Basic Ubuntu Commands for Beginner:

1. sudo

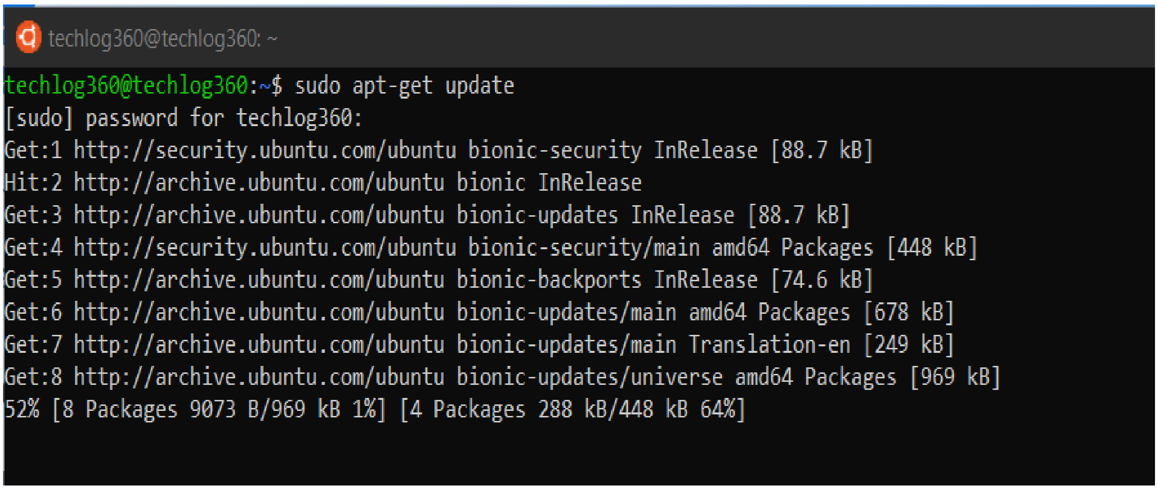
sudo (SuperUser DO) Linux command allows you to run programs or other commands with administrative privileges, just like “Run as administrator” in Windows. This is useful when, for example, you need to modify files in a directory that your user wouldn’t normally have access to.

2. apt-get

apt-get is the one of the most important Ubuntu commands every beginner must know. It is used to install, update, upgrade and remove any package. apt-get basically works on a database of available packages. Here is the list of different apt-get commands:

1. sudo apt-get update

apt-get update with super user privileges is the first command you need to run in any Linux system after a fresh install. This command updates the database and let your system know if there are newer packages available or not.



```
techlog360@techlog360:~$ sudo apt-get update
[sudo] password for techlog360:
Get:1 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Hit:2 http://archive.ubuntu.com/ubuntu bionic InRelease
Get:3 http://archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:4 http://security.ubuntu.com/ubuntu bionic-security/main amd64 Packages [448 kB]
Get:5 http://archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:6 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages [678 kB]
Get:7 http://archive.ubuntu.com/ubuntu bionic-updates/main Translation-en [249 kB]
Get:8 http://archive.ubuntu.com/ubuntu bionic-updates/universe amd64 Packages [969 kB]
52% [8 Packages 9073 B/969 kB 1%] [4 Packages 288 kB/448 kB 64%]
```

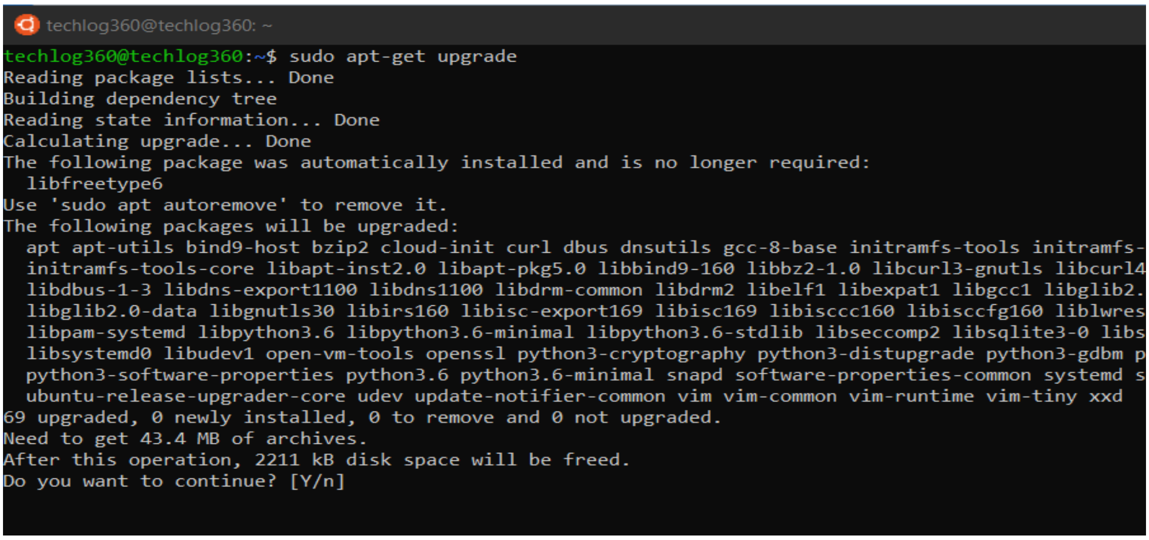
2. sudo apt-get upgrade

After updating the package database, next step is to to upgrade the installed packages. For upgrading all the packages with available updates you can use this command.

