

The two tasks given to me to perform after the 3rd round of interview were -

1. To install Arch Linux Operating System in a Virtual Box.
2. To understand the steps of complete execution of a code in Java.

I have completed both the tasks, have tried my best to explain the working and my understanding of both the tasks below.

Task 1 - Arch Linux Installation -

To install a new Operating System with the existing one from scratch appears to be like a death wish for many beginners. However, what must be done has to be done, so this way, my quest to understand the installation and working of a new Operating System named Arch Linux started.

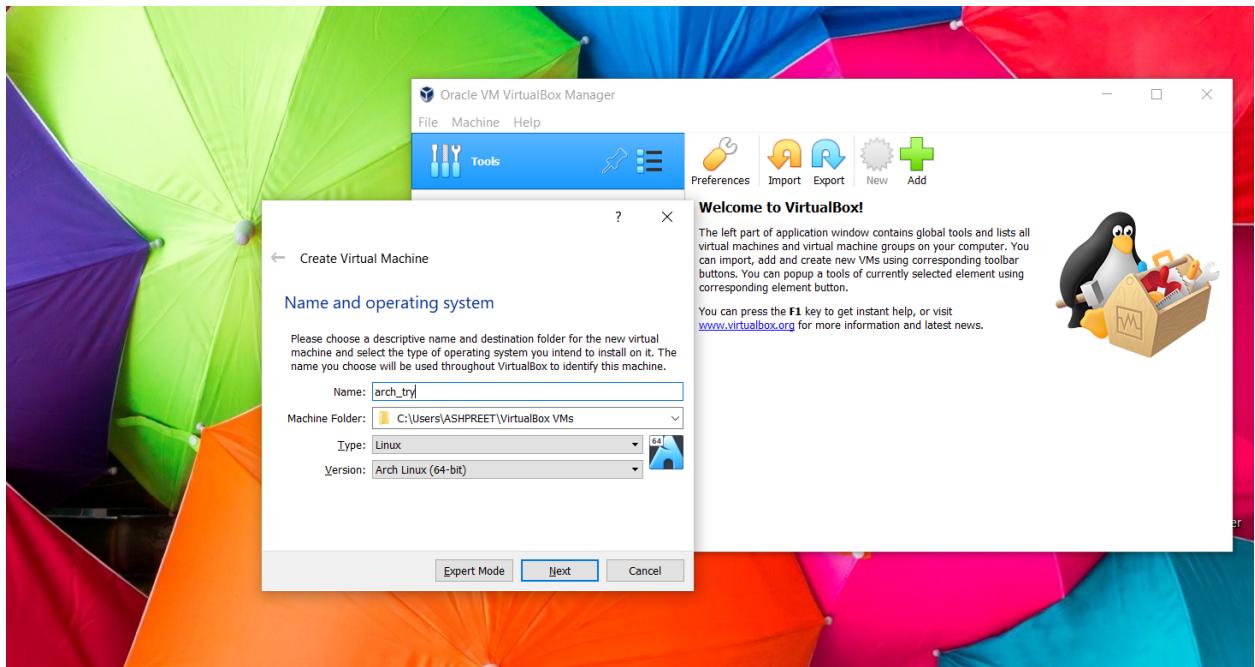
The installation of Arch Linux could be complex and time-consuming for the first time, but it was all worth my time and effort. I have installed it in a virtual box, tested it, and made sure that it runs perfectly.

I referred to a few documents, which helped me a lot in completing this task I shared in the reference section of this doc. Following are the steps which I followed for the installation.

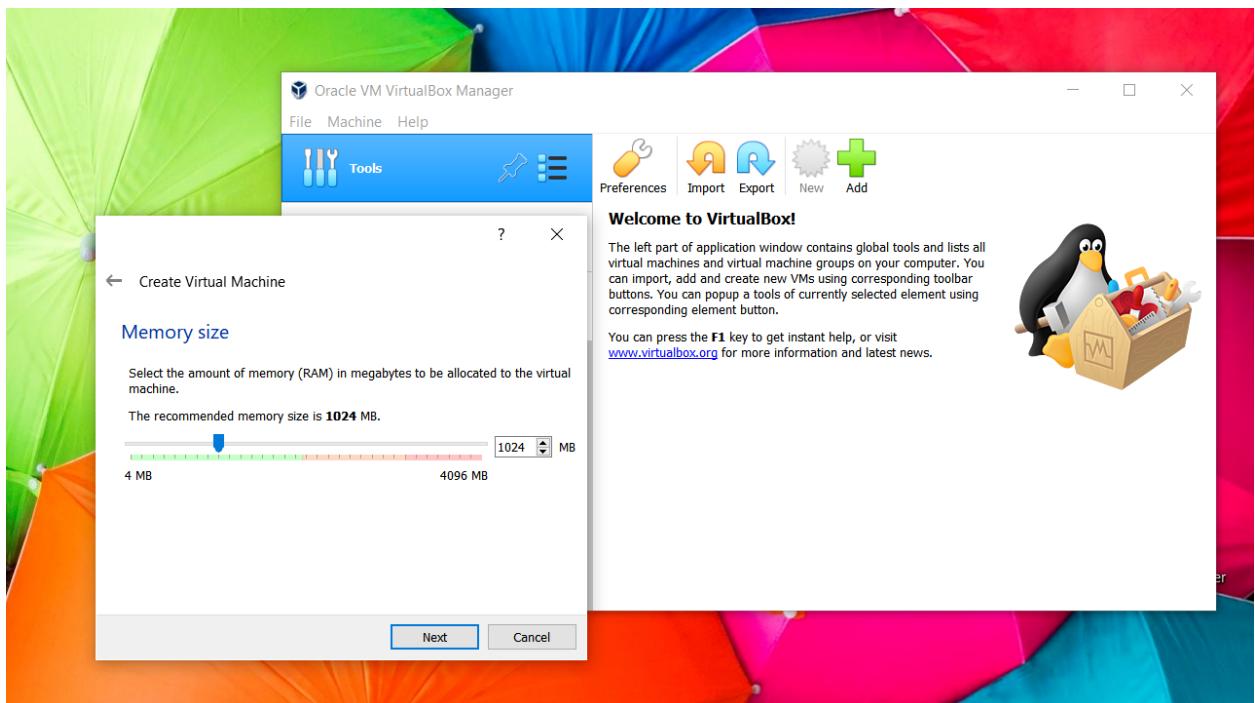
[**Note** - We have to make sure that we have minimum 1 GB of RAM and 20 MB of free disk space before starting the installation process.]

Virtual Box Configuration and settings

1. Firstly we should install Oracle VM virtual box 6.1; the reason for selecting this virtual box is its cross-platform powerful virtualization system, and it also supports a wide range of guest Operating System platforms.
2. As per the image(step 1) given on the next page, we can provide a base name to our machine and select the type and version of the Operating system after selecting the tools section.
3. As per our laptop/desktop's memory size, we can select the RAM size, disk type, disk size, etc. that we want to give our Operating system, but we have to make sure that it does not enter the red zone area as mentioned in the take care image in the next page.

