



Linux基础入门

讲师：王晓春

Linux 基础入门

内容概述

1 Linux 基础

- 1.1 用户类型
- 1.2 终端 terminal
 - 1.2.1 终端类型
 - 1.2.2 查看当前的终端设备
- 1.3 交互式接口
 - 1.3.1 交互式接口类型
 - 1.3.2 什么是shell
 - 1.3.3 各种Shell
 - 1.3.4 bash shell
- 1.4 设置主机名
- 1.5 命令提示符 prompt
- 1.6 执行命令
 - 1.6.1 执行命令过程
 - 1.6.2 shell中可执行的两类命令
 - 1.6.2.1 内部命令相关
 - 1.6.2.2 执行外部命令
 - 1.6.3 命令别名
 - 1.6.4 命令格式
- 1.7 常见命令
 - 1.7.1 查看硬件信息
 - 1.7.1.1 查看 cpu
 - 1.7.1.2 查看内存大小
 - 1.7.1.3 查看硬盘和分区情况
 - 1.7.2 查看系统版本信息
 - 1.7.2.1 查看系统架构
 - 1.7.2.2 查看内核版本
 - 1.7.2.3 查看操作系统发行版本
 - 1.7.3 日期和时间
 - 1.7.4 关机和重启
 - 1.7.5 用户登录信息查看命令
 - 1.7.6 文本编辑
 - 1.7.7 会话管理
 - 1.7.7.1 screen

- 1.7.7.2 tmux
- 1.7.8 输出信息 echo
 - 1.7.8.1 echo 基本用法
 - 1.7.8.2 echo 高级用法
- 1.8 字符集和编码
 - 1.8.1 ASCII码
 - 1.8.2 Unicode
- 1.9 命令行扩展和被括起来的集合
 - 1.9.1 命令行扩展: `` 和 \$(
 - 1.9.2 括号扩展: { }
- 1.10 tab 键补全
 - 1.10.1 命令补全
 - 1.10.2 路径补全
 - 1.10.3 双击Tab键
- 1.11 命令行历史
- 1.12 调用命令行历史
- 1.13 bash的快捷键
- 2 获得帮助**
 - 2.1 whatis
 - 2.2 查看命令的帮助
 - 2.2.1 内部命令帮助
 - 2.2.2 外部命令及软件帮助
 - 2.3 外部命令的--help 或 -h 选项
 - 2.4 man 命令
 - 2.5 info
 - 2.6 命令自身提供的官方使用指南
 - 2.7 系统及第三方应用官方文档
 - 2.7.1 通过在线文档获取帮助
 - 2.7.2 Linux官方在线文档和知识库
 - 2.7.3 红帽全球技术支持服务
 - 2.8 相关网站和搜索

Linux 基础入门

内容概述

- 用户
- 终端
- Shell介绍
- 执行命令
- 简单命令
- Tab键补全
- 命令行历史
- bash快捷键

- 帮助用法

1 Linux 基础

1.1 用户类型

- root 用户
 - 一个特殊的管理帐户
 - 也被称为超级用户
 - root已接近完整的系统控制
 - 对系统损害几乎有无限的能力
 - 除非必要,不要登录为 root
- 普通（非特权）用户
 - 权限有限
 - 造成损害的能力比较有限

1.2 终端 terminal



设备终端：键盘、鼠标、显示器

1.2.1 终端类型

- 控制台终端： `/dev/console`
- 串行终端： `/dev/ttyS#`
- 虚拟终端： `tty`: teletypewriters, `/dev/tty#`, `tty` 可有n个, `Ctrl+Alt+F#`
- 图形终端： `startx`, `xwindows`
 - CentOS 6: `Ctrl + Alt + F7`
 - CentOS 7: 在哪个终端启动, 即位于哪个虚拟终端
- 伪终端： `pty`: pseudo-tty, `/dev/pts/#` 如: SSH远程连接

1.2.2 查看当前的终端设备

tty 命令可以查看当前所在终端

```
tty
```

范例:

```
[root@centos8 ~]#tty  
/dev/pts/0
```

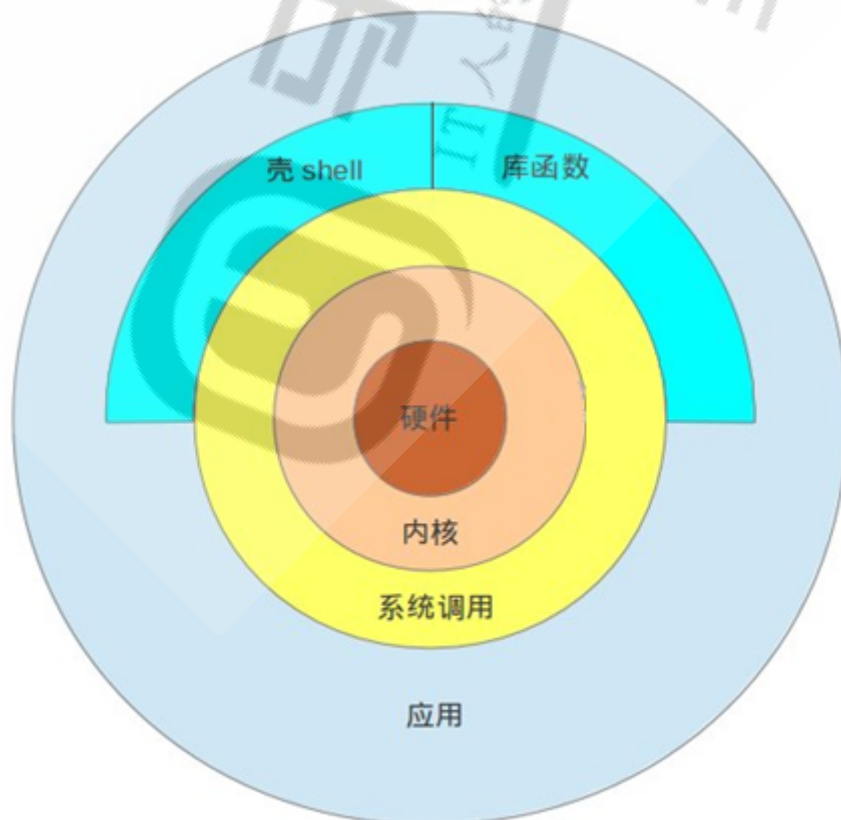
1.3 交互式接口

交互式接口：启动终端后，在终端设备附加一个交互式应用程序

1.3.1 交互式接口类型

- GUI: Graphic User Interface
X protocol, window manager, desktop
Desktop:
GNOME (C, 图形库gtk),
KDE (C++, 图形库qt)
XFCE (轻量级桌面)
- CLI: Command Line Interface
shell程序

