

# 一、挂载一块新磁盘

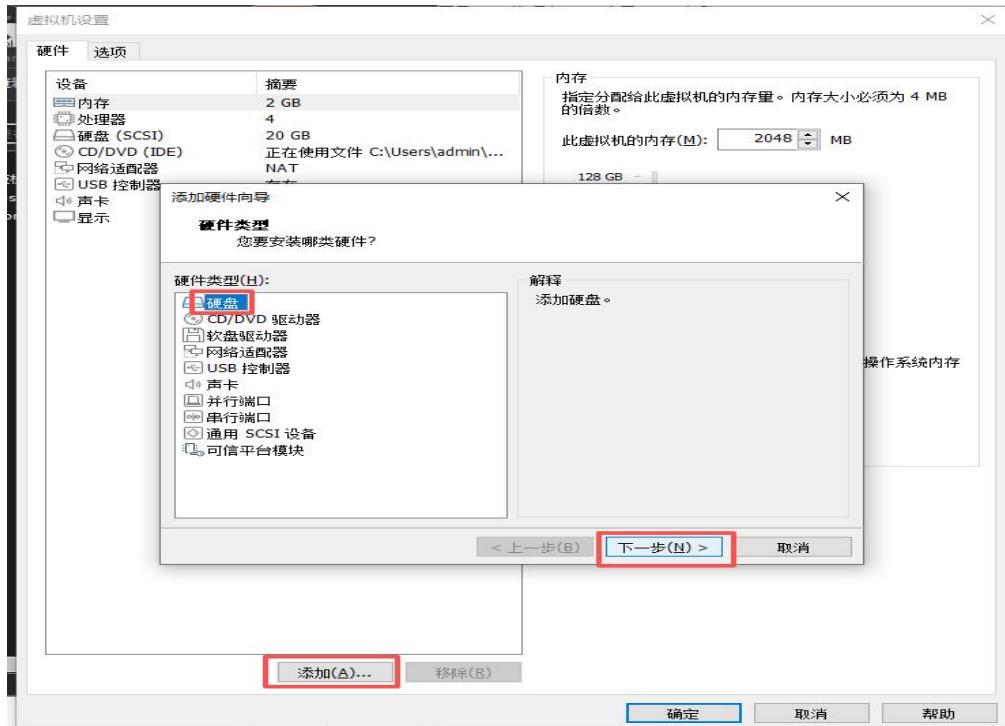
## 1、添加一块磁盘

检查现在虚拟机的磁盘状态

```
[root@localhost ~]# lsblk
NAME      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda        8:0    0   20G  0 disk
└─sda1     8:1    0    1G  0 part /boot
└─sda2     8:2    0   19G  0 part
  └─centos-root 253:0    0   17G  0 lvm   /
  └─centos-swap 253:1    0    2G  0 lvm   [SWAP]
sr0       11:0   1   4.4G 0 rom
```

```
[root@localhost ~]# lsblk
NAME      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda        8:0    0   20G  0 disk
└─sda1     8:1    0    1G  0 part /boot
└─sda2     8:2    0   19G  0 part
  └─centos-root 253:0    0   17G  0 lvm   /
  └─centos-swap 253:1    0    2G  0 lvm   [SWAP]
sr0       11:0   1   4.4G 0 rom
```

将准备好的虚拟机关机，添加一块硬盘



接下来的选项保持默认就行



添加完成后，开启虚拟机

设备	摘要
内存	2 GB
处理器	4
硬盘 (SCSI)	20 GB
新硬盘 (SCSI)	20 GB
CD/DVD (IDE)	正在使用文件 C:\Users\admin\...
网络适配器	NAT
USB 控制器	存在
声卡	自动检测
显示	自动检测