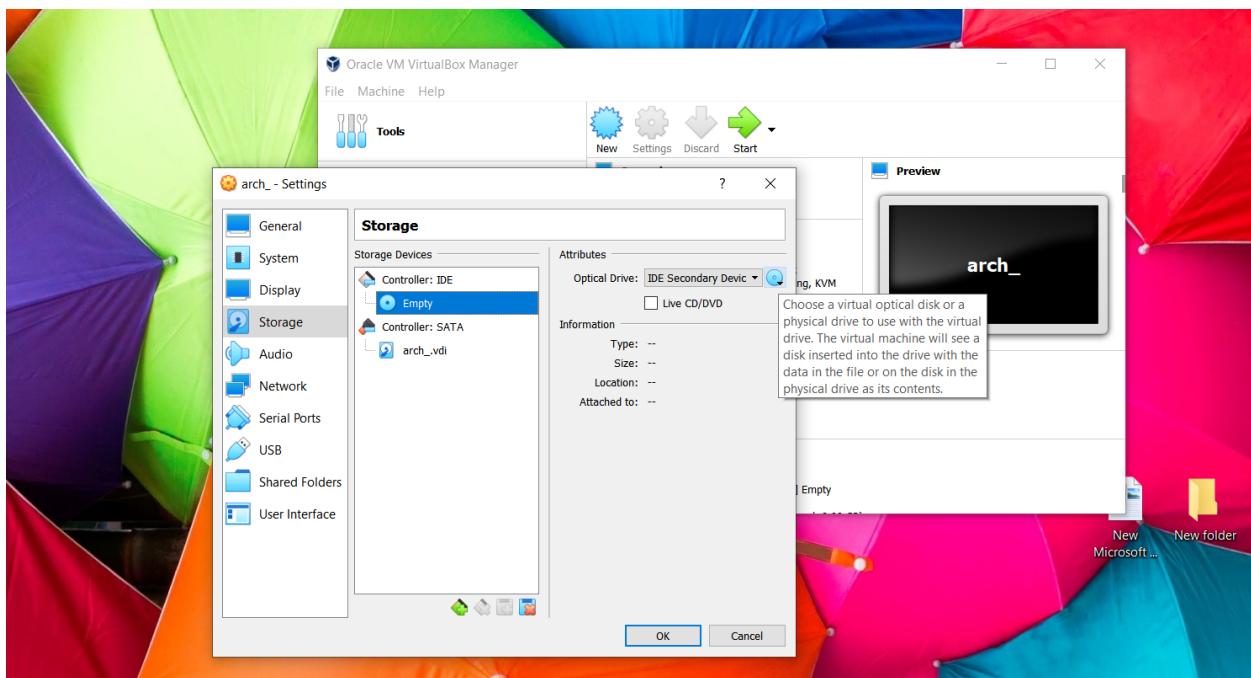


Step 1



Take care to not enter red zone

- After completing the initial configuration, we can visit the website <https://archlinux.org/download/> and then go to the nearest region of our area to download the x86_64 iso file . We can download archlinux-2021.05.01-x86_64.iso from <http://mirror.cse.iitk.ac.in/archlinux/iso/2021.05.01/> page .
- After getting the iso file , we should make few changes in the settings section of OS. In the storage section upload the iso image file in the empty section of the disk as shown in the below image -

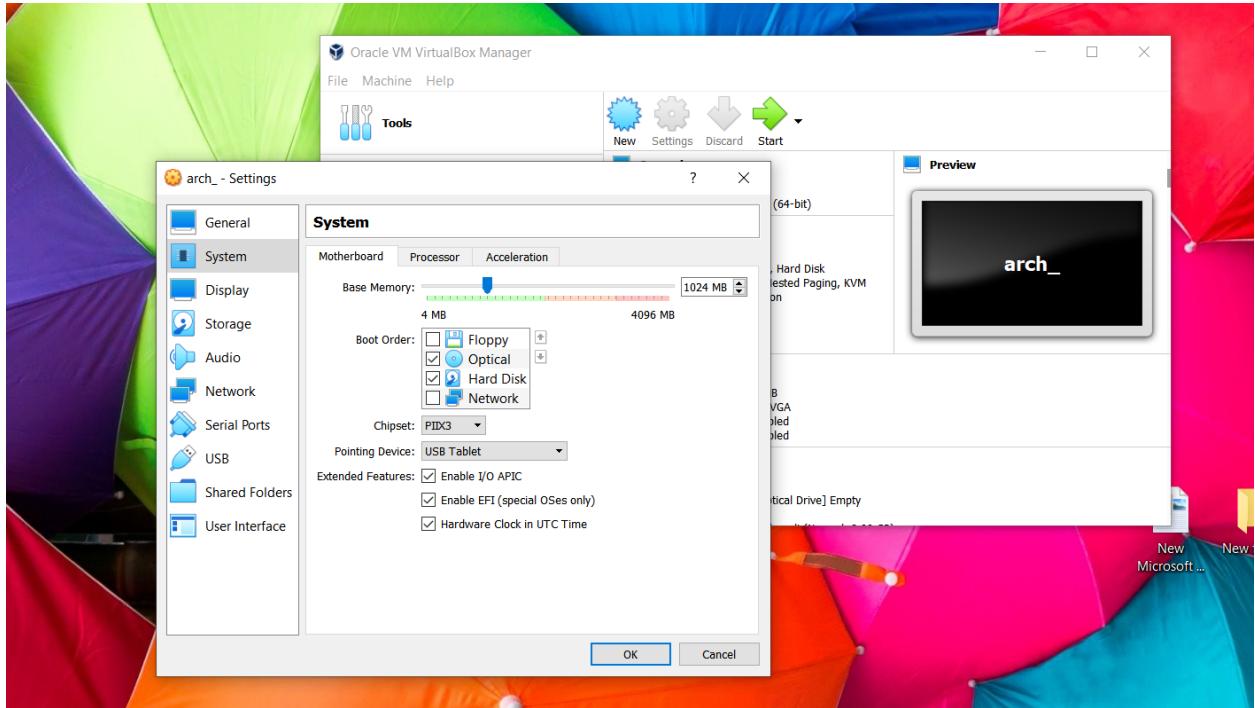


- In the Systems tab we should select the option of enable EFI to initiate booting process of ArchLinux using UEFI as shown in below image -

[Note -

- The process of booting in any operating system can be initiated in both BIOS as well as in UEFI mode, the main difference is the data about initialisation and startup in BIOS is stored in firmware chip on motherboard while in UEFI it is stored in .efi file.
- The .efi file is stored in the special EFI System Partition and to make it work we have selected the option of 'Enable EFI' present in the next page and unselected the floppy option.

- It supports driver size upto 9 zettabytes and provides faster boot time and most of the hardware these days supports EFI . If we want to emulate with newer BIOS firmware then it's a good practice to enable EFI option and test with EFI firmware.]



