

1. Use fdisk -l to locate information about the partition sizes.

```
maiyasser@localhost:~$ sudo fdisk -l
[sudo] password for maiyasser:
Disk /dev/sda: 223.6 GiB, 240857409536 bytes, 468862128 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
Disklabel type: dos
Disk identifier: 0xdeb638e0

Device Boot      Start         End      Sectors  Size Id Type
/dev/sda1 *        2048     104447       102400    50M 7 HPFS/NTFS/exFAT
/dev/sda2          104448    10447999    10434552    49.8G 7 HPFS/NTFS/exFAT
/dev/sda3          10448000    33175999    22728000    108.4G 7 HPFS/NTFS/exFAT
/dev/sda4          33176000    468860927    137084928    65.4G 5 Extended
/dev/sda5          331778048    333006847     1228800    600M ef EFI (FAT-12/16/32)
/dev/sda6          333008896    335186447     2097552    10.8G 83 Linux
/dev/sda7          335188896    468860927    133752832    63.8G 8e Linux LVM

Disk /dev/mapper/rhel-root: 38.5 GiB, 41297117184 bytes, 80658432 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 /dev/mapper/rhel-swap: 6.6 GiB, 7021264896 bytes, 13713408 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 /dev/mapper/rhel-home: 18.8 GiB, 20162019328 bytes, 39378944 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 /dev/sdb: 7.2 GiB, 7757398016 bytes, 15151168 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
Disklabel type: dos
Disk identifier: 0xa7028a0c

Device Boot      Start         End      Sectors  Size Id Type
/dev/sdb1 *         64    15151167    15151104    7.2G c W95 FAT32 (LBA)
maiyasser@localhost:~$
```

2. Use fdisk to add a new logical partition that is 1GB in size.

```
maiyasser@localhost:~$ sudo fdisk /dev/mapper/rhel-root

Welcome to fdisk (util-linux 2.32.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

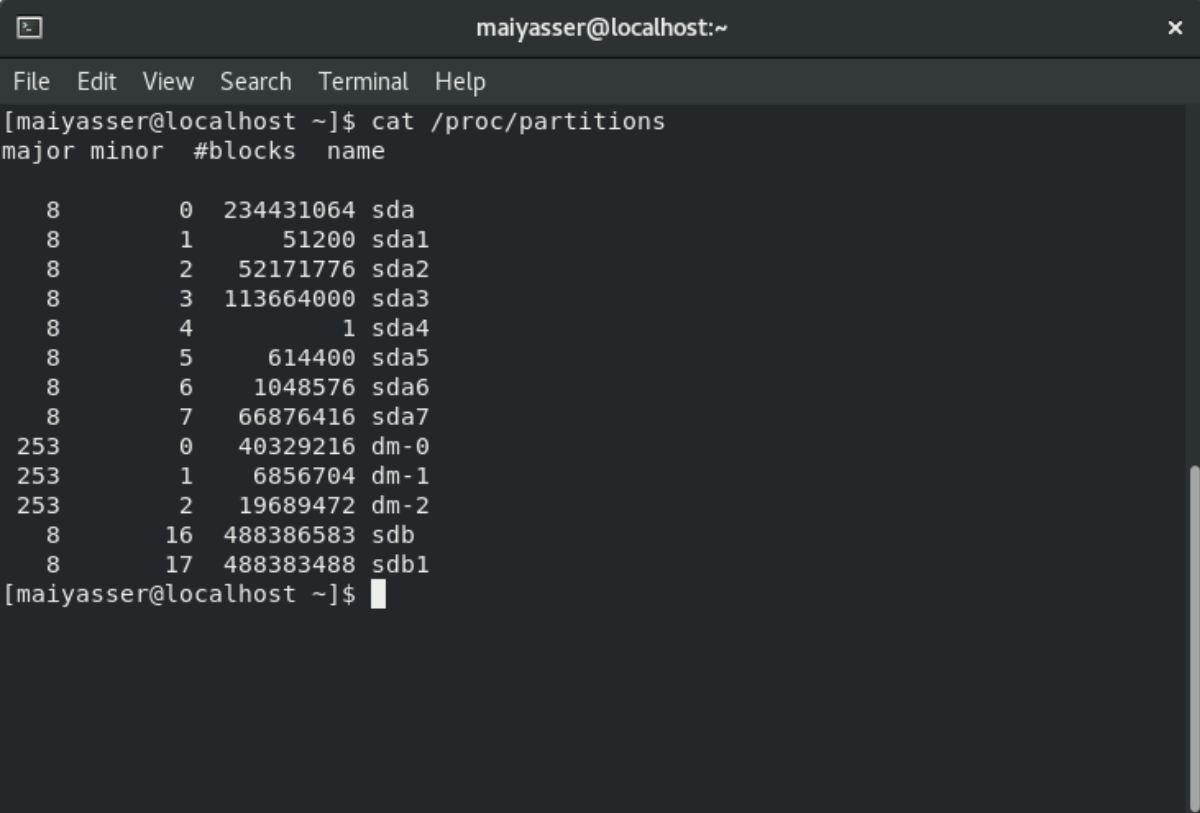
The old xfs signature will be removed by a write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0xda0acc8b.

Command (m for help): n
Partition type
   p   primary (0 primary, 0 extended, 4 free)
   e   extended (container for logical partitions)
Select (default p): e
Partition number (1-4, default 1):
First sector (2048-80658431, default 1):
Last sector, +sectors or +size{K,M,G,T,P} (2048-80658431, default 80658431):

Created a new partition 1 of type 'Extended' and of size 38.5 GiB.
Command (m for help):
```