再次检查一下，磁盘添加成功

```
[root@localhost ~]# lsblk
NAME      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda          8:0    0   20G  0 disk
└─sda1      8:1    0    1G  0 part /boot
└─sda2      8:2    0   19G  0 part
  ├─centos-root 253:0    0   17G  0 lvm   /
  └─centos-swap 253:1    0    2G  0 lvm   [SWAP]
sdb          8:16   0   20G  0 disk
sr0         11:0   1   4.4G 0 rom
```

```
[root@localhost ~]# lsblk
NAME      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda        8:0    0   20G  0 disk
└─sda1     8:1    0    1G  0 part /boot
└─sda2     8:2    0   19G  0 part
  ├─centos-root 253:0  0   17G  0 lvm  /
  └─centos-swap 253:1  0    2G  0 lvm  [SWAP]
sdb        8:16   0   20G  0 disk
sr0       11:0   1  4.4G  0 rom
[root@localhost ~]#
```

## 2、手动分区

对新磁盘进行分区

```
[root@localhost ~]# fdisk /dev/sdb
n  # 新建分区
p  # 主分区
1  # 分区号
Enter # 默认起始扇区
Enter # 默认使用全部磁盘空间
w  # 保存退出
```

```
[root@localhost ~]# fdisk /dev/sdb
欢迎使用 fdisk (util-linux 2.23.2)。

更改将停留在内存中，直到您决定将更改写入磁盘。
使用写入命令前请三思。

Device does not contain a recognized partition table
使用磁盘标识符 0x29039233 创建新的 DOS 磁盘标签。

命令(输入 m 获取帮助): n
Partition type:
  p  primary (0 primary, 0 extended, 4 free)
  e  extended
Select (default p): p
分区号 (1-4, 默认 1): 1
起始 扇区 (2048-41943039, 默认为 2048):
将使用默认值 2048
Last 扇区, +扇区 or +size[K,M,G] (2048-41943039, 默认为 41943039):
将使用默认值 41943039
分区 1 已设置为 Linux 类型, 大小设为 20 GiB

命令(输入 m 获取帮助): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
正在同步磁盘。
[root@localhost ~]#
```

再次检查，添加成功

```
[root@localhost ~]# lsblk
NAME      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda        8:0    0   20G  0 disk
└─sda1     8:1    0    1G  0 part /boot
└─sda2     8:2    0   19G  0 part
  ├─centos-root 253:0  0   17G  0 lvm  /
  └─centos-swap 253:1  0    2G  0 lvm  [SWAP]
sdb        8:16   0   20G  0 disk
└─sdb1     8:17   0   20G  0 part
```

```

sr0          11:0    1  4.4G  0 rom
[root@localhost ~]# lsblk
NAME      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda        8:0    0   20G  0 disk
└─sda1     8:1    0   1G  0 part /boot
  └─sda2     8:2    0   19G  0 part
    ├─centos-root 253:0  0   17G  0 lvm /
    └─centos-swap 253:1  0   2G  0 lvm [SWAP]
sdb        8:16   0   20G  0 disk
└─sdb1     8:17   0   20G  0 part
sr0        11:0    1  4.4G  0 rom
[root@localhost ~]#

```

### 3、格式化分区（创建文件系统）

格式化为 XFS：（Centos 自带）

```
mkfs.xfs /dev/sdb1
```

若要格式化为 ext4：

```
mkfs.ext4 /dev/sdb1
```

文件系统类型	特点	适用场景
XFS	稳定通用、支持日志恢复	小文件系统、兼容性好
ext4	性能强大、支持超大文件、CentOS 默认	大容量数据、数据库、/data