Challenges faced with virtual box

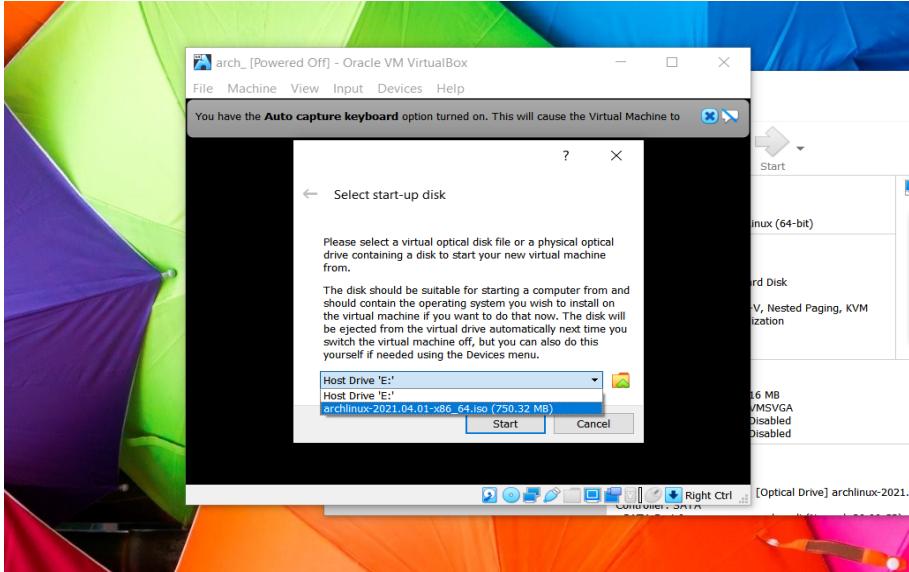
After installing the virtual box, while selecting the options for version of OS sometimes it not gives the option of 64 bit version for the guest Operating System platforms , there could be various reasons for this issue , but the reason for my issue of not showing 64 bit option was disabling of 'virtualisation' option in BIOS. After enabling it and restarting the laptop , the option of 64 bit got enabled in the virtual box as well.

Starting The Installation Procedure

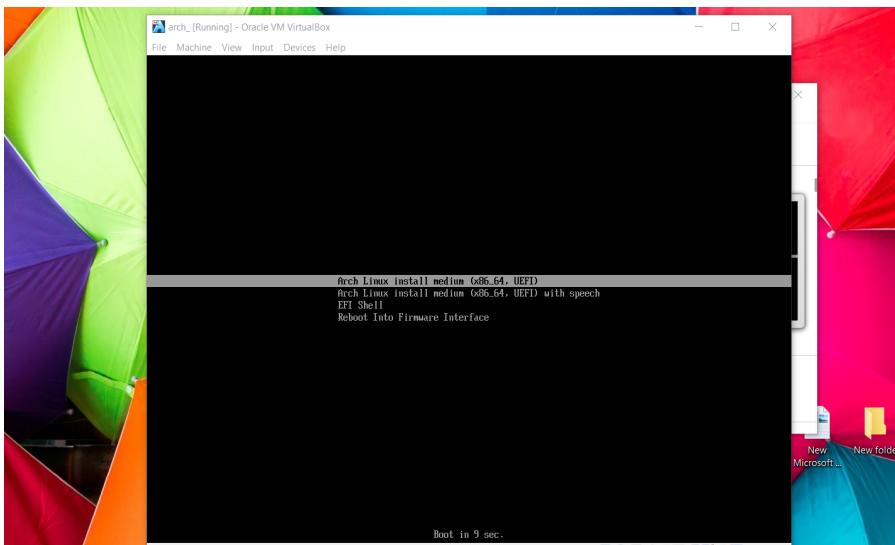
Once the virtual box settings are done we can start our installation procedure by clicking on the start button present in the above image.

1. In the first step of starting the installation we might get a pop up box to select among various disk options in the drop down menu , here as per our requirement we should select the iso image file as given in step1 image of next page.

2. After selecting the iso file we should select the option of ‘Boot Arch Linux install medium (x86_64)’ (for 64 bit) so that system will start temporary booting and then it move into the live version and we can get logged in as a root user as shown in step2 image in the next page.



Step: 1



Step: 2

3. In this step the first thing we should confirm having an internet connection so try to ping archLinux.org to know the result as shown in the next step

```

arch_ [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Arch Linux 5.11.11-arch1-1 (tty1)
archiso login: root (automatic login)

To install Arch Linux follow the installation guide:
https://wiki.archlinux.org/index.php/Installation_guide

For Wi-Fi, authenticate to the wireless network using the iwctl utility.
For mobile broadband (WWAN) modems, connect with the mmcli utility.
Ethernet, WLAN and WWAN interfaces using DHCP should work automatically.

After connecting to the internet, the installation guide can be accessed
via the convenience script Installation_guide.

root@archiso ~ # ping archlinux.org
PING archlinux.org (95.217.163.246) 56(84) bytes of data.
64 bytes from archlinux.org (95.217.163.246): icmp_seq=1 ttl=53 time=534 ms
64 bytes from archlinux.org (95.217.163.246): icmp_seq=2 ttl=53 time=206 ms
64 bytes from archlinux.org (95.217.163.246): icmp_seq=3 ttl=53 time=205 ms
64 bytes from archlinux.org (95.217.163.246): icmp_seq=4 ttl=53 time=206 ms
64 bytes from archlinux.org (95.217.163.246): icmp_seq=5 ttl=53 time=206 ms
64 bytes from archlinux.org (95.217.163.246): icmp_seq=6 ttl=53 time=206 ms