1.3.2 什么是shell



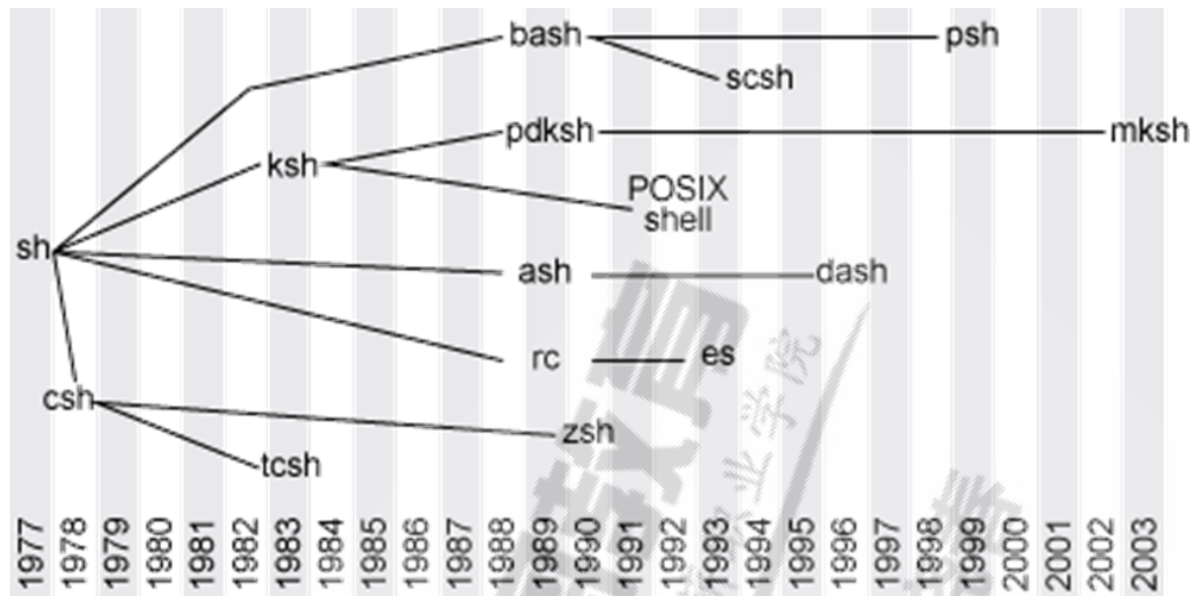
Shell 是Linux系统的用户界面，提供了用户与内核进行交互操作的一种接口。它接收用户输入的命令并把它送入内核去执行

shell也被称为Linux的命令解释器（command interpreter），Shell 本身是一个程序。将用户输入的命令行拆解为“命令名”与“参数”。接着，根据命令名找到对应要执行的程序，对被执行的程序进行初始化，然后将刚才解析出来的参数传给该程序并执行

shell是一种高级程序设计语言，提供了变量，函数，条件判断，循环等开发语言的功能

由于Shell本身是个程序，所以它可以被任何用户自己开发的各种Shell所代替

1.3.3 各种Shell



- sh: Steve Bourne
- bash: Bourne-Again Shell, GPL, CentOS 和 Ubuntu 默认使用
- csh: c shell, C 语言风格
- tcsh
- ksh: Korn Shell, AIX 默认 shell
- zsh: MacOS默认shell

1.3.4 bash shell

GNU Bourne-Again Shell(bash)是GNU计划中重要的工具软件之一，目前也是 Linux标准的shell，与sh兼容

显示当前使用的shell

```
echo ${SHELL}
```

显示当前系统使用的所有shell

```
cat /etc/shells
```

1.4 设置主机名

```
#临时生效
hostname NAME
```

```
#持久生效,支持CentOS7和Ubuntu18.04以上版本
hostnamectl set-hostname NAME
```

范例

```
[root@centos8 ~]#hostname      bj-yz-k8s-node1-100-10.magedu.local
```

注意:

- 主机名不支持使用下划线，但支持横线，可使用字母，横线或数字组合
- 有些软件对主机名有特殊要求

范例:

```
[root@centos8 ~]#hostnamectl set-hostname centos8.1
[root@centos8 ~]#systemctl restart postfix
Job for postfix.service failed because the control process exited with error
code.
See "systemctl status postfix.service" and "journalctl -xe" for details.
```

1.5 命令提示符 prompt

登录Linux后，默认的系统命令提示符毫无没有个性，无法明显辨别生产和测试环境，而导致误操作。可以通过修改PS1变量实现个性的提示符格式，避免这种低级错误

范例：默认的提示符

```
#CentOS默认提示符
[root@localhost ~]#

#Ubuntu默认提示符
root@ubuntu1804:~#
```

管理员

\$ 普通用户

显示提示符格式

```
[root@centos8 ~]#echo $PS1
\[ \e[1;35m\[ \u@\h \w\]\$ \e[0m\]
```

修改提示符格式范例

```
PS1="\[ \e[1;5;41;33m\[ \u@\h \w\]\$ \e[0m\]"
PS1="\[ \e[1;32m\[ \t \e[1;33m\[ \u\[\e[35m\]@\h\[\e[1;31m\ \w\[\e[1;32m\]\]\[ \e[0m\]\$"
```

提示符格式说明:

- \e 控制符\033
- \u 当前用户
- \h 主机名简称
- \H 主机名
- \w 当前工作目录
- \W 当前工作目录基名
- \t 24小时时间格式

- \T 12小时时间格式
- ! 命令历史数
- # 开机后命令历史数

范例：在CentOS系统实现持久保存提示符格式

```
[root@centos8 ~]# echo 'PS1="\[\e[1;32m\]\[t \[\e[1;33m\]\u\[\e[35m\]@\h\[\e[1;31m\] \w\[\e[1;32m\]]\[\e[0m\]\$"' > /etc/profile.d/env.sh
[root@centos8 ~]# cat /etc/profile.d/env.sh
PS1="\[\e[1;32m\]\[t \[\e[1;33m\]\u\[\e[35m\]@\h\[\e[1;31m\] \w\[\e[1;32m\]]\[\e[0m\]\$"
```

[root@centos8 ~]# exit

logout

Connection closed by foreign host.

Disconnected from remote host(centos8) at 15:28:38.

Type 'help' to learn how to use Xshell prompt.

[c:\~]\$

Reconnecting in 1 seconds. Press any key to exit local shell.

.

Connecting to 10.0.0.100:22...

Connection established.

To escape to local shell, press 'Ctrl+Alt+J'.

Activate the web console with: `systemctl enable --now cockpit.socket`

Last login: Wed Dec 11 15:28:34 2019 from 10.0.0.1

[15:28:40 root@centos8 ~]#

```
[root@centos8 ~]#PS1="\[\e[1;32m\]\[t \[\e[1;33m\]\u\[\e[35m\]@\h\[\e[1;31m\] \w\[\e[1;32m\]]\[\e[0m\]\$"
```

[23:42:04 root@centos8 ~]#

范例：实现Ubuntu系统持久保存提示符格式

```
[root@ubuntu2004 ~]#echo "PS1='\[\e[1;35m\]\[u@\h \w\]\$'\[\e[0m\]'" >> .bashrc
[root@ubuntu2004 ~]#tail -1 .bashrc
PS1='\[\e[1;35m\]\[u@\h \w\]\$'\[\e[0m\]'
```

```
Last login: Tue May 26 15:17:56 2020 from 10.0.0.1
[root@ubuntu2004 ~]#
```

1.6 执行命令

1.6.1 执行命令过程

输入命令后回车，提请shell程序找到键入命令所对应的可执行程序或代码，并由其分析后提交给内核分配资源将其运行起来

1.6.2 shell中可执行的两类命令

- 内部命令：由shell自带的，而且通过某命令形式提供，用户登录后自动加载并常驻内存中

- 外部命令：在文件系统路径下有对应的可执行程序文件,当执行命令时才从磁盘加载至内存中,执行完毕后从内存中删除

区别指定的命令是内部或外部命令

```
type COMMAND
```

范例: 查看是否存在对应内部和外部命令

```
[root@centos8 ~]#type -a echo
echo is a shell builtin
echo is /usr/bin/echo
```

1.6.2.1 内部命令相关

help 内部命令列表

enable 管理内部命令

- enable cmd 启用内部命令
- enable -n cmd 禁用内部命令
- enable -n 查看所有禁用的内部命令

1.6.2.2 执行外部命令

查看外部命令路径:

```
which -a |--skip-alias
whereis
```

Hash缓存表

系统初始hash表为空, 当外部命令执行时, 默认会从PATH路径下寻找该命令, 找到后会将这条命令的路径记录到hash表中, 当再次使用该命令时, shell解释器首先会查看hash表, 存在将执行之, 如果不存在, 将会去PATH路径下寻找, 利用hash缓存表可大大提高命令的调用速率

hash 命令常见用法

- hash 显示hash缓存
- hash -l 显示hash缓存, 可作为输入使用
- hash -p path name 将命令全路径path起别名为name
- hash -t name 打印缓存中name的路径
- hash -d name 清除name缓存
- hash -r 清除缓存

1.6.3 命令别名

对于经常执行的较长的命令, 可以将其定义成较短的别名, 以方便执行

显示当前shell进程所有可用的命令别名

```
alias
```


定义别名NAME，其相当于执行命令VALUE

```
alias NAME='VALUE'
```

范例: 扫描新加的磁盘

```
[root@centos8 ~]#alias scandisk='echo - - - >
/sys/class/scsi_host/host0/scan;echo - - - >
/sys/class/scsi_host/host1/scan;echo - - - > /sys/class/scsi_host/host2/scan'
```

范例: 持久保存别名

撤消别名: unalias

```
unalias [-a] name [name ...]
unalias -a #取消所有别名
```

注意: 在命令行中定义的别名, 仅对当前shell进程有效

如果想永久有效, 要定义在配置文件中

- 仅对当前用户: ~/.bashrc
- 对所有用户有效: /etc/bashrc

编辑配置给出的新配置不会立即生效, bash进程重新读取配置文件

```
source /path/to/config_file
. /path/to/config_file
```

如果别名同原命令同名, 如果要执行原命令, 可使用

```
\ALIASNAME
"ALIASNAME"
'ALIASNAME'
command ALIASNAME
/path/command #只适用于外部命令
```

1.6.4 命令格式

```
COMMAND [OPTIONS...] [ARGUMENTS...]
COMMAND [COMMAND] [COMMAND] ....
```

选项: 用于启用或关闭命令的某个或某些功能

- 短选项: UNIX 风格选项, -c 例如: -l, -h
- 长选项: GNU风格选项, --word 例如: --all, --human
- BSD风格选项: 一个字母, 例如: a, 使用相对较少

参数: 命令的作用对象, 比如: 文件名, 用户名等

范例:

```
[root@centos8 ~]#id -u wang
1000

[root@centos8 ~]#ls -a
[root@centos8 ~]#ls --all

[root@centos8 ~]#free -h
[root@centos8 ~]#free --human

[root@centos8 ~]#ps a
```

注意:

- 多个选项以及多参数和命令之间使用空白字符分隔
- 取消和结束命令执行: Ctrl+c, Ctrl+d
- 多个命令可以用 ";" 符号分开
- 一个命令可以用 \ 分成多行

1.7 常见命令

1.7.1 查看硬件信息

1.7.1.1 查看 cpu

lscpu命令可以查看cpu信息

cat /proc/cpuinfo也可查看看到

范例:

```
[root@centos8 ~]# lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                 8
On-line CPU(s) list:   0-7
Thread(s) per core:    1 #每个core 有几个线程
Core(s) per socket:    4 #每个槽位有4个core
Socket(s):              2 #服务器面板上有2个cpu 槽位
NUMA node(s):          2 #nodes的数量
Vendor ID:              GenuineIntel
CPU family:             6
Model:                  44
Stepping:               2
CPU MHz:                2128.025
BogoMIPS:               4256.03
Virtualization:         VT-x
L1d cache:              32K
L1i cache:              32K
L2 cache:               256K
L3 cache:               8192K
NUMA node0 CPU(s):     0,2,4,6 #对应的core
NUMA node1 CPU(s):     1,3,5,7
```

```
[root@centos8 ~]# cat /proc/cpuinfo
```

1.7.1.2 查看内存大小

```
[root@centos8 ~]#free
              total        used        free      shared  buff/cache   available
Mem:          816188      156888      412660         5792       246640       527024
Swap:        2097148           0      2097148

[root@centos8 ~]#cat /proc/meminfo
MemTotal:        816188 kB
MemFree:         412780 kB
MemAvailable:    527144 kB
Buffers:         5304 kB
Cached:          208252 kB
SwapCached:      0 kB
Active:          138976 kB
Inactive:        127508 kB
Active(anon):    53348 kB
Inactive(anon):  5368 kB
Active(file):    85628 kB
Inactive(file):  122140 kB
Unevictable:     0 kB
Mlocked:         0 kB
SwapTotal:       2097148 kB
SwapFree:        2097148 kB
Dirty:           0 kB
Writeback:       0 kB
AnonPages:       52964 kB
Mapped:          78456 kB
Shmem:           5792 kB
KReclaimable:    33084 kB
Slab:            69144 kB
SReclaimable:    33084 kB
SUnreclaim:     36060 kB
KernelStack:    4428 kB
PageTables:      5888 kB
NFS_Unstable:    0 kB
Bounce:          0 kB
WritebackTmp:    0 kB
CommitLimit:    2505240 kB
Committed_AS:    250040 kB
VmallocTotal:    34359738367 kB
VmallocUsed:     0 kB
VmallocChunk:    0 kB
HardwareCorrupted: 0 kB
AnonHugePages:   6144 kB
ShmemHugePages:  0 kB
ShmemPmdMapped:  0 kB
HugePages_Total: 0
HugePages_Free:  0
HugePages_Rsvd:  0
HugePages_Surp:  0
Hugepagesize:    2048 kB
Hugetlb:         0 kB
DirectMap4k:    108416 kB
DirectMap2M:    940032 kB
DirectMap1G:    0 kB
[root@centos8 ~]#
```

1.7.1.3 查看硬盘和分区情况

```
[root@centos8 ~]#lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda          8:0    0   200G  0 disk
├─sda1       8:1    0    1G  0 part /boot
├─sda2       8:2    0   100G  0 part /
├─sda3       8:3    0    50G  0 part /data
├─sda4       8:4    0     1K  0 part
└─sda5       8:5    0     2G  0 part [SWAP]
sr0         11:0    1     7G  0 rom

[root@centos8 ~]#cat /proc/partitions
major minor  #blocks  name

   8         0  209715200 sda
   8         1   1048576 sda1
   8         2  104857600 sda2
   8         3   52428800 sda3
   8         4             1 sda4
   8         5    2097152 sda5
  11         0   7377920 sr0
```

1.7.2 查看系统版本信息

1.7.2.1 查看系统架构

```
root@ubuntu2004:~# arch
x86_64
```

```
[root@centos8 ~]# arch
x86_64
```

```
[root@rhel5 ~]# arch
i686
```

1.7.2.2 查看内核版本

```
[root@centos8 ~]#uname -r
4.18.0-147.el8.x86_64
```

```
[root@centos7 ~]#uname -r
3.10.0-1062.el7.x86_64
```

```
[root@centos6 ~]# uname -r
2.6.32-754.el6.x86_64
```

```
[root@ubuntu1804 ~]#uname -r
4.15.0-29-generic
```

1.7.2.3 查看操作系统发行版本

#CentOS8 查看发行版本

```
[root@centos8 ~]#cat /etc/redhat-release
CentOS Linux release 8.1.1911 (Core)
```

```
[root@centos8 ~]#cat /etc/os-release
```

```
NAME="CentOS Linux"
VERSION="8 (Core)"
ID="centos"
ID_LIKE="rhel fedora"
VERSION_ID="8"
PLATFORM_ID="platform:el8"
PRETTY_NAME="CentOS Linux 8 (Core)"
ANSI_COLOR="0;31"
CPE_NAME="cpe:/o:centos:centos:8"
HOME_URL="https://www.centos.org/"
BUG_REPORT_URL="https://bugs.centos.org/"
```

```
CENTOS_MANTISBT_PROJECT="CentOS-8"
CENTOS_MANTISBT_PROJECT_VERSION="8"
REDHAT_SUPPORT_PRODUCT="centos"
REDHAT_SUPPORT_PRODUCT_VERSION="8"
```

```
[root@centos8 ~]#lsb_release -a
LSB Version:      :core-4.1-amd64:core-4.1-noarch
Distributor ID:  CentOS
Description:     CentOS Linux release 8.1.1911 (Core)
Release:         8.1.1911
Codename:        Core
[root@centos8 ~]#
```