我这里选得是 XFS

```

[root@localhost ~]# mkfs.xfs /dev/sdb1
meta-data=/dev/sdb1              isize=512    agcount=4, agsize=1310656 blks
                                =                      sectsz=512  attr=2, projid32bit=1
                                =                      crc=1      finobt=0, sparse=0
data     =                      bsize=4096   blocks=5242624, imaxpct=25
                                =                      sunit=0    swidth=0 blks
naming   =version 2             bsize=4096   ascii-ci=0 ftype=1
log      =internal log         bsize=4096   blocks=2560, version=2
                                =                      sectsz=512  sunit=0 blks, lazy-count=1
realtime =none                 extsz=4096   blocks=0, rtextents=0

[root@localhost ~]# mkfs.xfs /dev/sdb1
meta-data=/dev/sdb1              isize=512    agcount=4, agsize=1310656 blks
                                =                      sectsz=512  attr=2, projid32bit=1
                                =                      crc=1      finobt=0, sparse=0
data     =                      bsize=4096   blocks=5242624, imaxpct=25
                                =                      sunit=0    swidth=0 blks
naming   =version 2             bsize=4096   ascii-ci=0 ftype=1
log      =internal log         bsize=4096   blocks=2560, version=2
                                =                      sectsz=512  sunit=0 blks, lazy-count=1
realtime =none                 extsz=4096   blocks=0, rtextents=0

```

### 4、创建挂载点并挂载

创建、挂载、检查

```
[root@localhost ~]# mkdir /data  
[root@localhost ~]# mount /dev/sdb1 /data  
[root@localhost ~]# df -h | grep data  
/dev/sdb1           20G   33M   20G   1% /data
```

```
[root@localhost ~]# mount /dev/sdb1 /data  
[root@localhost ~]# df -h | grep data  
/dev/sdb1           20G   33M   20G   1% /data  
[root@localhost ~]# █
```

## 5、设置开机自动挂载（持久化）

查看分区的 UUID:

```
[root@localhost ~]# blkid /dev/sdb1  
/dev/sdb1: UUID="650f9d23-f939-4566-9faa-eb6eaec79306" TYPE="xfs"
```

编辑配置文件:

```
[root@localhost ~]# vi /etc/fstab  
# 将获得的 UUID 填写到文件最下面  
UUID=650f9d23-f939-4566-9faa-eb6eaec79306 /data xfs defaults 0 0
```

检查验证:

```
[root@localhost ~]# umount /data  
[root@localhost ~]# mount -a  
[root@localhost ~]# df -h | grep data  
/dev/sdb1           20G   33M   20G   1% /data
```

```
[root@localhost ~]# umount /data  
[root@localhost ~]# mount -a  
[root@localhost ~]# df -h | grep data  
/dev/sdb1           20G   33M   20G   1% /data  
[root@localhost ~]# █
```

重启一下，发现还是成功挂载的

```
[root@localhost ~]# reboot
Connection closing...Socket close.

Connection closed by foreign host.

Disconnected from remote host(192.168.10.100:22) at 17:09:46.

Type `help' to learn how to use Xshell prompt.
[C:\~]$ 
[C:\~]$ 
[C:\~]$ 

Connecting to 192.168.10.100:22...
Connection established.
To escape to local shell, press 'Ctrl+Alt+'.

WARNING! The remote SSH server rejected X11 forwarding request.
Last login: Wed Nov 12 16:21:12 2025 from 192.168.10.1
[root@localhost ~]# df -h
文件系统          容量  已用  可用  已用% 挂载点
devtmpfs          898M    0  898M   0% /dev
tmpfs             910M    0  910M   0% /dev/shm
tmpfs             910M  9.5M  901M   2% /run
tmpfs             910M    0  910M   0% /sys/fs/cgroup
/dev/mapper/centos-root  17G  1.5G  16G   9% /
/dev/sdbl          20G   33M   20G   1% /data
/dev/sdal          1014M  151M  864M  15% /boot
tmpfs             182M    0  182M   0% /run/user/0
[root@localhost ~]#
```