And if you like to upgrade a particular package, you should tweak the above command a little:

```
sudo apt-get upgrade <package-name>
```

Replace the <*package-name*> with your desired package.



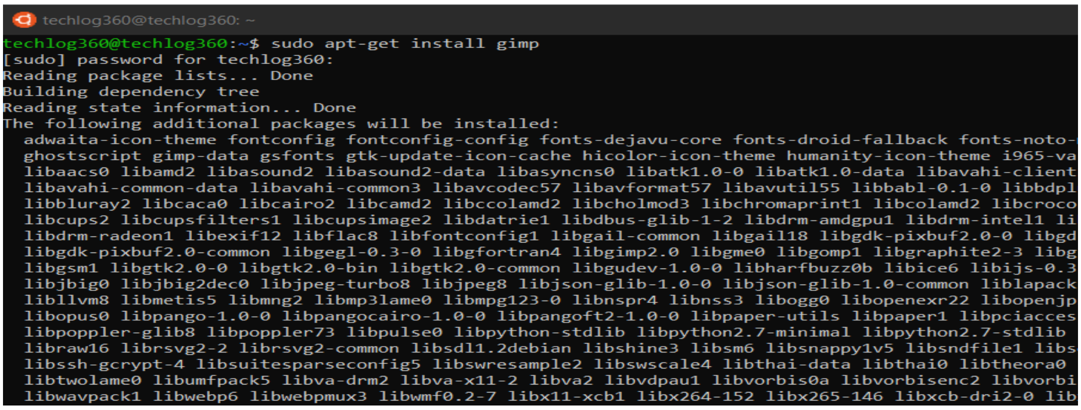
```
techlog360@techlog360:~$ sudo apt-get upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following package was automatically installed and is no longer required:
  libfreetype6
Use 'sudo apt autoremove' to remove it.
The following packages will be upgraded:
  apt apt-utils bind9-host bzip2 cloud-init curl dbus dnsutils gcc-8-base initramfs-tools initramfs-
  initramfs-tools-core libapt-inst2.0 libapt-pkg5.0 libbind9-160 libbz2-1.0 libcurl3-gnutls libcurl4
  libdbus-1-3 libdns-export1100 libdns1100 libdrm-common libdrm2 libelf1 libexpat1 libgcc1 libglib2.
  libglib2.0-data libgnutls30 libirs160 libisc-export169 libisc169 libisccc160 libisccfg160 liblwres
  libpam-systemd libpython3.6 libpython3.6-minimal libpython3.6-stdlib libseccomp2 libsqlite3-0 libs
  libsystemd0 libudev1 open-vm-tools openssl python3-cryptography python3-distupgrade python3-gdbm p
  python3-software-properties python3.6 python3.6-minimal snapd software-properties-common systemd s
  ubuntu-release-upgrader-core udev update-notifier-common vim vim-common vim-runtime vim-tiny xxd
69 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Need to get 43.4 MB of archives.
After this operation, 2211 kB disk space will be freed.
Do you want to continue? [Y/n]
```

3. sudo apt-get install

If you know the name of the package, then you can easily install a program using this command:

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

Replace the <*package-name*> with your desired package.



```
techlog360@techlog360:~$ sudo apt-get install gimp
[sudo] password for techlog360:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  adwaita-icon-theme fontconfig fontconfig-config fonts-dejavu-core fonts-droid-fallback fonts-noto-
  ghostscript gimp-data gsfonts gtk-update-icon-cache hicolor-icon-theme humanity-icon-theme i965-va
  libaacs0 libamd2 libasound2 libasound2-data libasyncns0 libatk1.0-0 libatk1.0-data libavahi-client
  libavahi-common-data libavahi-common3 libavcodec57 libavformat57 libavutil55 libbabl-0.1-0 libbdpl
  libbluray2 libcaca0 libcairo2 libcamd2 libccolamd2 libcholmod3 libchromaprint1 libcolamd2 libcroco
  libcups2 libcupsfilters1 libcupsimage2 libdatrie1 libdbus-glib-1-2 libdrm-amdgpu1 libdrm-intel1 li
  libdrm-radeon1 libexif12 libflac8 libfontconfig1 libgail-common libgail18 libgdk-pixbuf2.0-0 libgd
  libgdk-pixbuf2.0-common libgegl-0.3-0 libgfortran4 libgimp2.0 libgme0 libgomp1 libgraphite2-3 libg
  libgsm1 libgtk2.0-0 libgtk2.0-bin libgtk2.0-common libgudev-1.0-0 libharfbuzz0b libice6 libijs-0.3
  libjbig0 libjbig2dec0 libjpeg-turbo8 libjpeg8 libjson-glib-1.0-0 libjson-glib-1.0-common liblapack
  libllvm8 libmetis5 libmng2 libmp3lame0 libmpg123-0 libnspr4 libnss3 libogg0 libopenexr22 libopenjp
  libopus0 libpango-1.0-0 libpangocairo-1.0-0 libpangoft2-1.0-0 libpaper-utils libpaper1 libpciacces
  libpoppler-glib8 libpoppler73 libpulse0 libpython-stdlib libpython2.7-minimal libpython2.7-stdlib
  libraw16 librsvg2-2 librsvg2-common libsdl1.2debian libshine3 libsm6 libsnappy1v5 libsndfile1 libs
  libssh-gcrypt-4 libsuitesparseconfig5 libswresample2 libswscale4 libthai-data libthai0 libtheora0
  libtwolame0 libumfpack5 libva-drm2 libva-x11-2 libva2 libvdpau1 libvorbis0a libvorbisenc2 libvorbi
  libwavpack1 libwebp6 libwebpmux3 libwmf0.2-7 libx11-xcb1 libx264-152 libx265-146 libxcb-dri2-0 lib
```

If you are not sure about the package name, you can type a few letters and press tab and it will suggest all the packages available with those letters. Thanks for auto-completion feature.

4. sudo apt-get remove

When it comes to removing the installed program apt-get remove command suits your need. You only have to know the exact package name of the software you want to uninstall.

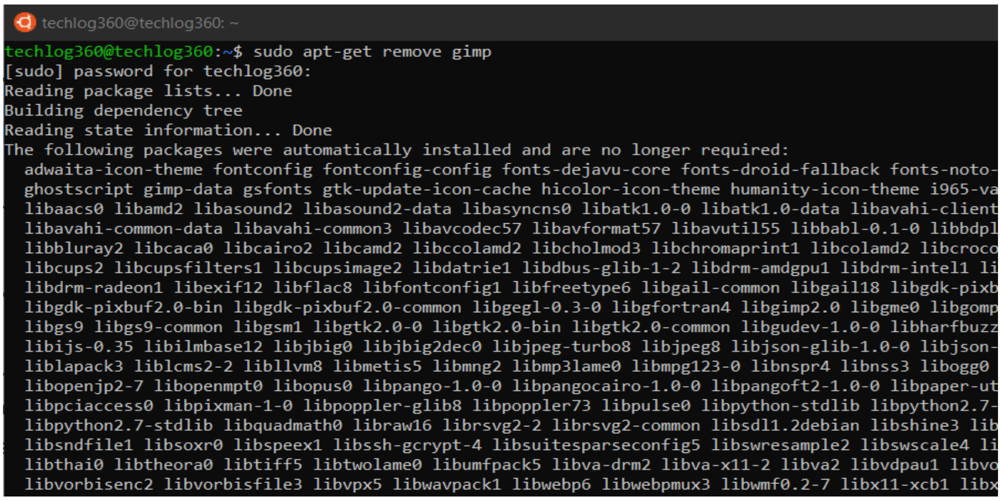
If you don’t know the package name, use below ubuntu basic command to list all the packages installed on your system and then copy the package name from the list:

```
dpkg --list
```

Now run the *apt-get remove* command as sudo in order to remove the software:

```
sudo apt-get remove <package-name>
```

Replace the <*package-name*> with the one you copied from the dpkg list.



```
techlog360@techlog360:~$ sudo apt-get remove gimp
[sudo] password for techlog360:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  adwaita-icon-theme fontconfig fontconfig-config fonts-dejavu-core fonts-droid-fallback fonts-noto-
  ghostscript gimp-data gsfonts gtk-update-icon-cache hicolor-icon-theme humanity-icon-theme i965-va
  libaacs0 libamd2 libasound2 libasound2-data libasyncns0 libatk1.0-0 libatk1.0-data libavahi-client
  libavahi-common-data libavahi-common3 libavcodec57 libavformat57 libavutil55 libbabl-0.1-0 libbdpl
  libbluray2 libcaca0 libcairo2 libcamd2 libccolamd2 libcholmod3 libchromaprint1 libcolamd2 libcroco
  libcups2 libcupsfilters1 libcupsimage2 libdatrie1 libdbus-glib-1-2 libdrm-amdgpu1 libdrm-intel1 li
  libdrm-radeon1 libexif12 libflac8 libfontconfig1 libfreetype6 libgail-common libgail18 libgdk-pixb
  libgdk-pixbuf2.0-bin libgdk-pixbuf2.0-common libgegl-0.3-0 libgfortran4 libgimp2.0 libgme0 libgomp
  libgs9 libgs9-common libgsm1 libgtk2.0-0 libgtk2.0-bin libgtk2.0-common libgudev-1.0-0 libharfbuzz
  libijs-0.35 libilmbase12 libjbig0 libjbig2dec0 libjpeg-turbo8 libjpeg8 libjson-glib-1.0-0 libjson-
  liblapack3 liblcms2-2 libllvm8 libmetis5 libmng2 libmp3lame0 libmpg123-0 libnspr4 libnss3 libogg0
  libopenjp2-7 libopenmpt0 libopus0 libpango-1.0-0 libpangocairo-1.0-0 libpangoft2-1.0-0 libpaper-ut
  libpciaccess0 libpixman-1-0 libpoppler-glib8 libpoppler73 libpulse0 libpython-stdlib libpython2.7-
  libpython2.7-stdlib libquadmath0 libraw16 librsvg2-2 librsvg2-common libsdl1.2debian libshine3 lib
  libsndfile1 libsoxr0 libspeex1 libssh-gcrypt-4 libsuitesparseconfig5 libswresample2 libswscale4 li
  libthai0 libtheora0 libtiff5 libtwolame0 libumfpack5 libva-drm2 libva-x11-2 libva2 libvdpau1 libvo
  libvorbisenc2 libvorbisfile3 libvpx5 libwavpack1 libwebp6 libwebpmux3 libwmf0.2-7 libx11-xcb1 libx
```

apt-get remove command only removes the software from your system but not the configuration or data files of the package. These files help in keeping the same settings when you want to reinstall the same software.

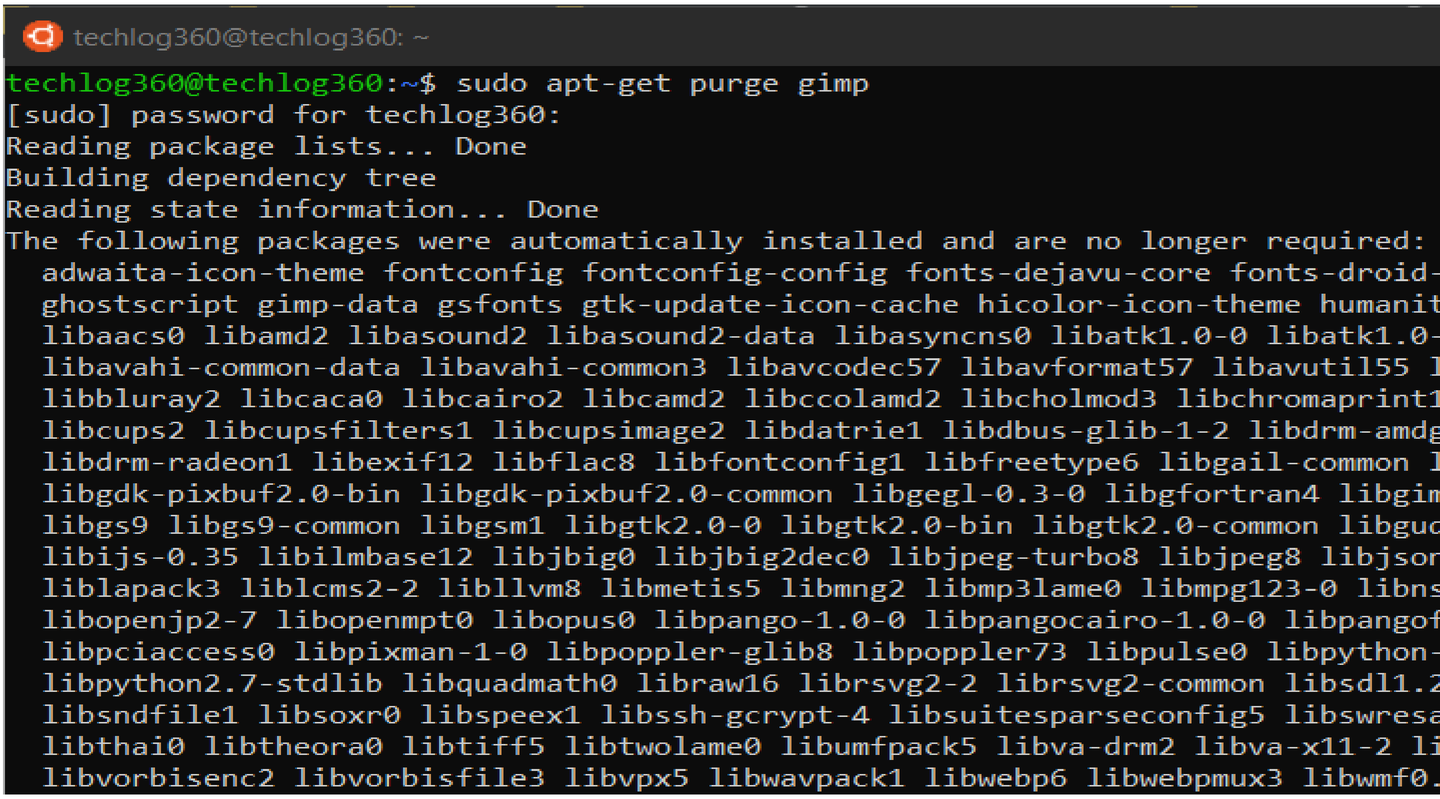
5. sudo apt-get purge

apt-get purge command is used when you want to remove a software completely from your system with its configuration or data files so that no longer personalized settings will be available during reinstallation.

Run the *apt-get purge* command as sudo in order to remove the software completely:

```
sudo apt-get purge <package-name>
```

Replace the <*package-name*> with the application that you want to remove or copied from the dpkg list.