```

Step: 3

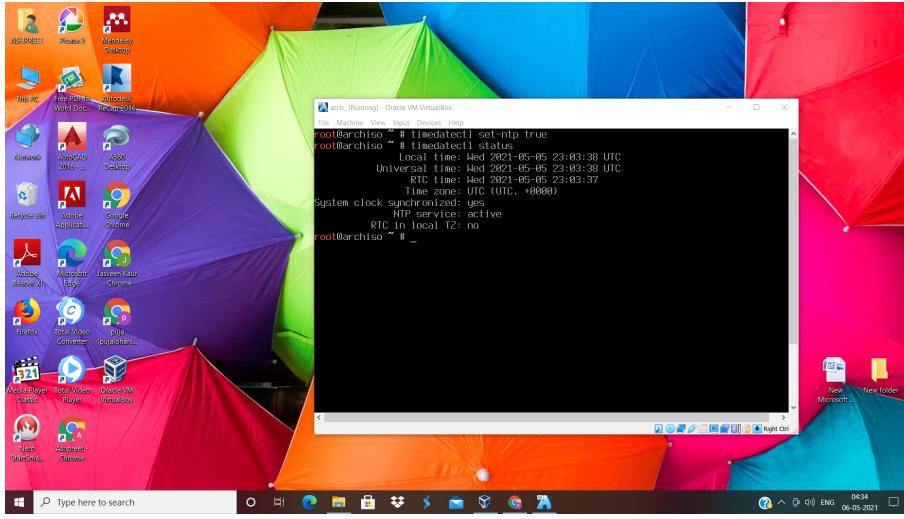
- Inorder to make sure that the system is booted to the the UEFI mode and we can go on installing in EFI based BIOS, use the command **ls /sys/firmware/efi/efivars/** to get the below output -

```

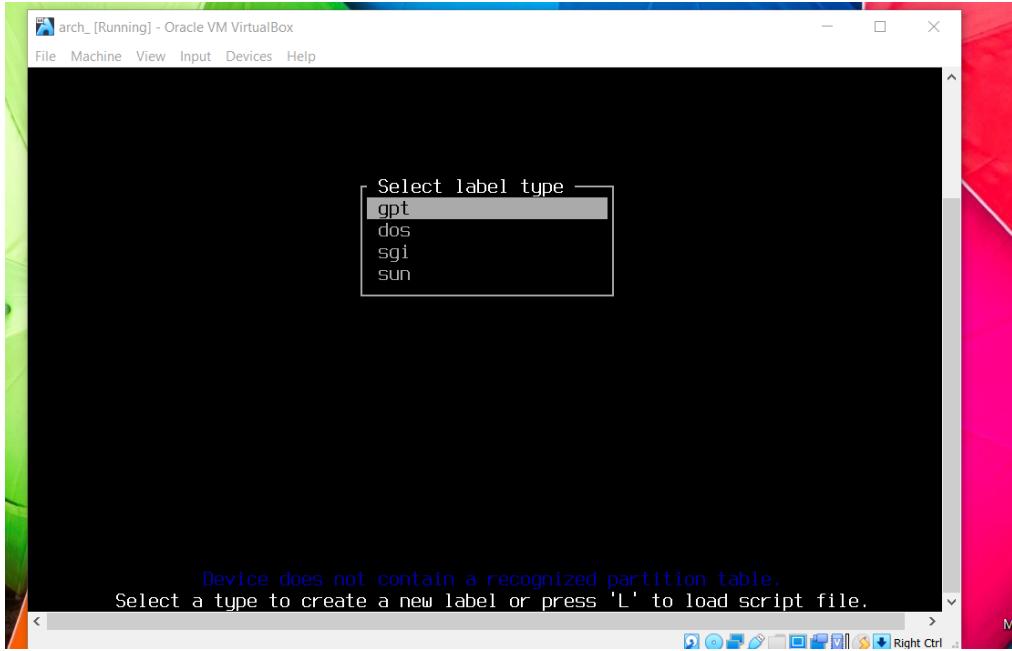
arch_ [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
FirmwareFeaturesMask-4d1ede05-38c7-4a6a-9cc6-4bccaa8b38c14
FirmwareFeatures-4d1ede05-38c7-4a6a-9cc6-4bccaa8b38c14
Key0000-8be4df61-93ca-11d2-aa0d-00e098032b8c
Key0001-8be4df61-93ca-11d2-aa0d-00e098032b8c
LangCodes-8be4df61-93ca-11d2-aa0d-00e098032b8c
Lang-8be4df61-93ca-11d2-aa0d-00e098032b8c
LoaderEntries-4a67b082-0a4c-41cf-b6c7-440b29bb8c4f
LoaderEntrySelected-4a67b082-0a4c-41cf-b6c7-440b29bb8c4f
LoaderFeatures-4a67b082-0a4c-41cf-b6c7-440b29bb8c4f
LoaderFirmwareInfo-4a67b082-0a4c-41cf-b6c7-440b29bb8c4f
LoaderFirmwareType-4a67b082-0a4c-41cf-b6c7-440b29bb8c4f
LoaderImageIdentifier-4a67b082-0a4c-41cf-b6c7-440b29bb8c4f
LoaderInfo-4a67b082-0a4c-41cf-b6c7-440b29bb8c4f
LoaderTimeExecUsec-4a67b082-0a4c-41cf-b6c7-440b29bb8c4f
LoaderTimeInitUsec-4a67b082-0a4c-41cf-b6c7-440b29bb8c4f
LoaderTimeMenuUsec-4a67b082-0a4c-41cf-b6c7-440b29bb8c4f
MTC-eb70d011-1402-11d3-8e77-00a0c969723b
OsIndicationsSupported-8be4df61-93ca-11d2-aa0d-00e098032b8c
PlatformLangCodes-8be4df61-93ca-11d2-aa0d-00e098032b8c
PlatformLang-8be4df61-93ca-11d2-aa0d-00e098032b8c
PlatformRecoverJob000-8be4df61-93ca-11d2-aa0d-00e098032b8c
Timeout-8be4df61-93ca-11d2-aa0d-00e098032b8c
VarErrorFlag-04037fe8-f6ae-48b0-bdd5-3798c5e89aa
boot-args-7c436110-ab2a-4bb-a800-fe41995c9782
root@archiso ~ #

```

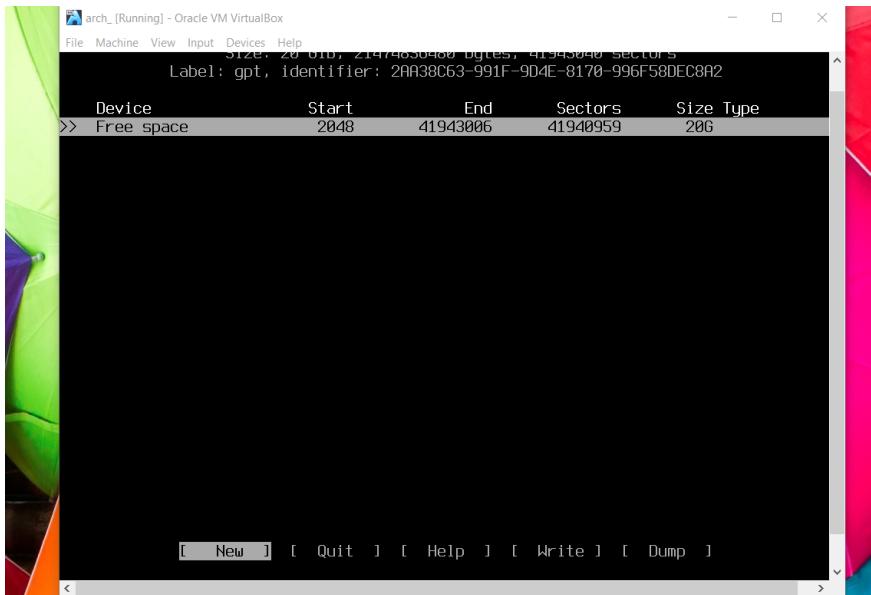
- Since we are connected to the internet we can set the time date by using the command **timedatectl set-ntp true** which will help us to get correct time and date from the server and we can check the status by the command **timedatectl status** as shown in the below image