## 二、扩容磁盘

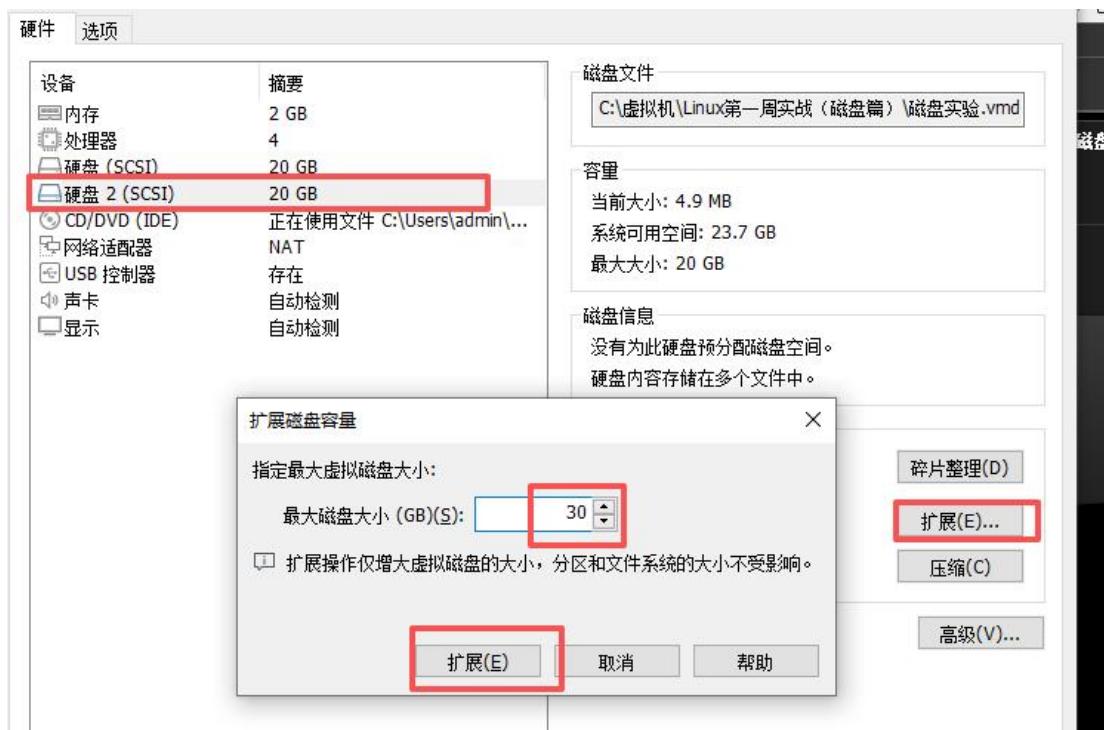
目标：在原磁盘基础上扩大磁盘容量并扩容文件系统 30G

### 1、增加磁盘容量

将虚拟机关机，

打开虚拟机设置 → 硬盘 → 修改容量，从 **20GB** 改为 **30GB**。

保存后开机。 **注：虚拟机中不可以存在快照，否则无法扩展。**



扩展成功后，重新进入虚拟机

```
Socket error Event: 32 Error: 10053.
Connection closing...Socket close.

Connection closed by foreign host.

Disconnected from remote host(192.168.10.100:22) at 17:22:26.

Type 'help' to learn how to use Xshell prompt.
[C:\~]$

Connecting to 192.168.10.100:22...
Connection established.
To escape to local shell, press 'Ctrl+Alt+'.

WARNING! The remote SSH server rejected X11 forwarding request.
Last login: Wed Nov 12 17:10:20 2025 from 192.168.10.1
[root@localhost ~]# lsblk
-bash: lsblk: 未找到命令
[root@localhost ~]# lsblk
NAME      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda        8:0    0   20G  0 disk
└─sda1     8:1    0   1G  0 part /boot
  └─centos-root 253:0  0   17G  0 lvm /
  └─centos-swap 253:1  0   2G  0 lvm [SWAP]
sdb        8:16   0   30G  0 disk
└─sdb1     8:17   0   20G  0 part /data
sr0       11:0    1  4.4G  0 rom
[root@localhost ~]#
```

