```
brew install curl
brew install wget


# download files using curl
curl --output latest.zip https://wordpress.org/latest.zip
curl -o vue-v2.6.10.js https://cdn.jsdelivr.net/npm/vue/dist/vue.js
curl -o curl-8.13.0.tar.gz https://curl.se/download/curl-8.13.0.tar.gz

# extract tar.gz file
tar -xzvf archive.tar.gz
tar -xzvf archive.tar.gz -C /target/directory

# list content without extracting
tar -tvf archive.tar.gz

# download files using wget
wget https://wordpress.org/latest.zip
wget https://github.com/ranaroussi/yfinance/blob/main/README.md
wget https://cdn.kernel.org/pub/linux/kernel/v4.x/linux-4.17.2.tar.xz
wget -O latest-hugo.zip https://github.com/gohugoio/hugo/archive/master.zip
wget -P /home/tsiameh/Desktop http://mirrors.mit.edu/centos/7/isos/x86_64/CentOS-7-x86_64-Minimal-1804.iso
wget -b https://download.opensuse.org/tumbleweed/iso/openSUSE-Tumbleweed-DVD-x86_64-Current.iso
wget -q --show-progress https://code.jquery.com/jquery-3.6.0.min.js
wget -O wordpress-install.zip https://wordpress.org/latest.zip


# create a zip file
zip archive.zip file1.txt file2.txt

# zip a directory recursively
zip -r archive.zip /path/to/directory
# Create a password-protected zip
zip -e secure.zip file.txt

# Update an existing zip file

zip -u archive.zip newfile.txt

# Extract to a specific directory

unzip archive.zip -d /target/directory

# List contents without extracting

unzip -l archive.zip
```

# Login into remote server

```
brew install openssh
ssh ts75230@99.48.1.100 -p 22
```

# copy files from local to remote server

```
scp file.txt user@host:/home/ts75080/Desktop -P 22
```

# download files from remote server to local

```
scp -P 22 user@host:/path .
scp -P 22 user@host:/path /user/tsiameh/Desktop/PythonCourse
```

Sync directories

```
rsync -avz /local/dir username@server-ip:/remote/dir
```

# SED and AWK Usage

Basic Text Replacement

```
# Replace "apple" with "orange" in file.txt
sed 's/apple/orange/g' file.txt
```

Case-Insensitive Replacement

```
# Replace "hello" (any case) with "Hi"
sed 's/hello/Hi/gI' file.txt
```

Example Input:

Hello world, hello everyone

Output:

Hi world, Hi everyone

Print Specific Columns

```
# Print 1st and 3rd columns of data.csv (comma-delimited)
awk -F',' '{print $1, $3}' data.csv

awk -F'\t' '{print $1,$3}' data.tsv  (tab-delimited)

awk -F',' '$2 > 25 {print $1,$3}' data.csv  # Only people over 25

awk -F',' '$2 > 25 {print $0}' data.csv | sed 's/Alice/Trump/g'
```

Example Input (data.csv**):**

```
John,25,USA
Alice,30,Canada
```

Output:

```
John USA
Alice Canada
```

Combining **sed** and **awk**

```
# Extract process IDs (PID) from `ps aux`, then replace "python" with "PY"
ps aux | awk '{print $2}' | sed 's/python/PY/g'
```

chmod (Change File Permissions) Examples

Basic Permission Assignment

```
# Give owner full permissions (7), group read/execute (5), others read/execute (5)
chmod 755 script.sh
```

Result:

```
-rwxr-xr-x script.sh
```

(Owner: read/write/execute, Group/Others: read/execute)

Symbolic Permissions (Human-Readable)

```
# Add execute permission for all users
chmod a+x script.sh

# Remove write permission from others
chmod o-w secret.txt

# Set owner=rwx, group=rx, others=rx (same as 755)
chmod u=rwx,g=rx,o=rx script.sh
```

Recursive Permission Change

```
# Apply 755 to all files/directories in /var/www
chmod -R 755 /var/www
```

chown (Change File Owner/Group) Examples

Basic Ownership Change

```
# Change owner to 'nginx' and group to 'www-data'
chown nginx:www-data /var/www/html
```