#ubuntu查看发行版本

```
[root@ubuntu1804 ~]#cat /etc/os-release
NAME="Ubuntu"
VERSION="18.04.1 LTS (Bionic Beaver)"
ID=ubuntu
ID_LIKE=debian
PRETTY_NAME="Ubuntu 18.04.1 LTS"
VERSION_ID="18.04"
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
VERSION_CODENAME=bionic
UBUNTU_CODENAME=bionic
```

```
[root@ubuntu1804 ~]#cat /etc/issue
Ubuntu 18.04.1 LTS \n \l
```

```
[root@ubuntu1804 ~]#lsb_release -a
No LSB modules are available.
Distributor ID:  Ubuntu
Description:     Ubuntu 18.04.1 LTS
Release:         18.04
Codename:        bionic
[root@ubuntu1804 ~]#
```

范例: 查看 OS 版本

```
[root@centos8 ~]#lsb_release -is
```

```
CentOS
[root@centos8 ~]#lsb_release -cs
Core
[root@centos8 ~]#lsb_release -rs
8.2.2004

[root@centos7 ~]#lsb_release -is
CentOS
[root@centos7 ~]#lsb_release -cs
Core
[root@centos7 ~]#lsb_release -rs
7.9.2009

[root@centos6 ~]#lsb_release -is
CentOS
[root@centos6 ~]#lsb_release -cs
Final
[root@centos6 ~]#lsb_release -rs
6.10

root@ubuntu2004:~# lsb_release -is
Ubuntu
root@ubuntu2004:~# lsb_release -cs
focal
root@ubuntu2004:~# lsb_release -rs
20.04

[root@ubuntu1804 ~]#lsb_release -is
Ubuntu
[root@ubuntu1804 ~]#lsb_release -cs
bionic
[root@ubuntu1804 ~]#lsb_release -rs
18.04
```

1.7.3 日期和时间

Linux的两种时钟

- 系统时钟：由Linux内核通过CPU的工作频率进行的
- 硬件时钟：主板

相关命令

- date 显示和设置系统时间

范例：


```
[root@centos8 ~]#date +%s
1584689051
[root@centos8 ~]#date -d @"date +%s`
Fri Mar 20 15:24:18 CST 2020

[root@centos8 ~]#date -d @1584689051
Fri Mar 20 15:24:11 CST 2020

[root@centos8 ~]#date -d @1584689051 +%F_%T
2020-03-20_15:24:11

[root@ubuntu ~]#date -d "2020-02-02" +%s
1580572800
```

- clock, hwclock: 显示硬件时钟

```
-s, --hctosys #以硬件时钟为准, 校正系统时钟
-w, --systemd #以系统时钟为准, 校正硬件时钟
```

范例:

```
[root@centos8 ~]#ll /usr/sbin/clock
lrwxrwxrwx. 1 root root 7 Apr 24 2020 /usr/sbin/clock -> hwclock
```

时区:

```
/etc/localtime
```

范例:

```
[root@centos8 ~]#timedatectl list-timezones

[root@centos8 ~]#timedatectl set-timezone Asia/Shanghai

[root@centos8 ~]#timedatectl status
      Local time: Mon 2020-07-27 17:20:56 CST
      Universal time: Mon 2020-07-27 09:20:56 UTC
      RTC time: Mon 2020-07-27 09:20:56
      Time zone: Asia/Shanghai (CST, +0800)
System clock synchronized: yes
      NTP service: active
      RTC in local TZ: no

[root@centos8 ~]#ll /etc/localtime
lrwxrwxrwx. 1 root root 35 Dec 11 11:19 /etc/localtime ->
../usr/share/zoneinfo/Asia/Shanghai

root@ubuntu1804:~# cat /etc/timezone
Asia/Shanghai
```

显示日历:

```
cal -y
```

范例:

```
[16:47:36 root@centos8 ~]#cal 9 1752
September 1752
Su Mo Tu We Th Fr Sa
                1  2 14 15 16
17 18 19 20 21 22 23
24 25 26 27 28 29 30
```

1.7.4 关机和重启

关机:

- halt
- poweroff

重启:

reboot
-f: 强制, 不调用shutdown
-p: 切断电源

关机或重启: shutdown

```
shutdown [OPTION]... [TIME] [MESSAGE]
```

-r: reboot
-h: halt
-c: cancel
TIME: 无指定, 默认相当于+1 (CentOS7)
now: 立刻, 相当于+0
+#: 相对时间表示法, 几分钟之后; 例如 +3
hh:mm: 绝对时间表示, 指明具体时间

1.7.5 用户登录信息查看命令

- whoami: 显示当前登录有效用户
- who: 系统当前所有的登录会话
- w: 系统当前所有的登录会话及所做的操作

1.7.6 文本编辑

- nano 工具可以实现文本的编辑, 上手容易, 适合初学者
- gedit 工具是图形工具

范例: 创建登录提示文件 /etc/motd

参考网站: <https://www.bootschool.net/ascii-art>


```
New release '20.04.1 LTS' available.  
Run 'do-release-upgrade' to upgrade to it.
```

[illegible]

命令行的典型使用方式是，打开一个终端窗口（terminal window，以下简称“窗口”），在里面输入命令。用户与计算机的这种临时的交互，称为一次“会话”（session）

一个典型的例子就是，SSH 登录远程计算机，打开一个远程窗口执行命令。这时，网络突然断线，再次登录的时候，是找不回上一次执行的命令的。因为上一次 SSH 会话已经终止了，里面的进程也随之消失了。为了解决这个问题，会话与窗口可以“解绑”：窗口关闭时，会话并不终止，而是继续运行，等到以后需要的时候，再让会话“绑定”其他窗口

- (1) 它允许在单个窗口中，同时访问多个会话。这对于同时运行多个命令程序很有用。
- (2) 它可以让新窗口"接入"已经存在的会话。
- (3) 它允许每个会话有多个连接窗口，因此可以多人实时共享会话。
- (4) 它还支持窗口任意的垂直和水平拆分。