3. Did the kernel feel the changes? No
Display the content of /proc/partitions file?
What did you notice? The new partition is not there
How to overcome that? reboot



```
maiyasser@localhost:~  
File Edit View Search Terminal Help  
[maiyasser@localhost ~]$ cat /proc/partitions  
major minor #blocks name  
  
 8         0 234431064 sda  
 8         1   51200 sda1  
 8         2 52171776 sda2  
 8         3 113664000 sda3  
 8         4      1 sda4  
 8         5   614400 sda5  
 8         6  1048576 sda6  
 8         7 66876416 sda7  
253        0 40329216 dm-0  
253        1   6856704 dm-1  
253        2  19689472 dm-2  
 8        16 488386583 sdb  
 8        17 488383488 sdb1  
[maiyasser@localhost ~]$
```

4. Make a new ext2 file system on the new logical partition you just created.
Bonus: Try creating the ext2 filesystem with 2k blocks and one inode per every 4k (two blocks) of filesystem.

```
maiyasser@localhost:~  
File Edit View Search Terminal Help  
[maiyasser@localhost ~]$ mkfs.ext4 /dev/sdb1  
mke2fs 1.45.6 (20-Mar-2020)  
Could not open /dev/sdb1: Permission denied  
[maiyasser@localhost ~]$ sudo mkfs.ext4 /dev/sdb1  
[sudo] password for maiyasser:  
mke2fs 1.45.6 (20-Mar-2020)  
/dev/sdb1 contains a ntfs file system labelled 'New Volume'  
Proceed anyway? (y,N) y  
Creating filesystem with 122095872 4k blocks and 30531584 inodes  
Filesystem UUID: 94ad65bb-a195-41a9-883c-fb8f04b1b01d  
Superblock backups stored on blocks:  
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,  
    4096000, 7962624, 11239424, 20480000, 23887872, 71663616, 78675968,  
    102400000  
  
Allocating group tables: done  
Writing inode tables: done  
Creating journal (262144 blocks): done  
Writing superblocks and filesystem accounting information: done  
  
[maiyasser@localhost ~]$
```

5. Create a directory, name it /data.

```
maiyasser@localhost:~  
File Edit View Search Terminal Help  
[maiyasser@localhost ~]$ blkid /dev/sdb1  
/dev/sdb1: LABEL="New Volume" BLOCK_SIZE="512" UUID="7014703E147008FC" TYPE="ntfs" PARTUUID="d682205a-01"  
[maiyasser@localhost ~]$ mkdir /mount_point_for_sdb1  
mkdir: cannot create directory '/mount_point_for_sdb1': Permission denied  
[maiyasser@localhost ~]$ sudo mkdir /mount_point_for_sdb1  
[sudo] password for maiyasser:  
[maiyasser@localhost ~]$ sudo mount -t ext4 /dev/sdb1 /mount_for_sdb1  
mount: /mount_for_sdb1: mount point does not exist.  
[maiyasser@localhost ~]$ sudo mount -t ext4 /dev/sdb1 /mount_point_for_sdb1  
[maiyasser@localhost ~]$ df -h /mount_point_for_sdb1/  
Filesystem      Size  Used Avail Use% Mounted on  
/dev/sdb1       458G   73M  435G   1% /mount_point_for_sdb1  
[maiyasser@localhost ~]$ mkdir /mount_point_for_sdb1/data  
mkdir: cannot create directory '/mount_point_for_sdb1/data': Permission denied  
[maiyasser@localhost ~]$ sudo mkdir /mount_point_for_sdb1/data  
[maiyasser@localhost ~]$
```