Before:

```
-rw-r--r-- 1 root root 1204 Jun 10 index.html
```

After:

```
-rw-r--r-- 1 nginx www-data 1204 Jun 10 index.html
```

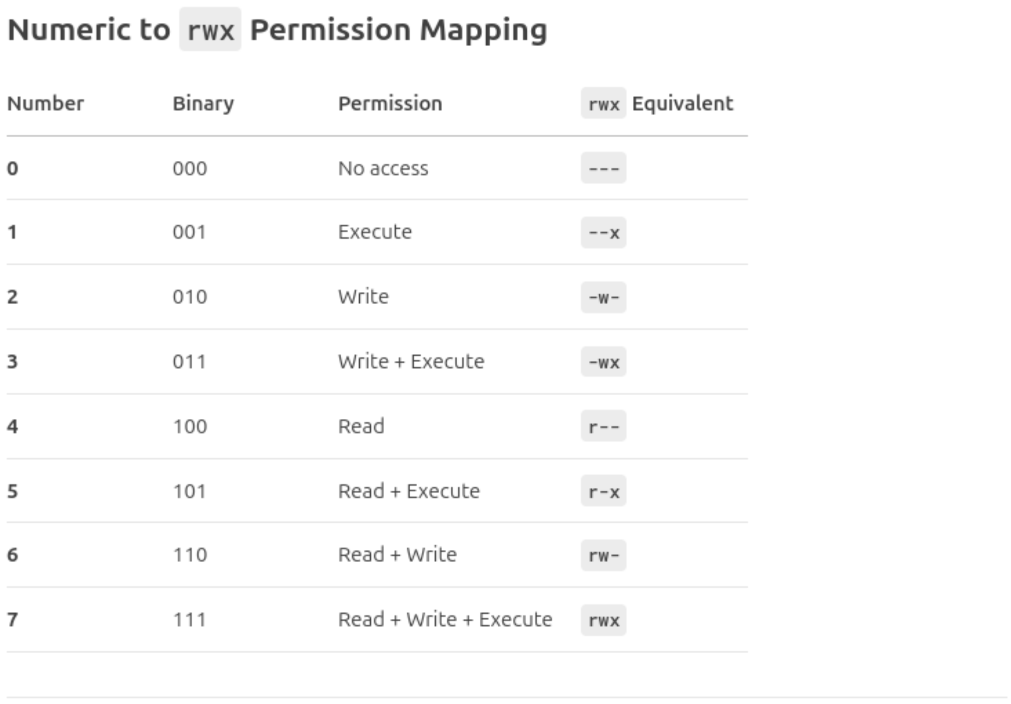
Recursive Ownership

```
# Change owner/group for all files in a directory
chown -R deploy:deploy /opt/myapp
```

Change Only Owner or Only Group

```
# Change owner only (keep group)
chown jenkins /opt/backups

# Change group only (keep owner)
chown :developers /src/code
```



Numeric to `rwx` Permission Mapping

Number	Binary	Permission	`rwx` Equivalent
0	000	No access	`---`
1	001	Execute	`--x`
2	010	Write	`-w-`
3	011	Write + Execute	`-wx`
4	100	Read	`r--`
5	101	Read + Execute	`r-x`
6	110	Read + Write	`rw-`
7	111	Read + Write + Execute	`rwx`

Common Permission Combinations

Numeric	`rwx` Form	Typical Use Case
777	`rwxrwxrwx`	Dangerous! Global read/write/execute (temporary directories only)
755	`rwxr-xr-x`	Scripts/executables (owner: full, others: read/execute)
644	`rw-r--r--`	Config files (owner: read/write, others: read-only)
600	`rw-------`	Private files (owner only, no group/others access)
750	`rwxr-x---`	Group-shared scripts (others: no access)
1777	`rwxrwxrwt`	Sticky bit (e.g., `/tmp` – anyone can add files, but only owners can delete thei

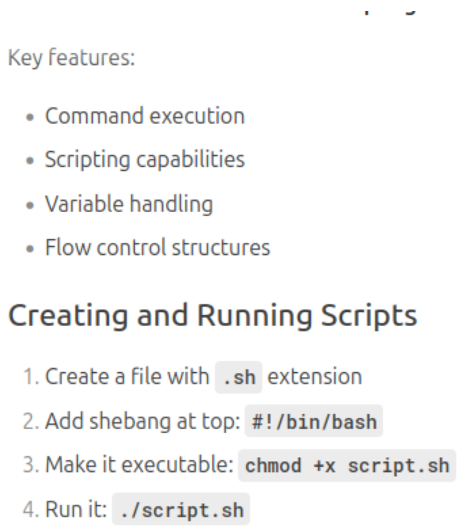
How to Calculate Numeric Permissions

1. **Convert `rwx` to binary**:
   - `r` (read) = **4**
   - `w` (write) = **2**
   - `x` (execute) = **1**
   - `-` (no permission) = **0**
2. **Sum the values for each group**:
   Example: `rwxr-xr--`
   - Owner (`rwx`): 4 (r) + 2 (w) + 1 (x) = **7**
   - Group (`r-x`): 4 (r) + 0 + 1 (x) = **5**
   - Others (`r--`): 4 (r) + 0 + 0 = **4**
     → 754

Table of Contents

Introduction to Bash

Bash (Bourne Again SHell) is a Unix shell and command language. It's widely used for scripting to automate tasks on Linux/Unix systems.