类似的终端复用器还有Screen, Tmux

1.7.7.1 screen

利用screen 可以实现会话管理,如: 新建会话,共享会话等

注意: CentOS7 来自于base源, CentOS8 来自于epel源

范例: 安装 screen

```
#CentOS7 安装screen
[root@centos7 ~]#yum -y install screen

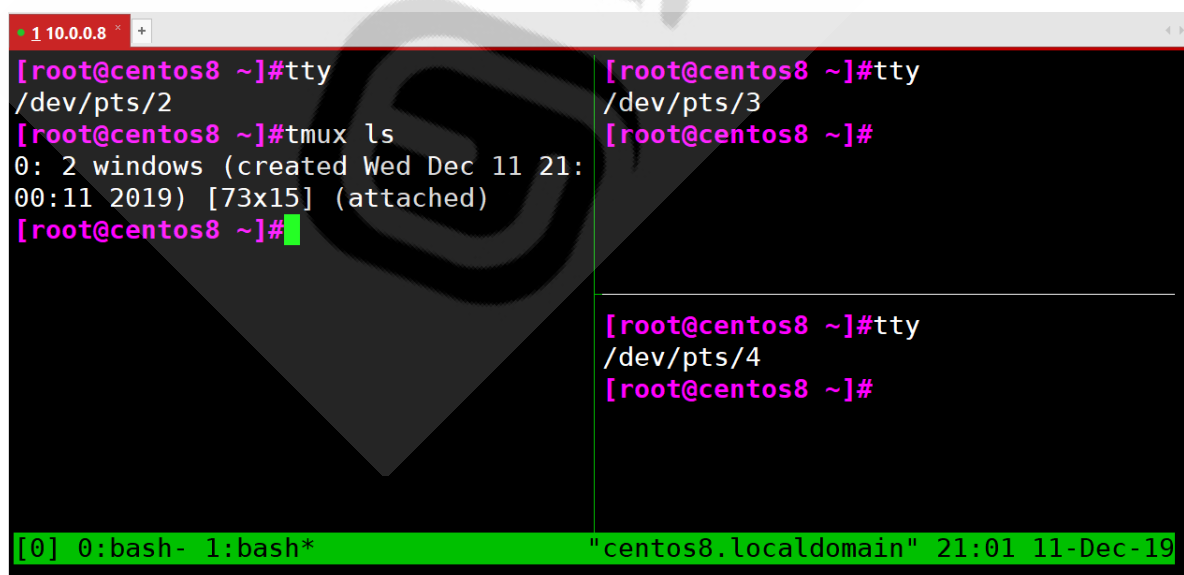
#CentOS8 安装screen
[root@centos8 ~]#dnf -y install epel-release
[root@centos8 ~]#dnf -y install screen
```

screen命令常见用法:

- 创建新screen会话

screen -S [SESSION]
- 加入screen会话
screen -x [SESSION]
- 退出并关闭screen会话
exit
- 剥离当前screen会话
Ctrl+a,d
- 显示所有已经打开的screen会话
screen -ls
- 恢复某screen会话
screen -r [SESSION]

1.7.7.2 tmux



```
1 10.0.0.8
[root@centos8 ~]#tty
/dev/pts/2
[root@centos8 ~]#tmux ls
0: 2 windows (created Wed Dec 11 21:00:11 2019) [73x15] (attached)
[root@centos8 ~]#

[root@centos8 ~]#tmux
/dev/pts/3
[root@centos8 ~]#

[root@centos8 ~]#tmux
/dev/pts/4
[root@centos8 ~]#

[0] 0: bash- 1: bash* "centos8.localdomain" 21:01 11-Dec-19
```

Tmux 是一个终端复用器 (terminal multiplexer) , 类似 screen, 但是更易用, 也更强大

Tmux 就是会话与窗口的"解绑"工具, 将它们彻底分离, 功能如下

- 它允许在单个窗口中, 同时访问多个会话。这对于同时运行多个命令程序很有用。
- 它可以让新窗口"接入"已经存在的会话。
- 它允许每个会话有多个连接窗口, 因此可以多人实时共享会话。
- 它还支持窗口任意的垂直和水平拆分

安装

```
yum install tmux
```

启动与退出

```
[root@centos8 ~]#tmux  
[root@centos8 ~]#exit  
logout
```

mux 窗口有大量的快捷键。所有快捷键都要通过前缀键唤起。默认的前缀键是 `Ctrl+b`，即先按下 `Ctrl+b`，快捷键才会生效。帮助命令的快捷键是 `Ctrl+b ?` 然后，按下 `q` 键，就可以退出帮助

新建会话

第一个启动的 Tmux 窗口，编号是0，第二个窗口的编号是1，以此类推。这些窗口对应的会话，就是 0 号会话、1 号会话。使用编号区分会话，不太直观，更好的方法是为会话起名。下面命令新建一个指定名称的会话。

```
tmux new -s <session-name>
```

tmux ls或Ctrl+b,s 可以查看当前所有的 Tmux 会话

```
tmux ls  
tmux list-session
```

分离会话

在 Tmux 窗口中，按下Ctrl+b d或者输入tmux detach命令，就会将当前会话与窗口分离。

```
tmux detach
```

接入会话

tmux attach 命令用于重新接入某个已存在的会话。

```
tmux attach -t <session-name>
```

范例：

```
tmux attach -t 0
```

杀死会话

tmux kill-session命令用于杀死某个会话。

```
tmux kill-session -t <session-name>
```

切换会话

tmux switch命令用于切换会话

```
tmux switch -t <session-name>
```


可以将窗口分成多个窗格 (pane) , 每个窗格运行不同的命令

上下分窗格

```
tmux split-window  
ctrl+b,"
```

左右分窗格

```
tmux split-window -h  
ctrl+b,%
```

窗格快捷键

Ctrl+b %: 划分左右两个窗格
Ctrl+b ": 划分上下两个窗格
Ctrl+b <arrow key>: 光标切换到其他窗格。<arrow key>是指向要切换到的窗格的方向键, 比如切换到下方窗格, 就按方向键↓
Ctrl+b ;: 光标切换到上一个窗格
Ctrl+b o: 光标切换到下一个窗格。
Ctrl+b {: 当前窗格左移
Ctrl+b }: 当前窗格右移
Ctrl+b Ctrl+o: 当前窗格上移
Ctrl+b Alt+o: 当前窗格下移
Ctrl+b x: 关闭当前窗格
Ctrl+b !: 将当前窗格拆分为一个独立窗口
Ctrl+b z: 当前窗格全屏显示, 再使用一次会变回原来大小
Ctrl+b Ctrl+<arrow key>: 按箭头方向调整窗格大小
Ctrl+b q: 显示窗格编号

