

Create partition and file system:

- partition
- create filesystem
- make swap space

block device(harddrive):

harddrive provide spaces .before working with the hard drive we have to divide it into pieces .it can be just one giant piece (means one partition) or it can be divide into multiple pieces (multiple partition).for example we can divide it to four primary partitions.if we want divide it more with extended partition with different size. And after that each partition could be formatted in an way that windows can recognize it another could be formatted just like the linux and so on. Each individual pieces works as a file system.where different data is stored and we can work with it.

To work with the partition we inserted a drive.we can do it physically or if you are on a virtual machine you can add blank drive.

After adding the drive (can be physical can be virtual) we can show the status by this command

=>**sudo fdisk -l**

and to see the block drives we can use the command

=>lsblk

for my computer I have added two virtual drives so the results for my computer is like this

```
[vagrant@localhost ~]$ lsblk
NAME                                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                                8:0    0   9.9G  0 disk
├─sda1                            8:1    0   500M  0 part /boot
├─sda2                            8:2    0   9.4G  0 part
│   ├─centos-root                 253:0    0   8.4G  0 lvm  /
│   └─centos-swap                 253:1    0 1016M  0 lvm
[SWAP]
sdb                                8:16    0   30G  0 disk
sdc                                8:32    0   30G  0 disk
sr0                               11:0    1 1024M  0 rom
sr1                               11:1    1 1024M  0 rom
```

so we have block devices sdb and sdc both 30 gigabytes.and its completely blank. its just a raw disk.

so these are the block devices it has not done any partition yet.the swap partition in the table are work as a virtual memory to support the ram .in case of ram is out of memory its helps ram to not going out of ram.

Create the partition

to create partition in block sdb

the command is:

[vagrant@localhost ~]\$ sudo fdisk /dev/sdb

then to see the command we have to type the 'm'

Command (m for help): m

Command action

- a** toggle a bootable flag
- b** edit bsd disklabel
- c** toggle the dos compatibility flag
- d** delete a partition
- g** create a new empty GPT partition table
- G** create an IRIX (SGI) partition table
- l** list known partition types
- m** print this menu
- n** add a new partition
- o** create a new empty DOS partition table
- p** print the partition table
- q** quit without saving changes
- s** create a new empty Sun disklabel
- t** change a partition's system id
- u** change display/entry units

- v** verify the partition table
- w** write table to disk and exit
- x** extra functionality (experts only)

to create the partition first enter p to print the table to see whether we are in the wrong block. after assuring that we type 'n'
n for new partition

and type p for primary
and give the partition number 1
and press enter for starting from the beginning from the drive
then we want to allocate the size
we have to enter "+<size G/M/K>
and press enter
then press w to save it

Command (m for help): n
Partition number (1-128, default 1): 1
First sector (2048-62914526, default 2048):
Last sector, +sectors or +size{K,M,G,T,P} (2048-62914526, default 62914526): +10G
Created partition 1

Command (m for help): w

The partition table has been altered!

**Calling ioctl() to re-read partition table.
Syncing disks.**

Partition id:

partition id is another important thing by default the partition id is 83 which actually for linux partition .to change it on fdisk we have to type 't' for type and press 'L' for the list of the id .then give the partition number and then type the partition id and after that we type 'w' for write.

For example we need to make the swap partition so we have to apply the following command

**Command (m for help): n
Partition number (2-128, default 2): 2
First sector (20973568-62914526, default 20973568):
Last sector, +sectors or +size{K,M,G,T,P} (20973568-62914526, default 62914526): +4G
Created partition 2**

**Command (m for help): t
Partition number (1,2, default 2): 2
Partition type (type L to list all types): 14
Changed type of partition 'Linux filesystem' to 'Linux swap'**

Command (m for help): p

Disk /dev/sdb: 32.2 GB, 32212254720 bytes, 62914560 sectors

Units = sectors of 1 * 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk label type: gpt

#	Start	End	Size	Type	Name
1	2048	20973567	10G	Linux filesystem	
2	20973568	29362175	4G	Linux swap	

Command (m for help): w

The partition table has been altered!

Calling ioctl() to re-read partition table.

Syncing disks.

Here 14 is used for swap but for mordan system it is 82 it is always good to check the id

to check the status we have to use the lsblk command

[vagrant@localhost ~]\$ lsblk

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
------	---------	----	------	----	------	------------

```
sda          8:0    0 9.9G 0 disk
├─sda1       8:1    0 500M 0 part /boot
├─sda2       8:2    0 9.4G 0 part
│   ├─centos-root 253:0  0 8.4G 0 lvm /
│   └─centos-swap 253:1  0 1016M 0 lvm [SWAP]
sdb          8:16   0 30G 0 disk
├─sdb1       8:17   0 10G 0 part
└─sdb2       8:18   0  4G 0 part
sdc          8:32   0 30G 0 disk
sr0         11:0    1 1024M 0 rom
sr1         11:1    1 1024M 0 rom
```

create file system in the partition :

after creating partition the next thing we have to do is creating file system

to create an ext4 file system in sdb1
the command is

```
[vagrant@localhost ~]$ sudo mkfs.ext4 /dev/sdb1
```

it will make the ext4 file system

to make a swap file system in sdb2 we have to do this command

```
[vagrant@localhost ~]$ sudo mkswap /dev/sdb2
```

we can use the ext2 ext3 xfs and riserfs

the command is

```
sudo mkfs -t ext2 /dev/sdb2  
sudo mkfs -t ext3 /dev/sdb2  
sudo mkfs -t xfs /dev/sdb2  
sudo mkfs -t riserfs /dev/sdb2
```

mounting the drive

after creating the file system we have to mount it on a folder to use it
to mount it

first we have to create a folder then use the command

```
[vagrant@localhost ~]$ sudo mkdir /first_drive  
[vagrant@localhost ~]$ sudo mount /dev/sdb1 /first_drive/  
[vagrant@localhost ~]$ cd /first_drive/  
[vagrant@localhost first_drive]$ ls  
lost+found
```

if we found lost+found directory we can assume that it is successfully added .

E can create file using touch command


```
[vagrant@localhost first_drive]$ sudo touch a1 a2  
[vagrant@localhost first_drive]$ ls  
a1 a2 lost+found
```