~$ cd /mnt/c/Users
pradhig@b5c7dd85f01583:~$
```

Run Linux Commands from Windows

```
PS C:\Users\Administrator> wsl pwd
/mnt/host/c/Users/Administrator
PS C:\Users\Administrator>
```

Run Windows Applications from Linux



Install Applications

Step 1: It ensures Git converts CRLF to LF on commit.

```
ms-adm1085c7d08f61583:~$ git config --global core.autocrlf input
ms-adm1085c7d08f61583:~$ sudo apt update
Hit:1 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Hit:2 http://security.ubuntu.com/ubuntu noble InRelease
Get:3 http://archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [615 kB]
Get:5 http://security.ubuntu.com/ubuntu noble-security/main Translation-en [118 kB]
Get:6 http://archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:7 http://archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:8 http://security.ubuntu.com/ubuntu noble-security/main amd64 Components [8972 B]
Get:9 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [803 kB]
Get:10 http://security.ubuntu.com/ubuntu noble-security/universe Translation-en [171 kB]
Get:11 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Components [52.0 kB]
Get:12 http://security.ubuntu.com/ubuntu noble-security/universe amd64 c-n-f Metadata [13.5 kB]
Get:13 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Packages [620 kB]
Get:14 http://security.ubuntu.com/ubuntu noble-security/restricted Translation-en [119 kB]
Get:15 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Components [208 B]
Get:16 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Packages [12.4 kB]
Get:17 http://security.ubuntu.com/ubuntu noble-security/multiverse Translation-en [2940 B]
Get:18 http://archive.ubuntu.com/ubuntu noble/universe Translation-en [5902 kB]
Get:19 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Components [208 B]
Get:20 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 c-n-f Metadata [356 B]
Get:21 http://archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:22 http://archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:23 http://archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:24 http://archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
Get:25 http://archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]
Get:26 http://archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 B]
Get:27 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [838 kB]
Get:28 http://archive.ubuntu.com/ubuntu noble-updates/main Translation-en [191 kB]
Get:29 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 Components [151 kB]
Get:30 http://archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [1003 kB]
Get:31 http://archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [250 kB]
Get:32 http://archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [315 kB]
Get:33 http://archive.ubuntu.com/ubuntu noble-updates/universe amd64 c-n-f Metadata [19.9 kB]
```

Step 2: This command installs essential packages for building software on Ubuntu, including compilers and libraries.

```

gradh@b05c7dd85f01503:~$ sudo apt-get install build-essential
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  bzip2 cpp cpp-13 cpp-13-x86-64-linux-gnu cpp-x86-64-linux-gnu dpkg-dev fakeroot g++ g++-13 g++-13-x86-64-linux-gnu gcc gcc-13
  gcc-13-base gcc-13-x86-64-linux-gnu gcc-x86-64-linux-gnu libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl libbam3 libasan8
  libatomic1 libc-dev-bin libc6-dev libc6-dev libc6-i386 libcc1-0 libcrypt-dev libdeb265-0 libdpkg-perl libfakeroot libfile-fcntllock-perl libgcc-13-dev libgdb3
  libgomp1 libhelf-plugin-asmdev libhelf-plugin-asmdev libhelf-plugin-libdeb265 libhelf1 libhwasa0 libisl23 libitm1 liblsan0 libmpc3 libquadmath0
  libstdc++-13-dev libstdc++13 libubsan1 libxpm4 linux-libc-dev lto-disabled-list make manpages-dev rpcsvc-proto openssl apache2-htcacheclean.service.
Suggested packages:
  bzip2-doc cpp-doc gcc-13-locales cpp-13-doc debian-keyring g++-multilib g++-13-multilib gcc-13-doc gcc-multilib autoconf automake libtool flex bison gdb
  gcc-doc gcc-13-multilib gdb-x86-64-linux-gnu glibc-doc bzip libgdb-tools libhelf-plugin-x265 libhelf-plugin-fmpegdev libhelf-plugin-jpegdev
  libhelf-plugin-jpegenc libhelf-plugin-j2kdec libhelf-plugin-j2kenc libhelf-plugin-ravie libhelf-plugin-svenc libstdc++-13-doc make-doc
The following NEW packages will be installed:
  build-essential bzip2 cpp cpp-13 cpp-13-x86-64-linux-gnu cpp-x86-64-linux-gnu dpkg-dev fakeroot g++ g++-13 g++-13-x86-64-linux-gnu gcc gcc-13
  gcc-13-base gcc-13-x86-64-linux-gnu gcc-x86-64-linux-gnu libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl libbam3
  libasan8 libatomic1 libc-dev-bin libc6-dev libc6-i386 libcc1-0 libcrypt-dev libdeb265-0 libdpkg-perl libfakeroot libfile-fcntllock-perl libgcc-13-dev
  libgdb3 libgomp1 libhelf-plugin-asmdev libhelf-plugin-asmdev libhelf-plugin-libdeb265 libhelf1 libhwasa0 libisl23 libitm1 liblsan0 libmpc3 libquadmath0
  libstdc++-13-dev libstdc++13 libubsan1 libxpm4 linux-libc-dev lto-disabled-list make manpages-dev rpcsvc-proto
0 upgraded, 54 newly installed, 0 to remove and 82 not upgraded.
Need to get 71.3 MB of archives.
After this operation, 242 MB of additional disk space will be used.
Do you want to continue? [Y/n]

```

Installing Graphical Linux Applications

Step 1: It installs the Gedit text editor on Ubuntu.

[illegible]

Step 2: The command `gedit` opens the Gedit text editor in Ubuntu.