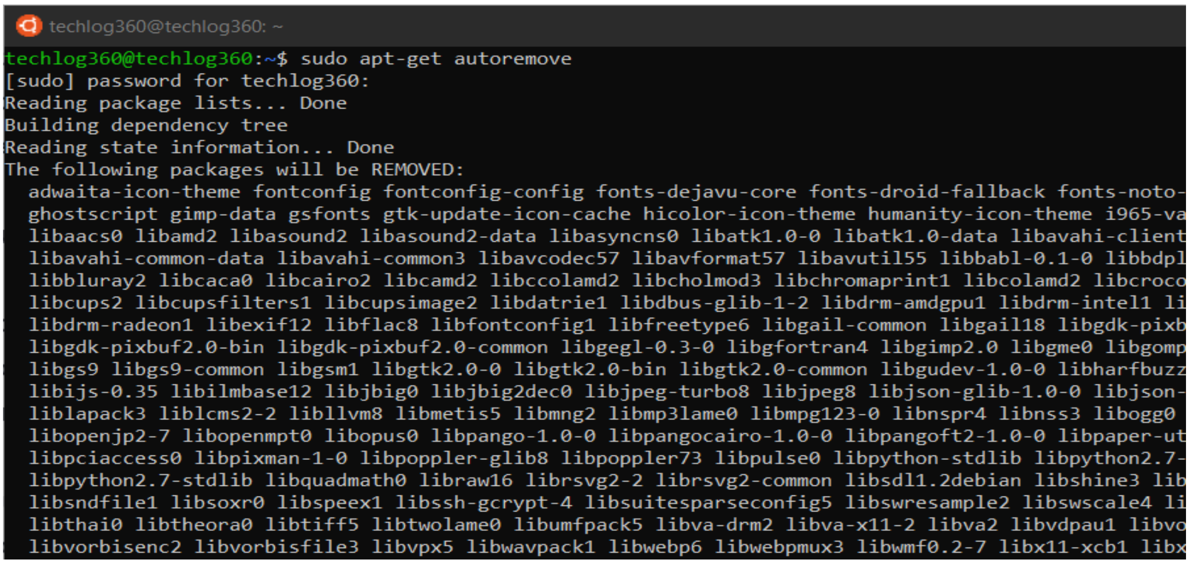


```
techlog360@techlog360:~$ sudo apt-get purge gimp
[sudo] password for techlog360:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  adwaita-icon-theme fontconfig fontconfig-config fonts-dejavu-core fonts-droid-
  ghostscript gimp-data gsfonts gtk-update-icon-cache hicolor-icon-theme humanit
  libaacs0 libamd2 libasound2 libasound2-data libasyncns0 libatk1.0-0 libatk1.0-
  libavahi-common-data libavahi-common3 libavcodec57 libavformat57 libavutil55 l
  libbluray2 libcaca0 libcairo2 libcamd2 libccolamd2 libcholmod3 libchromaprint1
  libcups2 libcupsfilters1 libcupsimage2 libdatrie1 libdbus-glib-1-2 libdrm-amdg
  libdrm-radeon1 libexif12 libflac8 libfontconfig1 libfreetype6 libgail-common l
  libgdk-pixbuf2.0-bin libgdk-pixbuf2.0-common libgegl-0.3-0 libgfortran4 libgim
  libgs9 libgs9-common libgsm1 libgtk2.0-0 libgtk2.0-bin libgtk2.0-common libguc
  libijs-0.35 libilmbase12 libjbig0 libjbig2dec0 libjpeg-turbo8 libjpeg8 libjsor
  liblapack3 liblcms2-2 libllvm8 libmetis5 libmng2 libmp3lame0 libmpg123-0 libns
  libopenjp2-7 libopenmpt0 libopus0 libpango-1.0-0 libpangocairo-1.0-0 libpangof
  libpciaccess0 libpixman-1-0 libpoppler-glib8 libpoppler73 libpulse0 libpython-
  libpython2.7-stdlib libquadmath0 libraw16 librsvg2-2 librsvg2-common libsdl1.2
  libsndfile1 libsoxr0 libspeex1 libssh-gcrypt-4 libsuitesparseconfig5 libswresa
  libthai0 libtheora0 libtiff5 libtwolame0 libumfpack5 libva-drm2 libva-x11-2 li
  libvorbisenc2 libvorbisfile3 libvpx5 libwavpack1 libwebp6 libwebpmux3 libwmf0.
```

6. sudo apt-get autoremove

apt-get autoremove command is used to remove any unnecessary packages. Unnecessary means, whenever you install an application, the system will also install the software that this application depends on. It is common in Ubuntu that applications share the same libraries. When you remove the application the dependency will stay on your system.

So run *apt-get autoremove* as sudo after uninstalling a package to remove unwanted software dependencies.



techlog360@techlog360: ~

```
techlog360@techlog360:~$ sudo apt-get autoremove
[sudo] password for techlog360:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be REMOVED:
  adwaita-icon-theme fontconfig fontconfig-config fonts-dejavu-core fonts-droid-fallback fonts-noto-
  ghostscript gimp-data gsfonts gtk-update-icon-cache hicolor-icon-theme humanity-icon-theme i965-va
  libaacs0 libamd2 libasound2 libasound2-data libasyncns0 libatk1.0-0 libatk1.0-data libavahi-client
  libavahi-common-data libavahi-common3 libavcodec57 libavformat57 libavutil55 libbabl-0.1-0 libbdpl
  libbluray2 libcaca0 libcairo2 libcamd2 libccolamd2 libcholmod3 libchromaprint1 libcolamd2 libcroco
  libcups2 libcupsfilters1 libcupsimage2 libdatrie1 libdbus-glib-1-2 libdrm-amdgpu1 libdrm-intel1 li
  libdrm-radeon1 libexif12 libflac8 libfontconfig1 libfreetype6 libgail-common libgail18 libgdk-pixb
  libgdk-pixbuf2.0-bin libgdk-pixbuf2.0-common libgegl-0.3-0 libgfortran4 libgimp2.0 libgme0 libgomp
  libgs9 libgs9-common libgsm1 libgtk2.0-0 libgtk2.0-bin libgtk2.0-common libgudev-1.0-0 libharfbuzz
  libijs-0.35 libilmbase12 libjbig0 libjbig2dec0 libjpeg-turbo8 libjpeg8 libjson-glib-1.0-0 libjson-
  liblapack3 liblcms2-2 libllvm8 libmetis5 libmng2 libmp3lame0 libmpg123-0 libnspr4 libnss3 libogg0
  libopenjp2-7 libopenmpt0 libopus0 libpango-1.0-0 libpangocairo-1.0-0 libpangoft2-1.0-0 libpaper-ut
  libpciaccess0 libpixman-1-0 libpoppler-glib8 libpoppler73 libpulse0 libpython-stdlib libpython2.7-
  libpython2.7-stdlib libquadmath0 libraw16 librsvg2-2 librsvg2-common libsdl1.2debian libshine3 lib
  libsndfile1 libsoxr0 libspeex1 libssh-gcrypt-4 libsuitesparseconfig5 libswresample2 libswscale4 li
  libthai0 libtheora0 libtiff5 libtwolame0 libumfpack5 libva-drm2 libva-x11-2 libva2 libvdpau1 libvo
  libvorbisenc2 libvorbisfile3 libvpx5 libwavpack1 libwebp6 libwebpmux3 libwmf0.2-7 libx11-xcb1 libx
```

So apt-get autoremove will remove those dependencies that were installed with applications and that are no longer used by anything else on the system.

3. ls

ls (list) command lists all files and folders in your current working directory. You can also specify paths to other directories if you want to view their contents.

4. cd

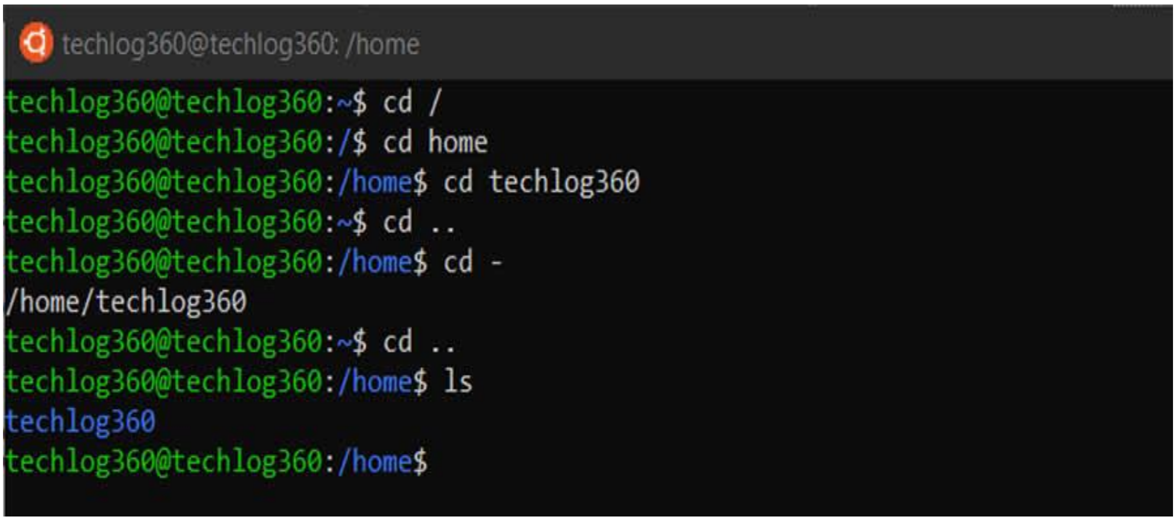
cd (change director”) Linux command also known as chdir used to change the current working directory. It’s one of the most used basic Ubuntu commands. Using this command is easy, just type cd followed by the the folder name. You can use full paths to folders or simply the name of a folder within the directory you are currently working. Some common uses are:

❿ cd / – Takes you to the root directory.
❿ cd .. – Takes you up one directory level.
❿ cd – – Takes you to the previous directory.

Here are some examples to how to use cd command in Ubuntu:

Example 1: **cd home** – open home folder in current directory.

Example 2: *cd Linux\ Drive* – open Linux Drive named folder in directory. Here you can see I use backslash because the folder name has spaces so *for each space you use “backslash+space”*. Like, if your folder name is “am a programmer” then the cd command will be, “cd am\ a\ programmer”.



techlog360@techlog360: /home

```
techlog360@techlog360:~$ cd /
techlog360@techlog360:/$ cd home
techlog360@techlog360:/home$ cd techlog360
techlog360@techlog360:~$ cd ..
techlog360@techlog360:/home$ cd -
/home/techlog360
techlog360@techlog360:~$ cd ..
techlog360@techlog360:/home$ ls
techlog360
techlog360@techlog360:/home$
```

5. pwd

pwd (print working directory) Ubuntu command displays the full pathname of the current working directory.