说明系统已经识别了 30G，但分区还没扩展。

## 2、手动安全扩分区（无数据丢失）

```
[root@localhost ~]# fdisk -l /dev/sdb

磁盘 /dev/sdb: 32.2 GB, 32212254720 字节, 62914560 个扇区
Units = 扇区 of 1 * 512 = 512 bytes
扇区大小(逻辑/物理): 512 字节 / 512 字节
I/O 大小(最小/最佳): 512 字节 / 512 字节
磁盘标签类型: dos
```

磁盘标识符: 0x29039233

设备	Boot	Start	End	Blocks	Id	System
/dev/sdb1		2048	41943039	20970496	83	Linux

记下 /dev/sdb1 的 起始扇区, 比如:

Start = 2048

然后:

```
[root@localhost ~]# fdisk /dev/sdb
d      # 删除旧分区
n      # 新建分区
p      # 主分区
1      # 分区号
2048   # 起始扇区 (必须与原来相同)
Enter   # 结束扇区直接回车 (使用全部空间)
w      # 保存退出
```

```
[root@localhost ~]# fdisk /dev/sdb
欢迎使用 fdisk (util-linux 2.23.2)。

更改将停留在内存中, 直到您决定将更改写入磁盘。
使用写入命令前请三思。

命令(输入 m 获取帮助): d
已选择分区 1
分区 1 已删除

命令(输入 m 获取帮助): n
Partition type:
  p  primary (0 primary, 0 extended, 4 free)
  e  extended
Select (default p): p
分区号 (1-4, 默认 1): 1
起始 扇区 (2048-62914559, 默认为 2048): 2048
Last 扇区, +扇区 or +size{K,M,G} (2048-62914559, 默认为 62914559):
将使用默认值 62914559
分区 1 已设置为 Linux 类型, 大小设为 30 GiB

命令(输入 m 获取帮助): w
The partition table has been altered!

Calling ioctl() to re-read partition table.

WARNING: Re-reading the partition table failed with error 16: 设备或资源忙。
The kernel still uses the old table. The new table will be used at
the next reboot or after you run partprobe(8) or kpartx(8)
正在同步磁盘。
```

刷新分区表在查看:

```
[root@localhost ~]# partprobe /dev/sdb
[root@localhost ~]# lsblk
NAME      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda        8:0    0   20G  0 disk
└─sda1     8:1    0    1G  0 part /boot
└─sda2     8:2    0   19G  0 part
  └─centos-root 253:0    0   17G  0 lvm   /
  └─centos-swap 253:1    0    2G  0 lvm   [SWAP]
sdb        8:16   0   30G  0 disk
```

```

└─sdb1          8:17    0   30G  0 part /data
sr0           11:0    1   4.4G  0 rom

[root@localhost ~]# partprobe /dev/sdb
[root@localhost ~]# lsblk
NAME      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda        8:0    0   20G  0 disk
└─sda1     8:1    0   1G  0 part /boot
└─sda2     8:2    0   19G  0 part
└─centos-root 253:0  0   17G  0 lvm  /
centos-swap 253:1  0   3G  0 lvm  [SWAP]
sdb        8:16   0   30G  0 disk
└─sdb1     8:17   0   30G  0 part /data
sr0        11:0   1   4.4G  0 rom

```

最后，扩展文件系统：

```

[root@localhost ~]# xfs_growfs /data
meta-data=/dev/sdb1             isize=512    agcount=4, agsize=1310656 blks
                                =                      sectsz=512   attr=2, projid32bit=1
                                =                      crc=1       finobt=0 spinodes=0
data     =                      bsize=4096   blocks=5242624, imaxpct=25
                                =                      sunit=0     swidth=0 blks
naming   =version 2            bsize=4096   ascii-ci=0 ftype=1
log      =internal             bsize=4096   blocks=2560, version=2
                                =                      sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                 extsz=4096   blocks=0, rtextents=0
data blocks changed from 5242624 to 7864064