窗口管理

除了将一个窗口划分成多个窗格, Tmux 也允许新建多个窗口

新建窗口

tmux new-window命令用来创建新窗口

```
tmux new-window
```

新建一个指定名称的窗口

```
tmux new-window -n <window-name>
```

切换窗口

tmux select-window命令用来切换窗口

切换到指定编号的窗口

```
tmux select-window -t <window-number>
```

切换到指定名称的窗口

```
tmux select-window -t <window-name>
```

窗口快捷键

Ctrl+b c: 创建一个新窗口，状态栏会显示多个窗口的信息。
Ctrl+b p: 切换到上一个窗口（按照状态栏上的顺序）。
Ctrl+b n: 切换到下一个窗口。
Ctrl+b <number>: 切换到指定编号的窗口，其中的<number>是状态栏上的窗口编号
Ctrl+b w: 从列表中选择窗口
Ctrl+b ,: 窗口重命名

列出所有快捷键，及其对应的 Tmux 命令

```
tmux list-keys
```

列出所有 Tmux 命令及其参数

```
tmux list-commands
```

1.7.8 输出信息 echo

1.7.8.1 echo 基本用法

echo 命令可以将后面跟的字符进行输出

功能：显示字符，echo会将输入的字符串送往标准输出。输出的字符串间以空白字符隔开, 并在最后加上换行号

语法：

```
echo [-neE] [字符串]
```

选项：

- -E （默认）不支持 \ 解释功能
- -n 不自动换行
- -e 启用 \ 字符的解释功能

显示变量

```
echo "$VAR_NAME"    #用变量值替换，弱引用
echo '$VAR_NAME'    #变量不会替换，强引用
```

启用命令选项-e，若字符串中出现以下字符，则特别加以处理，而不会将它当成一般文字输出

- \a 发出警告声
- \b 退格键
- \c 最后不加上换行符号
- \e escape, 相当于\033
- \n 换行且光标移至行首
- \r 回车，即光标移至行首，但不换行
- \t 插入tab
- \\ 插入\字符
- \0nnn 插入nnn（八进制）所代表的ASCII字符
- \xHH插入HH（十六进制）所代表的ASCII数字（man 7 ascii）

范例：

```
[root@centos8 ~]#echo -e 'a\x0Ab'
a
b
[root@centos8 ~]#echo -e '\033[43;31;1;5mmagedu\e[0m'
magedu

[root@centos8 ~]#echo -e '\x57\x41\x4E\x47'
WANG

[root@centos8 ~]#echo \$PATH
$PATH

[root@centos8 ~]#echo \
[root@centos8 ~]#echo \\
\
[root@centos8 ~]#echo \\\
>
[root@centos8 ~]#echo \\\\
\\

[root@centos8 ~]#echo "$PATH"
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/root/bin
[root@centos8 ~]#echo '$PATH'
$PATH
```

1.7.8.2 echo 高级用法

在终端中，ANSI定义了用于屏幕显示的Escape屏幕控制码具有颜色的字符，其格式如下：

```
"\033[字符背景颜色;字体颜色m字符串\033[0m"
```

\033[30m -- \033[37m 设置前景色

\033[40m -- \033[47m 设置背景色

#字符背景颜色范围：40--47

40: 黑
41: 红
42: 绿
43: 黄
44: 蓝
45: 紫
46: 深绿
47: 白色

#字体颜色：30--37

30: 黑
31: 红
32: 绿
33: 黄
34: 蓝
35: 紫
36: 深绿

加颜色只是以下控制码中的一种，下面是常见的一些ANSI控制码：

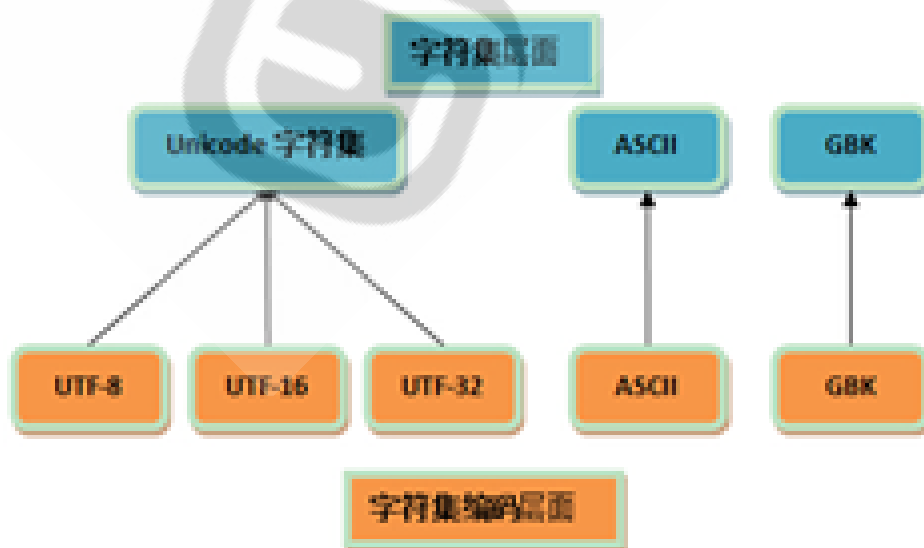
```
\033[0m  关闭所有属性
\033[1m  设置高亮度
\033[4m  下划线
\033[5m  闪烁
\033[7m  反显
\033[8m  消隐
\033[nA   光标上移n行
\033[nB   光标下移n行
\033[nC   光标右移n列
\033[nD   光标左移n列
\033[x;yH 设置光标位置x行y列
\033[2J   清屏
\033[K    清除从光标到行尾的内容
\033[s    保存光标位置
\033[u    恢复光标位置
\033[?25l  隐藏光标
\033[?25h  显示光标
\033[2J\033[0;0H  清屏且将光标置顶
```

