# Linux - Cheatsheet

# Ubuntu/Debian

Ubuntu/Debian

# Resize Filesystem Without Reboot

1. Toggle some informations about the root filesystem:

```
df /
# Filesystem     1K-blocks     Used Available Use% Mounted on
# /dev/vda3       71412992 58923184   9039732  87% /
```

2. `/dev/vda3` is the second partition of block device `/dev/vda`. To size up this partition, you can use the tool `parted`.

```
parted /dev/vda resizepart 3 100%
# Warning: Partition /dev/sda2 is being used. Are you sure you want to continue?
# Yes/No? yes
# Information: You may need to update /etc/fstab.
```

3. After changing size of partition, you have to size up the filsystem too:

```
resize2fs /dev/vda3
# resize2fs 1.45.5 (07-Jan-2020)
# Filesystem at /dev/vda3 is mounted on /; on-line resizing required
# old_desc_blocks = 4, new_desc_blocks = 9
# The filesystem on /dev/vda3 is now 18019403 (4k) blocks long.
```

4. This is for checking modifications:

```
df /
# Filesystem     1K-blocks    Used Available Use% Mounted on
# /dev/vda3       70817420 5253016  62349464   8% /
```

Ubuntu/Debian

# Change Swap Size

1. Turn off all running swap processes: `swapoff -a`
2. Resize swap `fallocate -l 1G /swapfile` (change 1G to the gigabyte size you want it to be)
3. CHMOD swap: `chmod 600 /swapfile`
4. Make file usable as swap `mkswap /swapfile`
5. Active the swap file `swapon /swapfile`

To verify your swap size run the following command and you will see the swap size: `free -m`

Ubuntu/Debian

# Unzip All Files In Directory

## .zip Files

```
unzip "*.zip"
```

Alternative for systems that only have 7zip installed.

```
find /volume1/Emulation/JDownloader/PS2 -type f -iname '*.zip' -execdir 7z x {} ';'
```

## .7z Files

Install 7zip first.

```
sudo apt install p7zip-full
```

```
7za -y x "*.7z"
```

### Unzip into individual directories

```
for archive in *.7z; do 7z x -o"`basename \"$archive\" .7z`" "$archive"; done
```

Ubuntu/Debian

# Zip All Files Into Each Individual Zip File

The solution is pretty easy. If you want to do this for every file, recursively, use `find`. It will list all files and directories, descending into subdirectories too.

## All Subdirectories:

```
find . -type f -execdir zip '{}.zip' '{}' \;
```

Explanation:

- The first argument is the directory you want to begin in, `.`
- Then we will restrict it to find files only (`-type f`)
- The `-execdir` option allows us to run a command on each file found, executing it from the file's directory
- This command is evaluated as `zip file.txt.zip file.txt`, for example, since all occurrences of `{}` are replaced with the actual file name. This command needs to be ended with `\;`

---

Of course, `find` has more options. If instead you just want to stay in your current directory, not descending into subdirectories:

## Current Directory Only:

```
find . -type f -maxdepth 1 -execdir zip '{}.zip' '{}' \;
```

If you want to restrict it to certain file types, use the `-name` option (or `-iname` for case-insensitive matching):

## Restrict To Specific Filetypes:

```
find . -type f -name "*.txt" …
```

Anything else (including looping with `for` over the output of `ls *`) is pretty ugly syntax in my opinion and likely to break, e.g. on files with spaces in their name or due to too many arguments.