```

### 3、验证：

```

[root@localhost ~]# df -h
文件系统          容量  已用  可用  已用% 挂载点
/devtmpfs         898M    0  898M    0% /dev
tmpfs            910M    0  910M    0% /dev/shm
tmpfs            910M  9.5M  901M    2% /run
tmpfs            910M    0  910M    0% /sys/fs/cgroup
/dev/mapper/centos-root  17G  1.5G  16G    9% /
/dev/sdb1         30G  33M  30G    1% /data
/dev/sda1        1014M 151M  864M   15% /boot
tmpfs            182M    0  182M    0% /run/user/0

```

### 三、inode 排查

#### 1、查看 inode 使用情况

```

[root@localhost ~]# df -i
文件系统          Inode 已用(I) 可用(I) 已用(I)% 挂载点
/devtmpfs         229864    423  229441    1% /dev

```

```

tmpfs          232879      1  232878      1% /dev/shm
tmpfs          232879     766  232113      1% /run
tmpfs          232879      16  232863      1% /sys/fs/cgroup
/dev/mapper/centos-root 8910848   31629 8879219      1% /
/dev/sdc1       7863808     198 7863610      1% /data
/dev/sda1       524288      327 523961      1% /boot
tmpfs          232879      1  232878      1% /run/user/0

```

```

[root@localhost ~]# df -i
文件系统          Inode 已用(I) 可用(I) 已用(I)% 挂载点
/devtmpfs        229864    423 229441      1% /dev
tmpfs           232879      1  232878      1% /dev/shm
tmpfs           232879     766  232113      1% /run
tmpfs           232879      16  232863      1% /sys/fs/cgroup
/dev/mapper/centos-root 8910848   31629 8879219      1% /
/dev/sdc1       7863808     198 7863610      1% /data
/dev/sda1       524288      327 523961      1% /boot
tmpfs          232879      1  232878      1% /run/user/0
[root@localhost ~]#

```

## 2、找出哪个目录生成大量小文件

```

[root@localhost ~]# du --inodes -d 2 /data | sort -nr | head
196 /data
187 /data/mysql
111 /data/mysql/performance_schema
33  /data/mysql/#innodb_redo
11  /data/mysql/#innodb_temp
8   /data/www
7   /data/www/nginx
7   /data/mysql/mysql
2   /data/mysql/sys

```

## 3、清理 inode (删除大量小文件)

删除 7 天前的日志 / 临时文件:

```
find /data/tmp -type f -mtime +7 -delete
```

删除空目录:

```
find /data/tmp -type d -empty -delete
```

## 4、验证 inode 是否恢复

```
df -i
```

## 四、配置 swap

### 1、创建 4GB swap 文件并且给权限

```
[root@localhost ~]# dd if=/dev/zero of=/swapfile bs=1M count=4096
```

记录了 4096+0 的读入

记录了 4096+0 的写出

4294967296 字节(4.3 GB)已复制, 15.9897 秒, 269 MB/秒

```
[root@localhost ~]# chmod 600 /swapfile
```

## 2、格式化为 swap

```
[root@localhost ~]# mkswap /swapfile
正在设置交换空间版本 1, 大小 = 4194300 KiB
无标签, UUID=0369960c-df18-4b7d-9d21-c437c7b45144
```

## 3、启用 swap 并确认

```
[root@localhost ~]# swapon /swapfile
[root@localhost ~]# free -h
              total        used         free      shared  buff/cache   available
Mem:       1.8G        257M        82M        9.5M        1.4G        1.4G
Swap:      6.0G          0B        6.0G
```