```
pradhi@8b5c7dd85f01583: ~  
Processing triggers for hicolor-icon-theme (0.17-2) ...  
Setting up gedit (46.2-2) ...  
/usr/lib/x86_64-linux-gnu/gedit/plugins/externaltools/library.py:212: SyntaxWarning: invalid escape sequence '\-'  
  RE_KEY = re.compile('^([a-zA-Z_][a-zA-Z0-9_\-]*)(\([([a-zA-Z_@]+)\])?$')  
/usr/lib/x86_64-linux-gnu/gedit/plugins/snippets/substitutionparser.py:162: SyntaxWarning: invalid escape sequence '\s'  
  match = re.match('\s\s\s\s*' % self.REG_GROUP, tokens)  
Setting up docbook-xml (4.5-12) ...  
Processing triggers for libc-bin (2.39-0ubuntu8.3) ...  
Setting up libadwaita-1-0:amd64 (1.5.0-1ubuntu2) ...  
Processing triggers for man-db (2.12.0-4build2) ...  
Setting up zenity (4.0.1-1build3) ...  
Setting up libgtk-4-media-gstreamer (4.14.2+ds-1ubuntu1) ...  
Processing triggers for dictionaries-common (1.29.7) ...  
aspell-autobuildhash: processing: en [en-common].  
aspell-autobuildhash: processing: en [en-variant_0].  
aspell-autobuildhash: processing: en [en-variant_1].  
aspell-autobuildhash: processing: en [en-variant_2].  
aspell-autobuildhash: processing: en [en-w_accents-only].  
aspell-autobuildhash: processing: en [en-wo_accents-only].  
aspell-autobuildhash: processing: en [en_AU-variant_0].  
aspell-autobuildhash: processing: en [en_AU-variant_1].  
aspell-autobuildhash: processing: en [en_AU-w_accents-only].  
aspell-autobuildhash: processing: en [en_AU-wo_accents-only].  
aspell-autobuildhash: processing: en [en_CA-variant_0].  
aspell-autobuildhash: processing: en [en_CA-variant_1].  
aspell-autobuildhash: processing: en [en_CA-w_accents-only].  
aspell-autobuildhash: processing: en [en_CA-wo_accents-only].  
aspell-autobuildhash: processing: en [en_GB-ise-w_accents-only].  
aspell-autobuildhash: processing: en [en_GB-ise-wo_accents-only].  
aspell-autobuildhash: processing: en [en_GB-ize-w_accents-only].  
aspell-autobuildhash: processing: en [en_GB-ize-wo_accents-only].  
aspell-autobuildhash: processing: en [en_GB-variant_0].  
aspell-autobuildhash: processing: en [en_GB-variant_1].  
aspell-autobuildhash: processing: en [en_US-w_accents-only].  
aspell-autobuildhash: processing: en [en_US-wo_accents-only].  
Processing triggers for sgml-base (1.31) ...  
Processing triggers for libc-bin (2.39-0ubuntu8.3) ...  
pradhi@8b5c7dd85f01583:~$ gedit
```

Gedit:

