APT Debian package manager

START HERE

`sudo apt update`
Refresh package lists

`sudo apt install <pkg>`
Install a package

`sudo apt upgrade`
Install available updates

`sudo apt remove <pkg>`
Uninstall, keep config

`apt search <term>`
Find a package

`sudo apt autoremove`
Drop unused dependencies

`apt show <pkg>`
Read package details

`apt list --upgradeable`
See pending updates

Sources sources.list → `apt update` → **Package lists** local index → `apt install` → **Downloaded .deb** /var/cache/apt → `dpkg` → **Installed system** dpkg database

CONCEPTS

> update only refreshes the lists; upgrade is what installs new versions. Run them in that order.

> apt is the interactive front end; apt-get and apt-cache behave more stably and are better in scripts.

> Debian and Ubuntu family only. Changing the system needs root, so prefix with sudo.

UPDATE AND UPGRADE

`sudo apt update`
Refresh the package lists

`sudo apt upgrade`
Upgrade, never remove packages

`sudo apt full-upgrade`
Upgrade, allow removals if needed
Use it when upgrade holds packages back; read the removal list first.

`apt list --upgradeable`
List packages with updates

`sudo apt install --only-upgrade <pkg>`
Upgrade one package only

`sudo apt -s upgrade`
Simulate, change nothing

`sudo apt -y upgrade`
Answer yes to prompts

`sudo apt-get dist-upgrade`
Script-friendly full-upgrade

INSTALL AND REMOVE

`sudo apt install <pkg>`
Install a package

`sudo apt install <pkg1> <pkg2>`
Install several at once

`sudo apt remove <pkg>`
Remove, keep config files

`sudo apt purge <pkg>`
Remove including config files

INSTALL AND REMOVE (cont.)

`sudo apt autoremove`
Remove unneeded dependencies

`sudo apt autoremove --purge`
Also delete their config

`sudo apt install ./<file>.deb`
Install a local .deb with deps

`sudo apt reinstall <pkg>`
Reinstall a package

`sudo apt install <pkg>=<version>`
Install an exact version

`sudo apt install -t <release> <pkg>`
Install from another release

`sudo apt install --no-install-recommends <pkg>`
Skip recommended packages

`sudo apt install -f`
Repair broken dependencies

`sudo apt build-dep <pkg>`
Install build dependencies

`sudo apt satisfy "<dep-string>"`
Install from a dependency string

SEARCH AND INSPECT

`apt search <term>`
Search names and descriptions

`apt show <pkg>`
Show version, depends, size

`apt list --installed`
List installed packages

`apt list --installed | grep <name>`
Is it installed?

`apt list --all-versions <pkg>`
Show every available version

`apt policy <pkg>`
Installed, candidate, sources

`apt depends <pkg>`
What a package needs

`apt rdepends <pkg>`
What needs a package

`apt-cache pkgnames <prefix>`
Complete package names

SEARCH AND INSPECT (cont.)

`apt changelog <pkg>`
Show the package changelog

`apt download <pkg>`
Fetch the .deb without installing

HOLD AND MARK

`sudo apt-mark hold <pkg>`
Freeze at current version

`sudo apt-mark unhold <pkg>`
Allow upgrades again

`apt-mark showhold`
List held packages

`apt-mark showmanual`
List manually installed packages

`sudo apt-mark manual <pkg>`
Protect from autoremove

`sudo apt-mark auto <pkg>`
Let autoremove take it

`apt-mark showauto`
List auto-installed packages

> Autoremove only removes packages marked auto. Mark a package manual to keep it.

SOURCES AND CACHE

`sudo apt edit-sources`
Edit the sources file

`sudo apt modernize-sources`
Convert .list to .sources

`ls /etc/apt/sources.list.d/`
See extra repositories

`sudo apt clean`
Delete all downloaded .deb files

`sudo apt-get autoclean`
Delete obsolete downloads only

`du -sh /var/cache/apt/archives`
Size of the download cache

`apt -o Dir::Cache=<dir> update`
Override a config option

`apt-get check`
Verify no broken dependencies

> Add third-party repos with their signing key kept in /etc/apt/keyrings, not the old apt-key.

DPKG PAIRING

`dpkg -l`
List installed packages

`dpkg -l <pattern>`
Status of matching packages

`dpkg -L <pkg>`
Files a package installed

`dpkg -S <path>`
Which package owns a file

`dpkg -s <pkg>`
Installed status and details

`sudo dpkg -i <file>.deb`
Install a .deb, no deps
Does not fetch dependencies; follow with apt install -f if it fails.

`sudo dpkg --configure -a`
Finish interrupted installs

`dpkg --get-selections`
Dump package selections

`dpkg --print-architecture`
Show native architecture

`sudo dpkg --add-architecture <arch>`
Enable a foreign architecture

RECIPES

Routine system update

```
sudo apt update
apt list --upgradeable
sudo apt upgrade
sudo apt autoremove
```

Find which package provides a file

```
dpkg -S /usr/bin/<cmd>
# not installed yet? use apt-file
sudo apt install apt-file &&
sudo apt-file update
apt-file search <path>
```

Pin a package, then release it

```
sudo apt-mark hold <pkg>
apt-mark showhold
sudo apt upgrade   # skips it
sudo apt-mark unhold <pkg>
```

Fix a broken install

```
sudo dpkg --configure -a
sudo apt install -f
sudo apt update && sudo apt
full-upgrade
```