```
Arch
        sudo pacman -Syy android-tools
Ubuntu/Debian
        sudo apt install adb fastboot
Fedora
        sudo dnf inStall android-tools
Opensuse
        sudo zypper in android-tools
Gentoo
        emerge -av dev-util/android-tools
```

```
Select the Following link
"https://dl.google.com/android/repository/platform-tools-latest-
windows.zip".

Go to Downloads and extract it. You will need to go into the directory, hold
shift, right click and select "Open Powershell Window" or go to the address
bar and type *cmd* .
```

```
Select the Following link
"https://dl.google.com/android/repository/platform-tools-latest-darwin.zip".

Go to Downloads and extract it. You will need to go to your *terminal*
application. You then will type:
```

```
cd ~/Downloads/[insert the name of the folder it extracted here]
```

```
run .\adb whenever your run adb applications.
```

```
# Install AUR Helper from source, YAY
sudo pacman -S --needed git base-devel
git clone https://aur.archlinux.org/yay.git
cd yay
makepkg -si

OR

# Install AUR Helper from source, YAY, but as a binary
sudo pacman -S --needed git base-devel
git clone https://aur.archlinux.org/yay-bin.git
cd yay-bin
makepkg -si

# Install dependencies
sudo pacman -Syy  --needed base base-devel bc python python2 python-pip
python2-pip jdk8-openjdk perl git gnupg flex bison gperf zip unzip sdl wxgtk
squashfs-tools ncurses libpng zlib libusb libusb-compat readline inetutils
```

```
schedtool gperf imagemagick lzop pngcrush rsync gcc-multilib gcc-libs-multilib libtool-multilib lib32-libusb lib32-readline lib32-glibc bash-completion lib32-zlib &&
yay -Sy --needed lib32-libusb-compat android-sdk android-sdk-platform-tools android-udev esd-oss pngcrush repo tcp_wrappers termcap perl-switch git-lfs
```

```
# Install dependencies
sudo apt-get install git-core gnupg flex bison build-essential zip curl zlib1g-dev libc6-dev-i386 x11proto-core-dev libx11-dev lib32z1-dev libgl1-mesa-dev libxml2-utils xsltproc unzip fontconfig

# For version 16.04/18.04:
sudo apt install libwxgtk2.8-dev

# For version 20.04:
sudo apt install libwxgtk3.0-dev

# For versions between 20.10 and 23.04:
sudo apt install lib32ncurses5-dev libncurses5 libncurses5-dev

# Install a few dependencies manually ONLY if on 23.10 or higher:
wget http://archive.ubuntu.com/ubuntu/pool/universe/n/ncurses/libtinfo5_6.4-2_amd64.deb && sudo dpkg -i libtinfo5_6.4-2_amd64.deb && rm -f libtinfo5_6.4-2_amd64.deb
wget http://archive.ubuntu.com/ubuntu/pool/universe/n/ncurses/libncurses5_6.4-2_amd64.deb && sudo dpkg -i libncurses5_6.4-2_amd64.deb && rm -f libncurses5_6.4-2_amd64.deb
```

```
# Install WSL/Ubuntu WSL Container
wsl --install -d Ubuntu
wsl --set-version 2
```

```
# Install dependencies
sudo apt-get install git-core gnupg flex bison build-essential zip curl zlib1g-dev libc6-dev-i386 x11proto-core-dev libx11-dev lib32z1-dev libgl1-mesa-dev libxml2-utils xsltproc unzip fontconfig
```

```
nano ~/.profile
```

```
# set PATH so it includes user's private bin if it exists
if [ -d "$HOME/bin" ] ; then
```

```
    PATH="$HOME/bin:$PATH"
fi
```

```
git config --global user.email "you@example.com"
git config --global user.name "Your Name"
```

you@example.com

```
export REPO=$(mktemp /tmp/repo.XXXXXXXXX)
curl -o ${REPO} https://storage.googleapis.com/git-repo-downloads/repo
gpg --recv-keys 8BB9AD793E8E6153AF0F9A4416530D5E920F5C65
curl -s https://storage.googleapis.com/git-repo-downloads/repo.asc | gpg --verify - ${REPO} && install -m 755 ${REPO} ~/bin/repo
```

```
sudo apt install repo
sudo apt install repo
```

```
## Install homebrew
/bin/bash -c "$(curl -fsSL
https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"

# Install repo with homebrew
brew install repo
```

```
export USE_CCACHE=1
export CCACHE_EXEC=/usr/bin/ccache
```

```
ccache -M [Insert size you want here]G
ccache -o compression=true
```

```
repo init -u https://github.com/LineageOS/android.git -b lineage-21.0 --git-lfs
```

```
repo init -u https://github.com/LineageOS/android.git -b lineage-20.0 --git-lfs
```

```
repo sync -c -j[insert number of threads you have minus 4]
```

```
[wsl2]

memory=[how much ram you have -6 gigs]GB
swap=[whatever size you want to make your ram + swap = at least 64Gig.]
swapFile=E:\\swap.vhdx
```

```
source build/envsetup.sh
```

```
breakfast oriole
```

Discord Matrix

```
./extract-files.sh
```

```
croot
brunch oriole
```

```
adb reboot fastboot && fastboot flasing unlock
```

```
fastboot flash boot[path to where you downloaded the boot.img] && \
fastboot flash dtbo.img[path to where you downloaded the dtbo.img] && \
fastboot flash vendor_boot.img[path to where you downloaded vendor_boot.img]
&& \
fastboot reboot recovery
```

```
adb sideload [drop the lineageos .zip file here]
```

this

```
adb sideload [path to the nikgapps.zip file]
```