## 4、写入 /etc/fstab (开机自动生效)

```
[root@localhost ~]# echo "/swapfile swap swap defaults 0 0" >> /etc/fstab
[root@localhost ~]# mount -a
```

```
[root@localhost ~]# reboot
Connection closing...Socket close.

Connection closed by foreign host.

Disconnected from remote host(192.168.10.22:22) at 15:05:22.

Type `help' to learn how to use Xshell prompt.
[C:\sim]$ 
[C:\sim]$ 

Connecting to 192.168.10.22:22...
Connection established.
To escape to local shell, press 'Ctrl+Alt+]'.

WARNING! The remote SSH server rejected X11 forwarding request.
Last login: Thu Nov 13 14:26:52 2025 from 192.168.10.1
[root@localhost ~]#
[root@localhost ~]# lsblk
NAME      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda        8:0    0  20G  0 disk
└─sda1     8:1    0   1G  0 part /boot
  └─sda2     8:2    0 19G  0 part
    ├─centos-root 253:0  0 17G  0 lvm  /
    └─centos-swap 253:1  0   2G  0 lvm  [SWAP]
sdb        8:16   0 30G  0 disk
└─sdb1     8:17   0 30G  0 part
sdc        8:32   0 15G  0 disk
└─sdc1     8:33   0 15G  0 part /data
sr0       11:0    1  4.4G 0 rom
[root@localhost ~]# free -h
              total        used         free      shared  buff/cache   available
Mem:       1.8G        263M        1.2G        9.5M        284M        1.4G
Swap:      6.0G          0B        6.0G
```

每周考题：

## 1、查看当前磁盘空间使用率的命令是什么？

磁盘使用率 = df -h

查谁占空间 = du -sh

找出占用空间大的目录 = du -sh /var/\*

## 2、如何查看系统中所有块设备（硬盘、分区）？

lsblk

查看文件系统类型 + UUID

Blkid

```
[root@localhost ~]# blkid  
/dev/sdal: UUID="83956e18-5911-4618-ab8b-0b6e1321b789" TYPE="xfs"  
/dev/sda2: UUID="y0zJeM-7Q6T-uzu7-RRZo-GTUn-RVMT-vobJh8" TYPE="LVM2_member"  
/dev/sdb1: UUID="b9899c77-ff91-405e-9896-47616c6f4654" TYPE="xfs"  
/dev/sr0: UUID="2020-11-04-11-36-43-00" LABEL="CentOS 7 x86_64" TYPE="iso9660" PTTYPE="dos"  
/dev/mapper/centos-root: UUID="62a4f626-5cda-4d64-a52c-e6fd45d32f4" TYPE="xfs"  
/dev/mapper/centos-swap: UUID="dfa99eec-a159-4aae-85fd-f1935a1af705" TYPE="swap"
```

## 3、du 与 df 的区别？

df 看的是“磁盘整体使用情况”

du 看的是“目录或文件占空间情况”（谁占了多少）

例：

Df -h 查看磁盘分区整体使用率（总空间、已用、剩余）

Du -sh /path 查看某个目录或文件实际占用多少空间

查看整个 /data 分区的使用率：

df -h /data

查看 /data 里哪个目录占空间：

du -sh /data/\*

## 4、如何永久挂载一个磁盘？

### 1、确认磁盘

lsblk

sdb

└─sdb1

## 2、格式化

```
mkfs.xfs /dev/sdb1
```

## 3、创建挂载点目录

```
mkdir /data
```

## 4、临时挂载（先看看能不能正常挂）

```
mount /dev/sdb1 /data
```

验证：

```
df -h | grep data
```

## 5、关键步骤：获取 UUID

你永久挂载一定要用 UUID（设备名会变，UUID 不会）。

```
blkid /dev/sdb1
```

输出示例：

```
/dev/sdb1: UUID="3a4d-fg12-8ab9-99e1" TYPE="xfs"
```

## 6、写入 /etc/fstab（永久挂载）

打开文件

```
vi /etc/fstab
```

在最后一行添加：

```
UUID=3a4d-fg12-8ab9-99e1 /data xfs defaults 0 0
```

## 7、测试配置有没有写对（非常关键）

```
umount /data
```

```
mount -a
```

## 8、验证

```
df -h
```