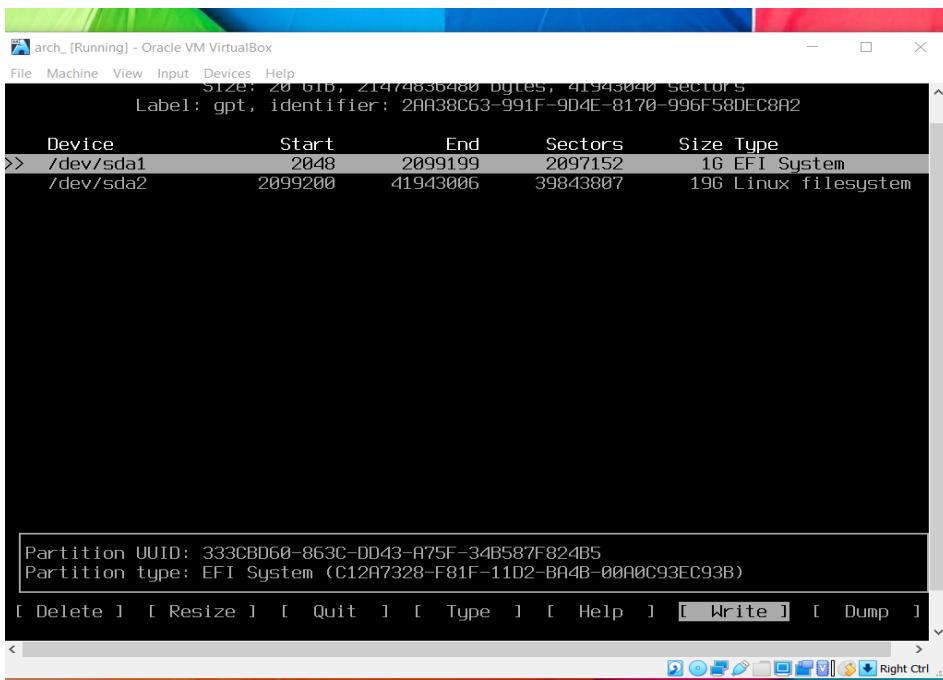
- Following the above step we can set up our disks by using the command **fdisk**, after this since we don't have anything on the current disk setup it'll ask to select a label type as shown below



- So next we will go forward by selecting the option of 'gpt' and it'll show the free space as per the disk space allocated by us during the configuration of Virtual box. I had selected 20 GB of space so can see free space as 20 GB with the start and end of disk sectors

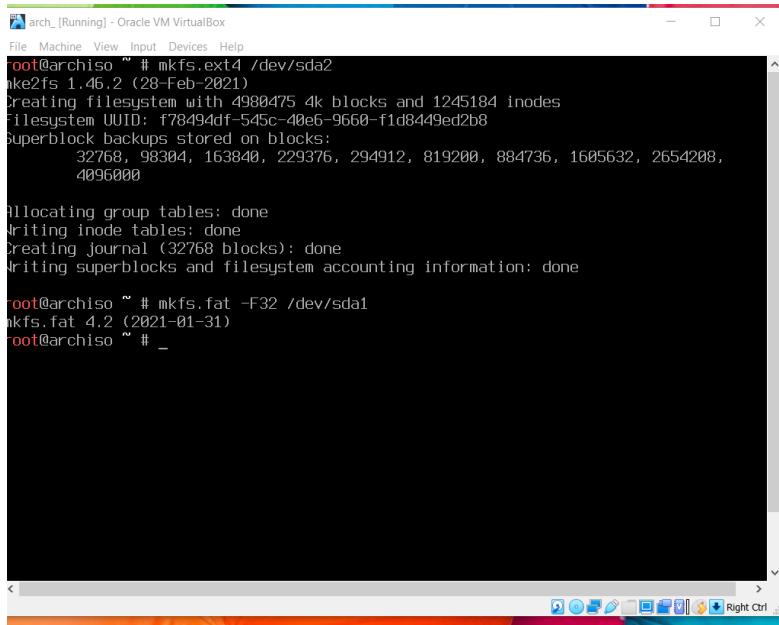


- After selecting the 'new' option in the bottom we can modify our space of the disk as per our requirement so we'll allocate 1 GB to the EFI partition and rest 19 GB to the linux root file system and after completing this we'll type 'yes' in the 'Write' section to save our changes and once everything is done we can select the 'Quit' option to save the changes.



- Now as we have created the two partitions we'll format them now by using the command

mkfs.ext4 /dev/sda2 (since sda2 is root file partition) and **mkfs.fat -F32 /dev/sda1** this means we are creating File Allocation Table for the location of EFI partition of disk , the output for this will look like



The screenshot shows a terminal window titled "arch_ [Running] - Oracle VM VirtualBox". The terminal contains the following command history:

```
root@archiso ~ # mkfs.ext4 /dev/sda2
mke2fs 1.46.2 (28-Feb-2021)
Creating filesystem with 4980475 4k blocks and 1245184 inodes
Filesystem UUID: f78494df-545c-40e6-9660-f1d8449ed2b8
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done

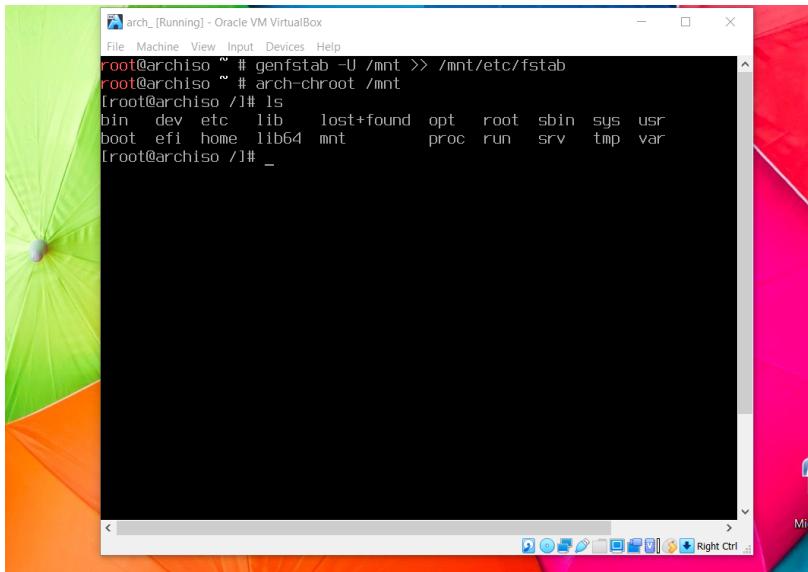
root@archiso ~ # mkfs.fat -F32 /dev/sda1
mkfs.fat 4.2 (2021-01-31)
root@archiso ~ # _
```

10. Since now we have formatted the file system , the next step will be to mount both the disk partitions , so we can do it by using the following command -
mount /dev/sda2 /mnt (here mnt is the directory where we are trying to mount sda2)
mount /dev/sda1 /mnt/efi (make sure there is a directory efi inside mount , if not then create one and then mount it) .

[**Note** - Mounting is the process by which the Operating System makes files and directories in a storage device. The location in the Virtual File System (VFS) where a newly mounted system has been registered is called a mount point.]