6. cp

cp (copy) Linux command allows you to copy a file. You should specify both the file you want to be copied and the location you want it copied to – f*or example, cp* xyz */home/*myfiles *would copy the file “*xyz*” to the directory “/home/*myfiles*”*.

7. mv

mv (move) command allows you to move files. You can also rename files by moving them to the directory they are currently in, but under a new name. The usage is the same as cp – f*or example mv* xyz */home/*myfiles *would move the file “*xyz*” to the directory “/home/*myfiles*”*.

8. rm

rm (remove) command removes the specified file.

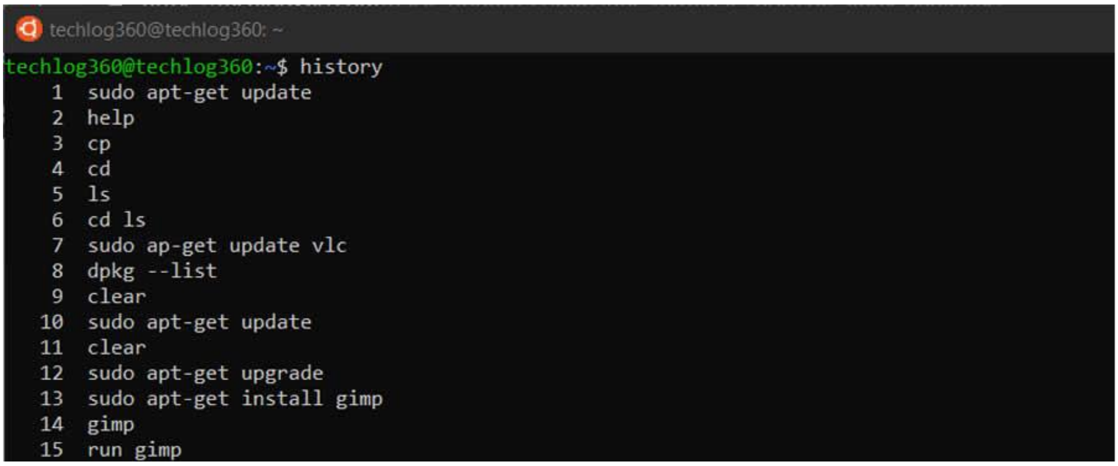
❿ rmdir (“remove directory”) – Removes an empty directory.
❿ rm -r (“remove recursively”) – Removes a directory along with its content.

9. mkdir

mkdir (make directory) command allows you to create a new directory. You can specify where you want the directory created – if you do not do so, it will be created in your current working directory.

10. history

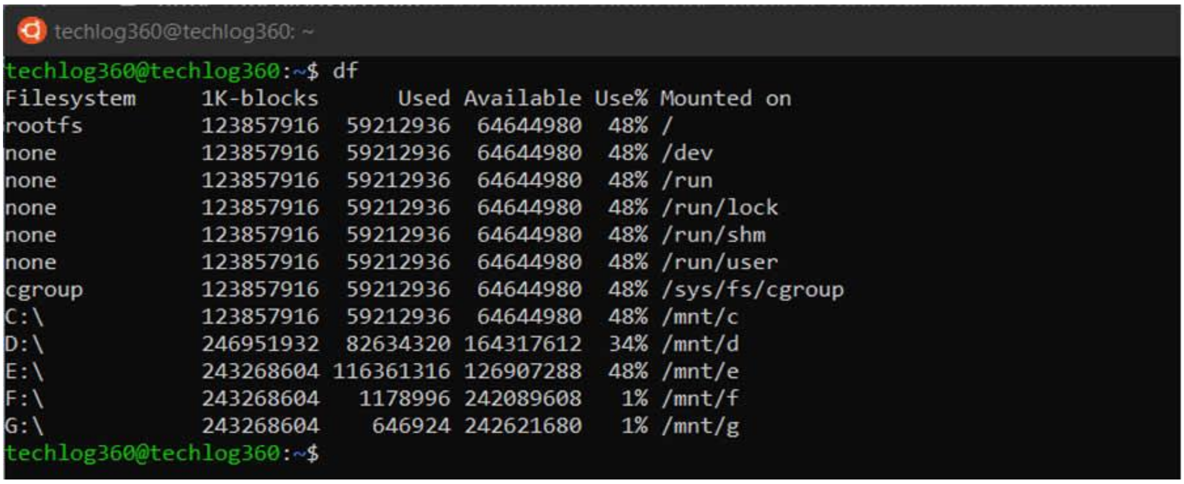
history command displays all of your previous commands up to the history limit.



```
techlog360@techlog360: ~
techlog360@techlog360:~$ history
    1  sudo apt-get update
    2  help
    3  cp
    4  cd
    5  ls
    6  cd ls
    7  sudo ap-get update vlc
    8  dpkg --list
    9  clear
   10  sudo apt-get update
   11  clear
   12  sudo apt-get upgrade
   13  sudo apt-get install gimp
   14  gimp
   15  run gimp
```