Key features:

- Command execution
- Scripting capabilities
- Variable handling
- Flow control structures

Creating and Running Scripts

1. Create a file with `.sh` extension
2. Add shebang at top: `#!/bin/bash`
3. Make it executable: `chmod +x script.sh`
4. Run it: `./script.sh`

Command	Description	Example
`ls`	List files	`ls -l`
`cd`	Change directory	`cd /home`
`pwd`	Print working directory	`pwd`
`cp`	Copy files	`cp file.txt backup/`
`mv`	Move/rename files	`mv old.txt new.txt`
`rm`	Remove files	`rm file.txt`
`mkdir`	Create directory	`mkdir new_folder`
`rmdir`	Remove empty directory	`rmdir empty_dir`
`touch`	Create empty file	`touch newfile.txt`
`find`	Search for files	`find /home -name "*.txt"`

File Viewing/Editing

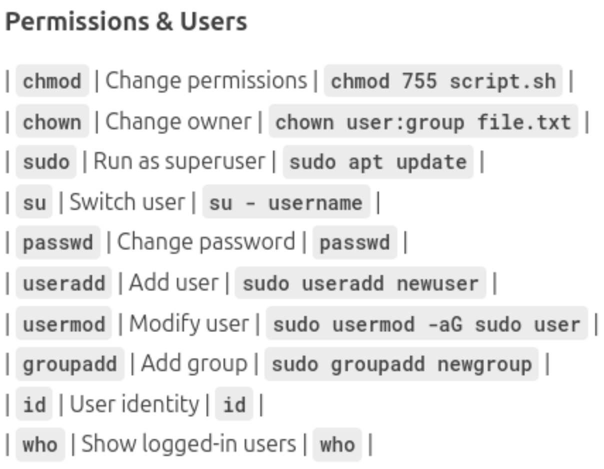
`cat`	Display file content	`cat file.txt`
`less`	View file page by page	`less large.log`
`head`	Show first lines	`head -n 5 file.txt`
`tail`	Show last lines	`tail -f log.txt`
`grep`	Search text	`grep "error" log.txt`
`sed`	Stream editor	`sed 's/old/new/g' file.txt`
`awk`	Text processing	`awk '{print $1}' file.txt`
`nano`	Text editor	`nano file.txt`
`vim`	Advanced editor	`vim file.txt`
`diff`	Compare files	`diff file1.txt file2.txt`

System Information

`uname`	System info	`uname -a`
`top`	Process viewer	`top`
`htop`	Interactive process viewer	`htop`
`ps`	Process status	`ps aux`
`free`	Memory usage	`free -h`
`df`	Disk space	`df -h`
`du`	Directory size	`du -sh *`
`uptime`	System uptime	`uptime`
`whoami`	Current user	`whoami`
`history`	Command history	`history`

Networking

`ping`	Test connection	`ping google.com`
`wget`	Download files	`wget https://example.com/file.zip`
`curl`	Transfer data	`curl -O https://example.com/file.zip`
`ssh`	Remote login	`ssh user@host`
`scp`	Secure copy	`scp file.txt user@host:/path`
`ifconfig`	Network interfaces	`ifconfig`
`netstat`	Network stats	`netstat -tulnp`
`dig`	DNS lookup	`dig example.com`
`traceroute`	Network path	`traceroute google.com`
`hostname`	System hostname	`hostname`



Permissions & Users

`chmod`	Change permissions	`chmod 755 script.sh`
`chown`	Change owner	`chown user:group file.txt`
`sudo`	Run as superuser	`sudo apt update`
`su`	Switch user	`su - username`
`passwd`	Change password	`passwd`
`useradd`	Add user	`sudo useradd newuser`
`usermod`	Modify user	`sudo usermod -aG sudo user`
`groupadd`	Add group	`sudo groupadd newgroup`
`id`	User identity	`id`
`who`	Show logged-in users	`who`

# GITHUB

git clone https://github.com/ranaroussi/yfinance.git

https://github.com/wkentaro/gdown

gdown https://drive.google.com/uc?id=1l_5RK28JRL19wpT22B-DY9We3TVXnnQQ

gdown 1xQ89cpZCnafsW5T3G3ZQWvR7q682t2B