11. From here it'll start getting more fun as we'll use pacstrap inorder to get the base package, the base kernel and firmware. We can do this by
pacstrap /mnt base linux linux-firmware (this means we are installing the base package , linux package and linux firmware in mnt directory because we have just now mounted our linux root filesystem here). This process takes some time to get completed depending upon our internet speed.
12. Once these packages are installed we'll move forward by the command
genfstab -U /mnt >> /mnt/etc/fstab, here we are generating fstab from the mnt directory to the mnt/etc/fstab directory .
[**Note** - Linux file system (fstab) is a configuration table that is designed to ease the burden of mounting and unmounting file systems to a machine].

13. In this step we'll move to the root file system which we have mounted in previous step by using the command **arch-chroot /mnt** and then by using the command **ls** we can see all the different directories located in the root file system that we have just created.



```
arch_ [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@archiso ~ # genfstab -U /mnt >> /mnt/etc/fstab
root@archiso ~ # arch-chroot /mnt
[root@archiso /]# ls
bin dev etc lib lost+found opt root sbin sys usr
boot efi home lib64 mnt proc run srv tmp var
[root@archiso /]# _
```

14. In the next step we can set up our timezone by using the command
In -sf /usr/share/zoneinfo/ , now by using the 'tab' key we can get all the possible zones from which we'll select Asia and after this we can select our region in that zone , in my case I selected Kolkata and then we'll use the command
In -sf /usr/share/zoneinfo/Asia/Kolkata /etc/localtime which will link specific timezone to local time .

```

arch_ [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Arctic/ EST5EDT Greenwich MST ROK posix/
Asia/ Egypt HST MST7MDT Singapore posixrules
Atlantic/ Eire Hongkong Mexico/ Turkey right/
Australia/ Etc/ Iceland NZ UCT tzdata.zi
Brazil/ Europe/ Indian/ NZ-CHAT US/ zone.tab
CET Factory Iran Navajo UTC zone1970.tab
CST6CDT GB Israel PRC Universal
Canada/ GB-Eire Jamaica PST8PDT W-SU
Chile/ GMT Japan Pacific/ WET
[root@archiso /]# ln -sf /usr/share/zoneinfo/Asia/
Aden Chongqing Jerusalem Novokuznetsk Tashkent
Almaty Chungking Kabul Novosibirsk Tbilisi
Amman Colombo Kamchatka Omsk Tehran
Anadyr Dacca Karachi Oral Tel_Aviv
Aqtau Damascus Kashgar Phnom_Penh Thimbu
Aktobe Dhaka Kathmandu Pontianak Thimphu
Ashgabad Dili Katmandu Pyongyang Tokyo
Ashkhabad Dubai Khandyga Qatar Tomsk
Atyrau Dushanbe Kolkata Qostanay Ujung_Pandang
Baghdad Famagusta Krasnoyarsk Qyzylorda Ulaanbaatar
Bahrain Gaza Kuala_Lumpur Rangoon Ulan_Bator
Baku Harbin Kuching Riyadh Urumqi
Bangkok Hebron Kuwait Saigon Ust-Nera
Barnaul Ho_Chi_Minh Macao Sakhalin Vientiane
Beirut Hong_Kong Macau Samarkand Vladivostok
Bishkek Hovd Magadan Seoul Yakutsk
Brunei Irkutsk Makassar Shanghai Yangon
Calcutta Istanbul Manila Singapore Yekaterinburg
Chita Jakarta Muscat Srednekojamsk Yerevan
Choibalsan Jayapura Nicosia Taipei
[root@archiso /]# ln -sf /usr/share/zoneinfo/Asia/Kolkata /etc/localtime
[root@archiso /]#

```

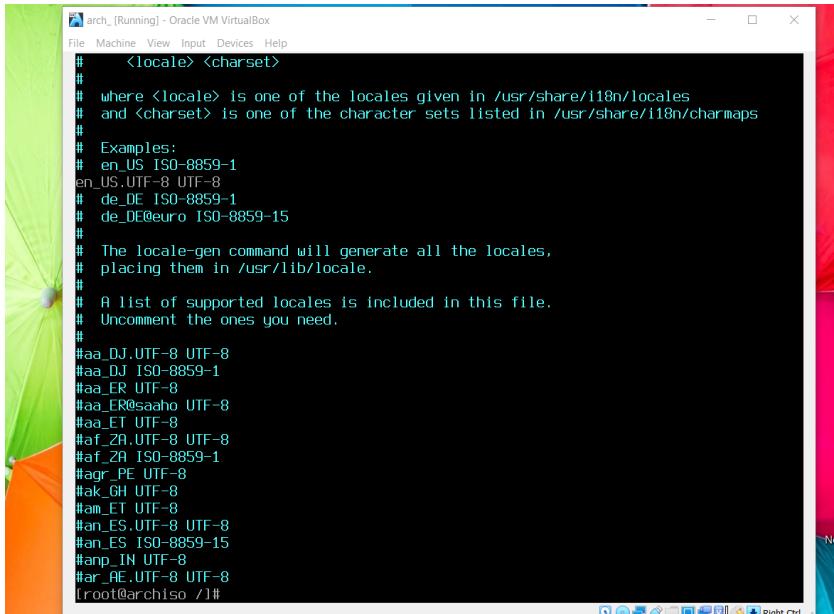
15. We can synchronise the hardware clock of the system by using the command **hwclock --systohc**.
16. We can install the package for vim by using the command **pacman -Syu vim** as shown in the below image -

```

arch_ [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
[root@archiso /]# hwclock --systohc
[root@archiso /]# pacman -Syu vim
:: Synchronizing package databases...
core           130.7 KiB  341 KiB/s 00:00 [########################################] 100%
extra          1601.6 KiB 3.87 MiB/s 00:00 [########################################] 100%
community      0.0   B  0.00   B/s 00:00 [----]  0%

```

17. Now we'll open the file `locale.gen` by using the command **vim/etc/locale.gen** in order to use the comfortable character set by uncommenting that in `locale.gen` file , here we have uncommented the character set `en_US.UTF-8` `UTF-8` and will save the changes by :x .