11. df

df (display filesystem) command displays information about the disk space usage of all mounted filesystems.



```
techlog360@techlog360: ~
techlog360@techlog360:~$ df
Filesystem     1K-blocks      Used Available Use% Mounted on
rootfs         123857916  59212936  64644980  48% /
none           123857916  59212936  64644980  48% /dev
none           123857916  59212936  64644980  48% /run
none           123857916  59212936  64644980  48% /run/lock
none           123857916  59212936  64644980  48% /run/shm
none           123857916  59212936  64644980  48% /run/user
cgroup         123857916  59212936  64644980  48% /sys/fs/cgroup
C:\            123857916  59212936  64644980  48% /mnt/c
D:\            246951932  82634320 164317612  34% /mnt/d
E:\            243268604 116361316 126907288  48% /mnt/e
F:\            243268604   1178996 242089608   1% /mnt/f
G:\            243268604    646924 242621680   1% /mnt/g
techlog360@techlog360:~$
```

12. du

du (directory usage) command displays the size of a directory and all of its subdirectories.

13. free

free – Displays the amount of free space available on the system.

14. uname -a

uname -a – Provides a wide range of basic information about the system.

15. top

top – Displays the processes using the most system resources at any given time. “q” can be used to exit.

16. man

man command displays a “manual page”. Manual pages are usually very detailed, and it’s recommended that you read the man pages for any command you are unfamiliar with. Some uses are :

- ➓ man man – Provides information about the manual itself.
- ➓ man intro – Displays a brief introduction to Linux commands.

17. info

Similar to man, but often provides more detailed or precise information.

18. <command name> -h or <command name> –help

This command is a third alternative to get help. While not as detailed as the info or man pages, this will provide a quick overview of the command and its uses.

For example: *man -h* or *man -help*

19. passwd

passwd Ubuntu basic command is used to change user password using Terminal. What you have to do is run the below command, where is the username whose password has to change:

```
passwd <user>
```

20. whatis

whatis command shows a brief description of what is the functionality of specific built-in Linux command.

```
whatis <command>
```

Some examples are:

- ➓ whatis cd
- ➓ whatis man
- ➓ whatis help

Above commands will display the purpose of cd, man and help commands.

Ubuntu Terminal Shortcuts:

To further ease up your skill, these Ubuntu Terminal keyboard shortcuts would help.

Ubuntu Terminal Shortcuts	Function
Ctrl + Shift + T	Open new tab on current terminal
Ctrl + Shift + W	Close the current tab
Ctrl + A	Move cursor to beginning of line
Ctrl + E	Move cursor to end of line
Ctrl + U	Clears the entire current line
Ctrl + K	Clears the command from the cursor right
Ctrl + W	Delete the word before the cursor
Ctrl + R	Allows you to search your history for commands matching what you have typed
Ctrl + C	Kill the current process
Ctrl + Z	Suspend the current process by sending the signal SIGSTOP
Ctrl + L	Clears the terminal output

Alt + F	Move forward one word
Alt + B	Move backward one word
Ctrl + Shift + C	Copy the highlighted command to the clipboard
Ctrl + Shift + V or Shift + Insert	Paste the contents of the clipboard
Up/Down Arrow keys	To scroll through your command history, allowing you to quickly execute the same command multiple times
TAB	Used to complete the command you are typing. If more than one command is possible, you can press it multiple times to scroll through the possible completions. If a very wide number of commands are possible, it can output a list of all possible completions.

Git & GitHub

Git is version-control software. It was created by Linus Torvalds, the guy who invented Linux. You don't have to be running Linux to use git–you can use it on Windows and Mac, too
Git lets you easily keep track of every revision you and your team make during the development of your software. You all share one repository of code that is worked on independently and then merged back together. You also do not need to be connected all the time because the project is both saved locally on each machine and remotely (probably at Github).

Required Vocabulary

When you're first starting to use git, it can be intimidating. Commands and concepts may seem totally foreign if you've never used any form of version control in the past. You will pick them up super fast.

Repository:

Often referred to as a repo. A repository is the collection of files and folders that you're using git to track. The repository consists of the entire history of your team's changes to the project. It's the big ole box you and your team throw your code into.

Github:

The most popular remote storage solution for git repos. It also allows you to set access permissions for projects, track and submit bugs, accept feature requests, subscribe to repository notifications, and utilize a graphic interface, rather than use the command line. Repos default to public, but paid accounts can have private ones.

Clone:

Cloning a repo is pretty much exactly what it sounds like. It takes the entire online repository and makes an exact copy of it on your local machine. You will need to do this for any number of reasons, not the least of which are starting in the middle of a project with a new team, swapping workstations, or starting over from a corrupted repo.

Commit:

Think of this as saving your work. When you commit to a repository, it's like you're gathering up the files as they exist at that moment and putting them in a time capsule. The commit will only exist on your local machine until it is pushed to a remote repository.

Push:

Pushing is the bread to committing's butter. Committing throws your files into the timecapsule, and pushing is what launches the capsule into space. Pushing is essentially syncing your commits to the cloud (again, probably Github). You can push multiple commits at once, too. You can work offline, commit lots of work, and then push it all up to Github when you're back in civilization with that sweet, sweet wifi.

https://www.youtube.com/watch?v=ZMgLZUYd8Cw&t=430s