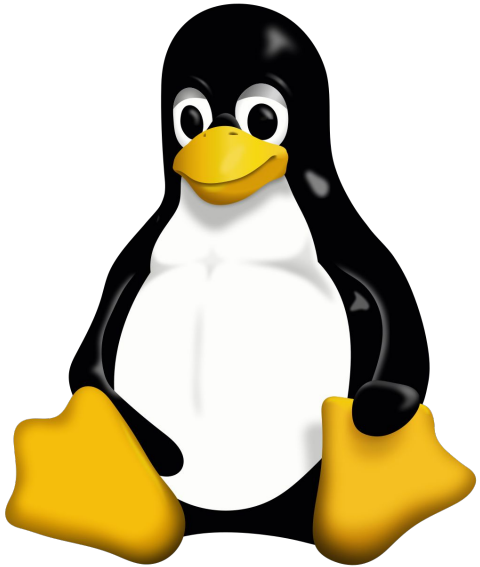
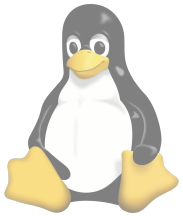


Intro to Linux

ubuntu

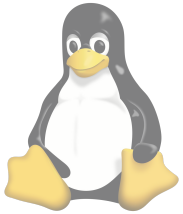


Why Linux?

- Open source code
- Security
- Scalability
- Package management
- Flexibility and variety
- Performance
- Simplicity

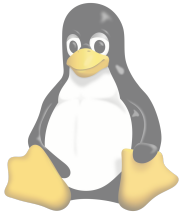
Is it for me?

- Users, who are new to Linux, usually shun it by falsely considering it as a difficult and technical OS to operate but, to state the truth, in the last few years Linux operating systems have become a lot more user-friendly than their counterparts like Windows, so trying them is the best way to know whether Linux suits you or not.
- There are thousands of Best Linux OSs and Linux softwares available based on the Linux Kernel; most of them offer state-of-the-art security and applications, all of it for free!



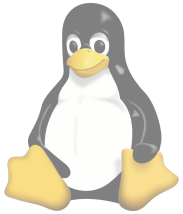
I am asked to Learn Unix? Then why Linux?

- UNIX is called the mother of operating systems which laid out the foundation to Linux. Unix is designed mainly for mainframes and is in enterprises and universities. While Linux is fast becoming a household name for computer users, developers, and server environment. You may have to pay for a Unix kernel while in Linux it is free.
- But, the commands used on both the operating systems are usually the same. There is not much difference between UNIX and Linux. Though they might seem different, at the core, they are essentially the same. Since Linux is a clone of UNIX. So learning one is same as learning another.



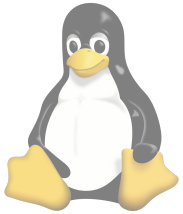
Linux distributions

A **Linux distribution** (often abbreviated as **distro**) is an operating system made from a software collection that is based upon the Linux kernel and, often, a package management system. Linux users usually obtain their operating system by downloading one of the Linux distributions, which are available for a wide variety of systems ranging from embedded devices (for example, OpenWrt) and personal computers (for example, Linux Mint) to powerful supercomputers (for example, Rocks Cluster Distribution).



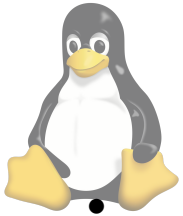
Widely used GNU-based or GNU-compatible distributions:

- **Debian**, a non-commercial distribution and one of the earliest, maintained by a volunteer developer community with a strong commitment to free software principles and democratic project management.
    - **Knoppix**, the first Live CD distribution to run completely from removable media without installation to a hard disk, derived from Debian.
    - **Linux Mint Debian Edition** (LMDE) uses Debian packages directly (rather than Ubuntu's)
    - **Ubuntu**, a desktop and server distribution derived from Debian, maintained by British company Canonical Ltd.
        - **Linux Mint**, a distribution based on and compatible with Ubuntu. Supports multiple desktop environments, among others GNOME Shell fork Cinnamon and GNOME 2 fork MATE.
- **Fedora**, a community distribution sponsored by American company Red Hat and the successor to the company's previous offering,
- **Red Hat Linux**. It aims to be a technology testbed for Red Hat's commercial Linux offering, where new open source software is prototyped, developed, and tested in a communal setting before maturing into Red Hat Enterprise Linux.
        - **CentOS**, a distribution derived from the same sources used by Red Hat, maintained by a dedicated volunteer community of developers with both 100% Red Hat-compatible versions and an upgraded version that is not always 100% upstream compatible.
        - **Oracle Linux,** which is a derivative of Red Hat Enterprise Linux, maintained and commercially supported by Oracle



Widely used GNU-based or GNU-compatible distributions:

- Android, Google's commercial operating system based on Android OSP that runs on many devices such as smartphones, smart TVs, set-top boxes.
- Chrome OS, Google's commercial operating system based on Chromium OS that only runs on Chromebooks, Chromeboxes and tablet computers. Like Android, it has the Google Play Store and other Google apps. Support for applications that require GNU compatibility is available through a virtual machine called Crostini and referred to by Google as Linux support, see Chromebook#Integration with Linux.

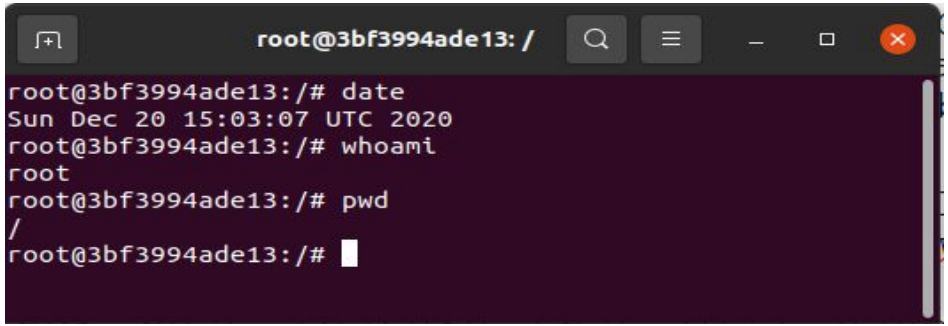


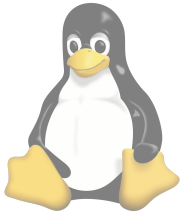
Bash or Bourne Again SHell

- Bash (Bourne-again shell) is a command language interpreter. Simply put, it is a text windows interprets commands that you type. It was written by Brian Fox for the GNU Project as a replacement for the Bourne shell (sh), one of the earliest UNIX shells. Bash offers functional improvements over sh for both programming and interactive use.

- Bash is the default shell in most Linux distributions. It supports some advanced features like wildcarding, piping, command substitution, variables, and the history of commands entered. Another neat feature that will save you a lot of time is the tab completion, which means that you can type just enough of the filename to uniquely identify it and than press the Tab key. Bash will automatically complete your command.

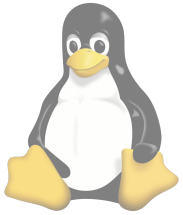
- Here is a picture of Bash in Ubuntu:





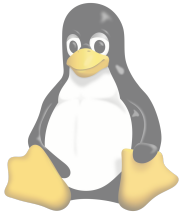
Linux Basic Commands

- **pwd** - print the name of the current working directory
- **cd** - change the working directory
- **ls** - lists directory contents of files and directories
- **cat** - display the content of text files and combine several files to one file
- **cp** - copy files and directories
- **mv** - move files and directories
- **mkdir** - make directory(ies) if they do not already exist
- **rmdir** - remove directories
- **rm** - remove files and directories
- **touch** - create file or update file stats
- **locate** – to search for a file
- **find** - Similar to the locate command, using find also searches for files and directories. The difference is, you use the find command to locate files within a given directory.
- **grep** - for searching plain-text data sets for lines that match a regular expression.

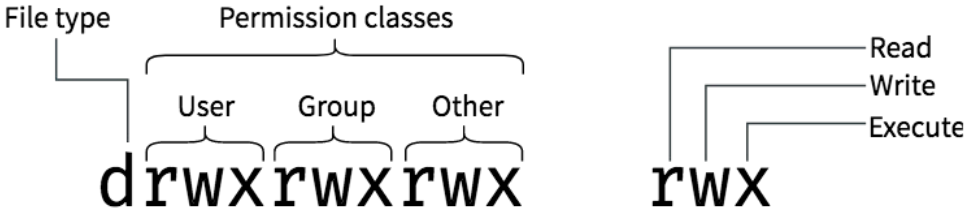


Linux Basic Commands

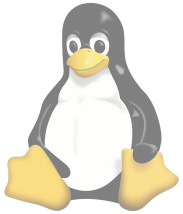
- **df** - get a report on the system's disk space usage
- **du** - check how much space a file or a directory takes
- **head** - view the first lines of any text file
- **tail** - display the last ten lines of a text file
- **diff** - compares the contents of two files line by line
- **tar** - archive multiple files into a tarball — a common Linux file format that is similar to zip format, with compression being optional.
- **sudo** - run command with root permissions
- **chmod** - change the read, write, and execute permissions of files and directories
- **chown** - change or transfer the ownership of a file to the specified username



File Permissions in Linux

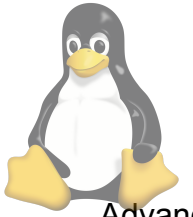


	Permission Type	Symbol
0	No Permission	---
1	Execute	--x
2	Write	-w-
3	Execute + Write	-wx
4	Read	r--
5	Read + Execute	r-x
6	Read +Write	rw-
7	Read + Write +Execute	rwx



Linux Basic Commands

- **jobs** - display all current jobs along with their statuses
- **kill** – terminate the process
- **ping** - check your connectivity status to a server
- **wget** - download files from the internet
- **uname** - print certain system information
- **top** - display a list of running processes and how much CPU each process uses
- **history** - review the commands you've entered before
- **man** - show the manual instruction of the specific command
- **echo** - move some data into a file
- **zip**, **unzip** - compress your files into a zip archive, extract the zipped files from a zip archive
- **hostname** - display the name of your host/network
- **useradd**, **userdel** – add user account or delete user account
- **clear** - clear the terminal screen
- **whoami** - print effective userid



Package manager

Advanced Package Tool, or APT, is a free-software user interface that works with core libraries to handle the installation and removal of software on Debian, Ubuntu, and related Linux distributions. APT simplifies the process of managing software on Unix-like computer systems by automating the retrieval, configuration and installation of software packages, either from precompiled files or by compiling source code.

At this point, I must mention apt-get is perhaps the most popular tool around APT.

Update package database with apt:

sudo apt update

Upgrade installed packages with apt:

sudo apt upgrade

Install new packages with apt:

sudo apt install <package_name>

sudo apt install <package_1> <package_2> <package_3>

sudo apt install <package_name>=<version_number>

Remove installed packages with apt:

sudo apt remove <package_name>

sudo apt purge <package_name>

Search for packages:

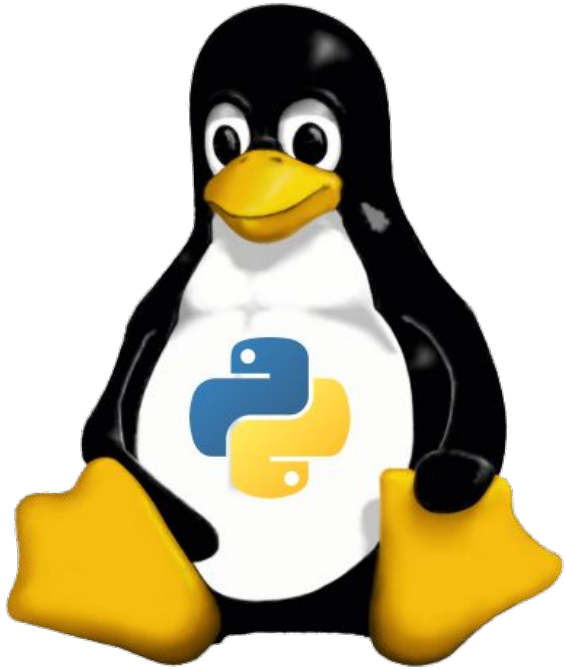
apt search <search term>

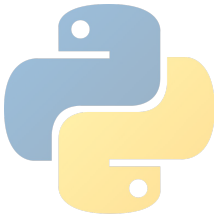
Installed packages:

apt list –installed

apt list --all-versions

Linux & Python





Python installation

Option 1: Install Python 3 Using apt (Easier)

Step 1: Update and Refresh Repository Lists

sudo apt update

Step 2: Install Supporting Software

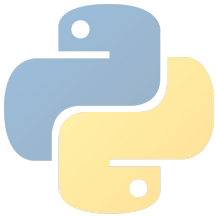
The software-properties-common package gives you better control over your package manager by letting you add PPA (Personal Package Archive) repositories. Install the supporting software with the command:
sudo apt install software-properties-common

Step 3: Add Deadsnakes PPA

Deadsnakes is a PPA with newer releases than the default Ubuntu repositories. Add the PPA by entering the following:
sudo add-apt-repository ppa:deadsnakes/ppa

Step 4: Install Python 3

sudo apt install python3.8
Allow the process to complete and verify the Python version was installed successfully:
python —version



Python installation

Option 2: Install Python 3.7 From Source Code (Latest Version)

Step 1: Update Local Repositories

sudo apt update

Step 2: Install Supporting Software

sudo apt install build-essential zlib1g-dev libncurses5-dev libgdbm-dev libnss3-dev libssl-dev libreadline-dev libffi-dev wget

install additional software for python

Step 3: Download the Latest Version of Python Source Code

To download the newest release of Python Source Code, navigate to the /tmp directory and use the wget command:

cd /tmp

wget https://www.python.org/ftp/python/3.7.5/Python-3.7.5.tgz

Step 4: Extract Compressed Files

tar –xf Python-3.7.5.tgz

Step 5: Test System and Optimize Python

cd python-3.7.5

./configure —enable–optimizations

This step can take up to 30 minutes to complete.

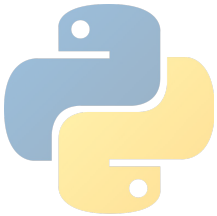
Step 6: Install a Second Instance of Python (recommended)

sudo make altinstall

sudo make install

Step 7: Verify Python Version

python3 —version



Setup Python Virtualenv

Local, save, isolated python environment that can be created per project.

To install use:

python3 -m pip install --user virtualenv

To create use:

python3 -m venv /path/to/new/virtual/environment

To activate use:

source env/bin/activate

Activating a virtual environment will put the virtual environment-specific python and pip executables into your shell's PATH.

To deactivate use:

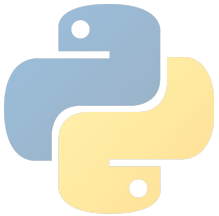
deactivate

To export a list of all installed packages and their versions:

pip freeze > requirements.txt

To install all of the packages in from the requirements.txt file:

pip install -r requirements.txt



Python IDEs and Code Editors

- **IDLE** - when you install Python, IDLE is also installed by default.
- **Sublime Text 3** - Sublime Text is a popular code editor that supports many languages including Python. It's fast, highly customizable and has a huge community.
- **Atom** - Atom is an open-source code editor developed by Github that can be used for Python development (similar Sublime text)
- **PyCharm** - PyCharm is an IDE for professional developers. It is created by JetBrains, a company known for creating great software development tools.
- **Visual Studio Code** - Visual Studio Code (VS Code) is a free and open-source IDE created by Microsoft that can be used for Python development.