6. Add a label to the new filesystem, name it data.

```
maiyasser@localhost:~  
File Edit View Search Terminal Help  
[maiyasser@localhost ~]$ sudo e2label /dev/sdb1 /data  
[maiyasser@localhost ~]$
```

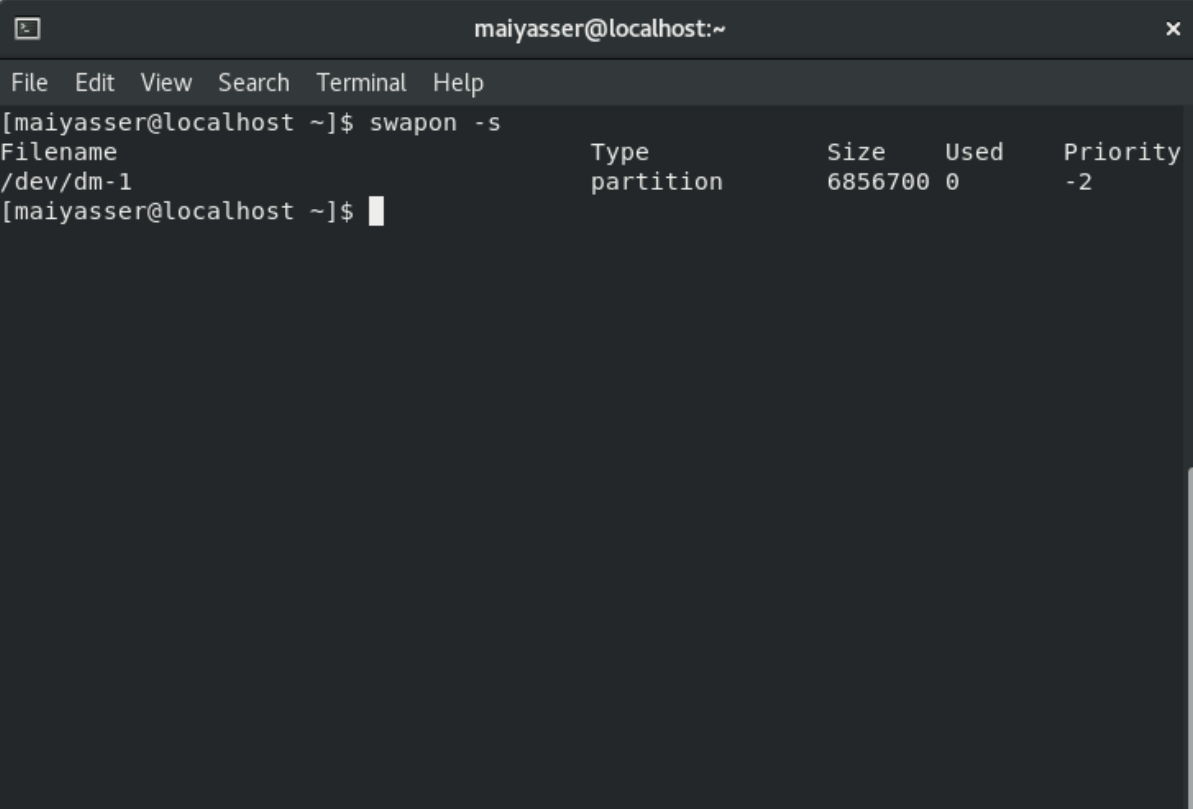
7. Add a new entry to /etc/fstab for the new filesystem using the label you just create.

```
maiyasser@localhost:~  
File Edit View Search Terminal Help  
#  
# /etc/fstab  
# Created by anaconda on Mon Nov 21 15:07:02 2022  
#  
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.  
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.  
#  
# After editing this file, run 'systemctl daemon-reload' to update systemd  
# units generated from this file.  
#  
/dev/mapper/rhel-root / xfs defaults 0 0  
UUID=cb3d4298-e702-4421-81ae-b27ee12b8865 /boot xfs default  
ts 0 0  
UUID=40D4-C733 /boot/efi vfat umask=0077,shortname=win  
nt 0 2  
/dev/mapper/rhel-home /home xfs defaults 0 0  
/dev/mapper/rhel-swap none swap defaults 0 0  
LABEL=data /data ext4 defaults 0 0  
~  
~  
~  
~  
-- INSERT -- 17,76 Bot
```

8. Mount the new filesystem.

mount /dev/sdb1 /data

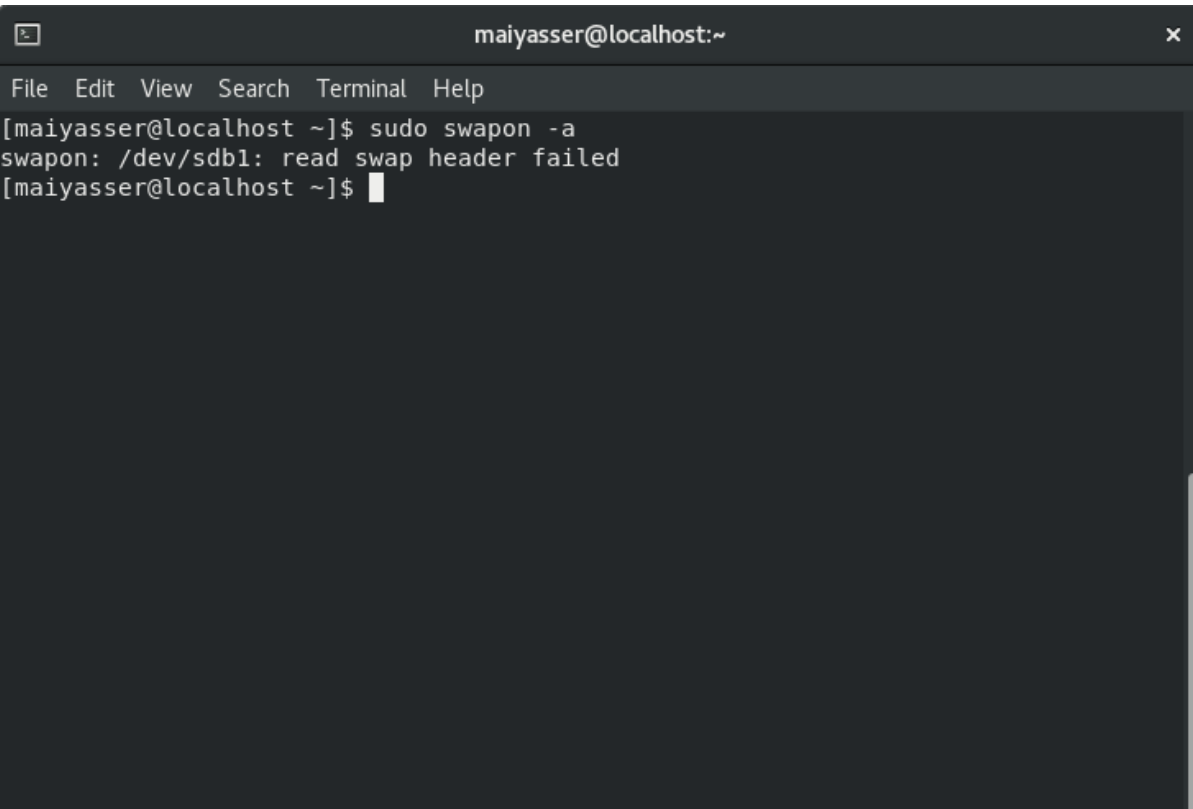
9. Display your swap size.



```
maiyasser@localhost:~  
File Edit View Search Terminal Help  
[maiyasser@localhost ~]$ swapon -s  
Filename                                Type      Size      Used      Priority  
/dev/dm-1                               partition 6856700  0         -2  
[maiyasser@localhost ~]$
```

10. Create a swap file of size 512MB.

```
LABEL=SWAP-sdb1 swap      swap  defaults    0 0  
mkswap /dev/sdb1
```



```
maiyasser@localhost:~  
File Edit View Search Terminal Help  
[maiyasser@localhost ~]$ sudo swapon -a  
swapon: /dev/sdb1: read swap header failed  
[maiyasser@localhost ~]$
```

11. Add the swap file to the virtual memory of the system.
12. Display the swap size
13. Implement disk quotas for users on the /home directory by taking the following actions
 - a. Edit /etc/fstab and add the usrquota option to the /home filesystem
 - b. Remount the filesystem with the command `mount -o remount /home`
 - c. Use the `quotacheck` command to create the quota-tracking file

`quotacheck /home`

- d. Use the `quotaon` command to enable quota tracking by the kernel `quotaon /home`