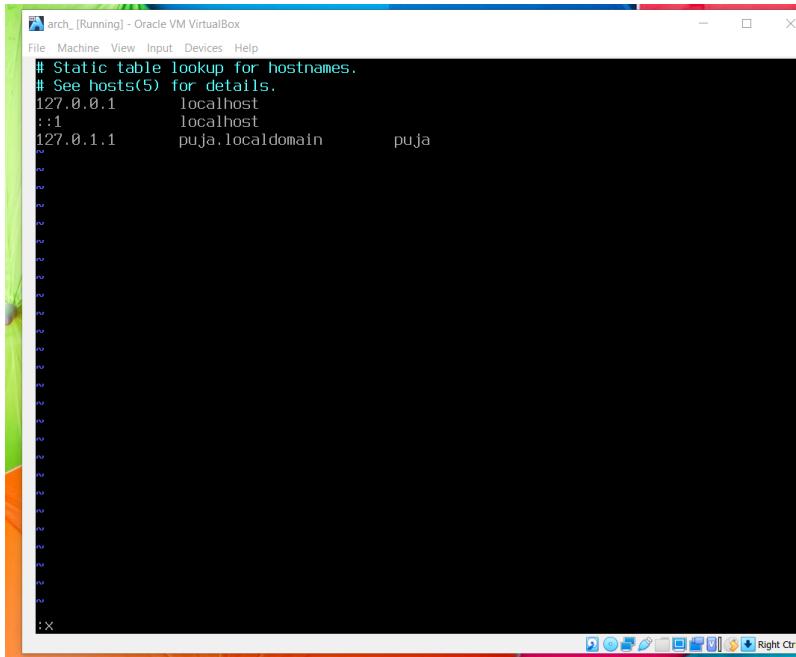
```
# locale-gen
# where <locale> is one of the locales given in /usr/share/i18n/locales
# and <charset> is one of the character sets listed in /usr/share/i18n/charmaps
#
# Examples:
# en_US ISO-8859-1
en_US.UTF-8 UTF-8
# de_DE ISO-8859-1
# de_DEeuro ISO-8859-15
#
# The locale-gen command will generate all the locales,
# placing them in /usr/lib/locale.
#
# A list of supported locales is included in this file.
# Uncomment the ones you need.
#
#aa_DJ.UTF-8 UTF-8
#aa_DJ ISO-8859-1
#aa_ER.UTF-8
#aa_ER@saaho UTF-8
#aa_ET.UTF-8
#af_ZA.UTF-8 UTF-8
#af_ZA ISO-8859-1
#agr_PE.UTF-8
#ak_GH.UTF-8
#am_ET.UTF-8
#an_ES.UTF-8 UTF-8
#an_ES ISO-8859-15
#anp_IN.UTF-8
#ar_AE.UTF-8 UTF-8
[root@archiso ~]#
```

18. In this step we are going to create a new hostname file by using the command **vim /etc/hostname** and then type the hostname inside the file as given below -



```
puja
~
```

19. Now the next step is to open the hosts file by using the command **vim /etc/hosts** and will add few lines as given below -



The screenshot shows a terminal window titled "arch_ [Running] - Oracle VM VirtualBox". The window displays the contents of the /etc/hosts file. The file contains the following entries:

```
# Static table lookup for hostnames.
# See hosts(5) for details.
127.0.0.1      localhost
::1            localhost
127.0.1.1      puja.localdomain    puja
```

After exiting from this file we can set up the root password by **passwd** command.

20. Now we'll create a user who is non root by using the command

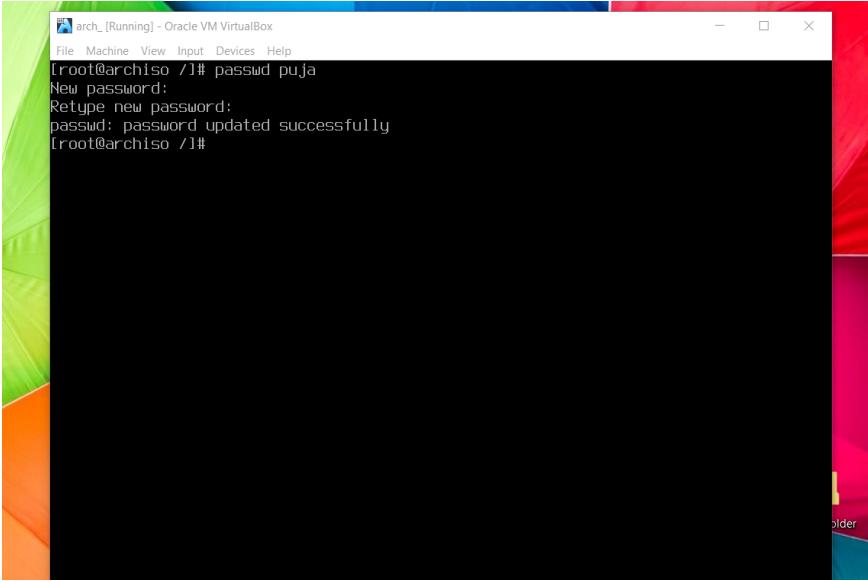
useradd -G wheel,audio,video -m puja (here wheel, audio, video are the access permissions given to the user 'puja' and -m will create a new home directory for the user named puja).

21. We'll download some more packages by using the the command

pacman -Syu netctl and **pacman -Syu dhcpcd** which will help us to connect to the network once we are finished installing Arch Linux base system. If we are using some wireless device like a dongle for the network then we need some additional packages whose link I'll mention in the reference section of the doc.

22. In this step we'll create a password for the new user we created by using the command

passwd puja (puja is the new username created a few steps back).



The screenshot shows a terminal window titled "arch_ [Running] - Oracle VM VirtualBox". The window contains the following text:

```
[root@archiso ~]# passwd puja
New password:
Retype new password:
passwd: password updated successfully
[root@archiso ~]#
```

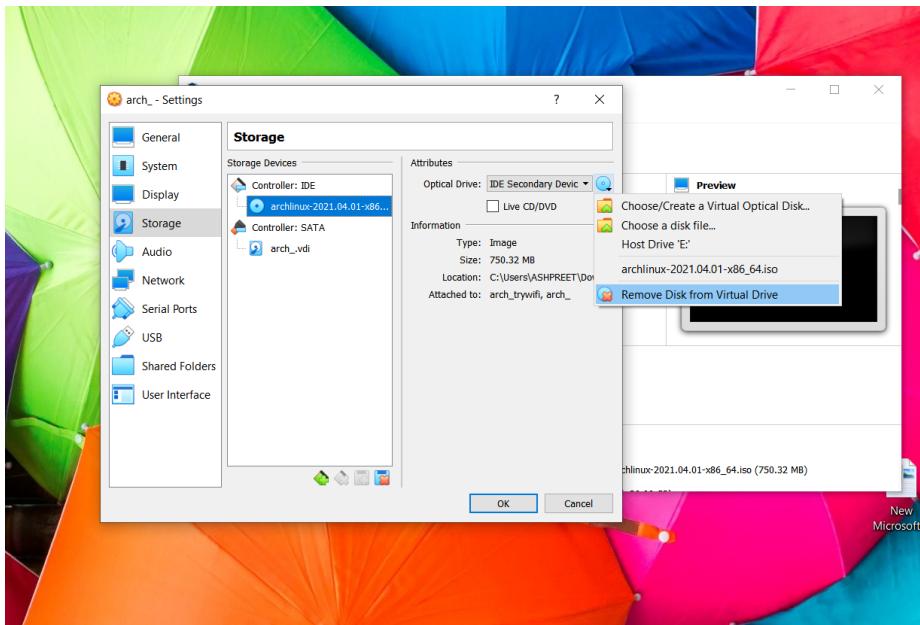
23. We'll use few more packages like grub and efibootmgr by using the command
pacman -Syu grub efibootmgr .

[Note - pacman is the default package manager for arch while pacstrap is a tool for arch installation which should not be used in installed system]

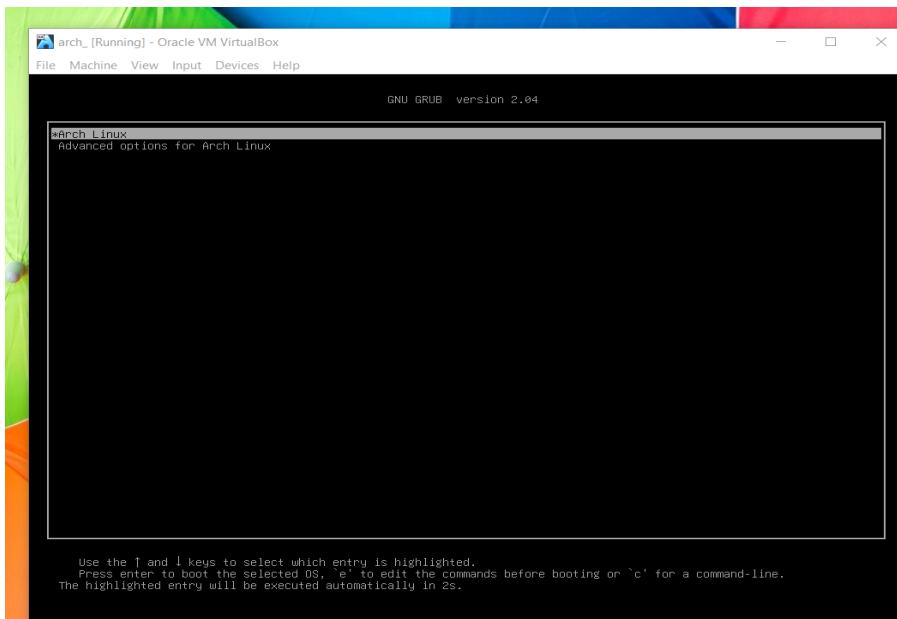
24. Now since we have the grub package we can install it by using the command
grub-install --target=x86_64-efi --efi-directory=/efi/ --bootloader-id= Arch.
25. After this we'll make a grub config file and output it to the boot/grub/grub.cfg file by running the command **grub-mkconfig -o /boot/grub/grub.cfg** .

[Note - a grub is a complete program that is used for loading and managing the boot process. It is the most common bootloader for many linux distributions.]

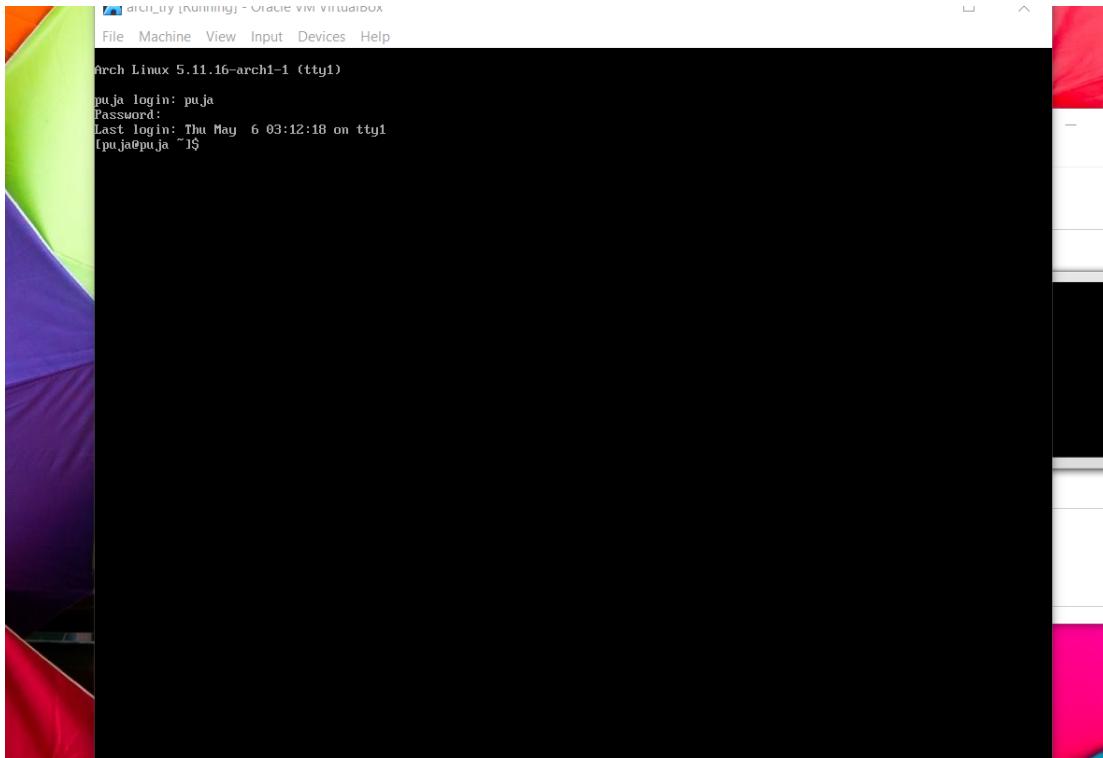
26. Now at this point we are ready to restart the system inorder to boot into our new Arch Linux platform. We can come out of the mounted root directory and then shutdown the system by using the **exit** and **shutdown now** commands respectively.
27. Now we are ready to go back to our virtual box settings section and can remove the iso image from the storage part as shown below -



28. Now we can go back on the system by clicking on the 'start' option and will come across the below screen



Now after selecting the option of Arch Linux we'll get the below screen to enter the username and password to start the system -



Reference -

https://wiki.archlinux.org/title/installation_guide

https://wiki.archlinux.org/title/Network_configuration/Wireless

<https://www.howtoforge.com/tutorial/install-arch-linux-on-virtualbox/>

Task 2 -

The purpose of this task was to get a deep understanding of the execution of a java program. Since java is a platform-independent language, it involves a two-step process, in which the first step is a compilation by an OS-independent compiler. In contrast, the second step is execution via JVM (Java Virtual Machine), which provides a runtime environment to the java code/applications.

Phase 1 (compilation phase) -

- Firstly the programmer uses an editor to write a java program and save it using the '.java' extension.
- The programmer gives the **javac** command to compile the .java file, and then this file is converted into byte code (a language understood by JVM) if the program is error-free.

- In order to convert the source code to bytecode, steps like parsing (to convert the token sequence to syntax trees), entering symbols to the symbol table, name resolution, type checking, constant folding, data flow analysis, and generation of .class files are performed by the compiler.
- The compilation process translates the java program to a sequence of commands that can be directly executed by our computer.

Phase 2 (execution)-

- The class file generated by the compiler is independent of any Operating System, which allows them to be run on any system.
- The .class file can be loaded from the hard disk or from the network to the main memory. This process is performed by the class loader, which contains the bytecodes.
- The verifier checks whether the .class file is loaded with invalid or security breaching instructions; if not, then the JVM interprets and runs the program.
- The execution of the program is done through the command `java classname`.