1.8 字符集和编码

许多场合下，字符集与编码这两个概念常被混为一谈，但两者是有差别的。字符集与字符集编码是两个不同层面的概念

charset是character set的简写，即字符集，即二进制和字符的对应关系，不关注最终的存储形式

encoding是charset encoding的简写，即字符集编码，简称编码，实现如何将字符转化为实际的二进制进行存储或相反，编码决定了空间的使用的大小



1.8.1 ASCII码

计算机内部，所有信息最终都是一个二进制值。上个世纪60年代，美国制定了一套字符编码，对英语字符与二进制位之间的关系，做了统一规定，即ASCII（American Standard Code for Information Interchange）码

ASCII 码一共规定了128个字符的编码，占用了一个字节的后面7位，最前面的一位统一规定为 0

Decimal	Hexadecimal	Binary	Octal	Char	Decimal	Hexadecimal	Binary	Octal	Char	Decimal	Hexadecimal	Binary	Octal	Char
0	0	0	0	[NULL]	48	30	110000	60	0	96	60	1100000	140	-
1	1	1	1	[START OF HEADING]	49	31	110001	61	1	97	61	1100001	141	a
2	2	10	2	[START OF TEXT]	50	32	110010	62	2	98	62	1100010	142	b
3	3	11	3	[END OF TEXT]	51	33	110011	63	3	99	63	1100011	143	c
4	4	100	4	[END OF TRANSMISSION]	52	34	110100	64	4	100	64	1100100	144	d
5	5	101	5	[ENQUIRY]	53	35	110101	65	5	101	65	1100101	145	e
6	6	110	6	[ACKNOWLEDGE]	54	36	110110	66	6	102	66	1100110	146	f
7	7	111	7	[BELL]	55	37	110111	67	7	103	67	1100111	147	g
8	8	1000	10	[BACKSPACE]	56	38	111000	70	8	104	68	1101000	150	h
9	9	1001	11	[HORIZONTAL TAB]	57	39	111001	71	9	105	69	1101001	151	i
10	A	1010	12	[LINE FEED]	58	3A	111010	72	:	106	6A	1101010	152	j
11	B	1011	13	[VERTICAL TAB]	59	3B	111011	73	;	107	6B	1101011	153	k
12	C	1100	14	[FORM FEED]	60	3C	111100	74	<	108	6C	1101100	154	l
13	D	1101	15	[CARRIAGE RETURN]	61	3D	111101	75	=	109	6D	1101101	155	m
14	E	1110	16	[SHIFT OUT]	62	3E	111110	76	>	110	6E	1101110	156	n
15	F	1111	17	[SHIFT IN]	63	3F	111111	77	?	111	6F	1101111	157	o
16	10	10000	20	[DATA LINK ESCAPE]	64	40	1000000	100	@	112	70	1110000	160	p
17	11	10001	21	[DEVICE CONTROL 1]	65	41	1000001	101	A	113	71	1110001	161	q
18	12	10010	22	[DEVICE CONTROL 2]	66	42	1000010	102	B	114	72	1110010	162	r
19	13	10011	23	[DEVICE CONTROL 3]	67	43	1000011	103	C	115	73	1110011	163	s
20	14	10100	24	[DEVICE CONTROL 4]	68	44	1000100	104	D	116	74	1110100	164	t
21	15	10101	25	[NEGATIVE ACKNOWLEDGE]	69	45	1000101	105	E	117	75	1110101	165	u
22	16	10110	26	[SYNCHRONOUS IDLE]	70	46	1000110	106	F	118	76	1110110	166	v
23	17	10111	27	[ENG OF TRANS. BLOCK]	71	47	1000111	107	G	119	77	1110111	167	w
24	18	11000	30	[CANCEL]	72	48	1001000	110	H	120	78	1111000	170	x
25	19	11001	31	[END OF MEDIUM]	73	49	1001001	111	I	121	79	1111001	171	y
26	1A	11010	32	[SUBSTITUTE]	74	4A	1001010	112	J	122	7A	1111010	172	z
27	1B	11011	33	[ESCAPE]	75	4B	1001011	113	K	123	7B	1111011	173	{
28	1C	11100	34	[FILE SEPARATOR]	76	4C	1001100	114	L	124	7C	1111100	174	
29	1D	11101	35	[GROUP SEPARATOR]	77	4D	1001101	115	M	125	7D	1111101	175	}
30	1E	11110	36	[RECORD SEPARATOR]	78	4E	1001110	116	N	126	7E	1111110	176	~
31	1F	11111	37	[UNIT SEPARATOR]	79	4F	1001111	117	O	127	7F	1111111	177	[DEL]
32	20	100000	40	[SPACE]	80	50	1010000	120	P					
33	21	100001	41	!	81	51	1010001	121	Q					
34	22	100010	42	"	82	52	1010010	122	R					
35	23	100011	43	#	83	53	1010011	123	S					
36	24	100100	44	\$	84	54	1010100	124	T					
37	25	100101	45	%	85	55	1010101	125	U					
38	26	100110	46	&	86	56	1010110	126	V					
39	27	100111	47	'	87	57	1010111	127	W					
40	28	101000	50	(88	58	1011000	130	X					
41	29	101001	51)	89	59	1011001	131	Y					
42	2A	101010	52	*	90	5A	1011010	132	Z					
43	2B	101011	53	+	91	5B	1011011	133	[
44	2C	101100	54	,	92	5C	1011100	134	\					
45	2D	101101	55	-	93	5D	1011101	135]					
46	2E	101110	56	.	94	5E	1011110	136	^					
47	2F	101111	57	/	95	5F	1011111	137	_					

范例：查看 ascii 表

```
[root@centos8 ~]#dnf -y install man-pages
[root@centos8 ~]#man ascii
```

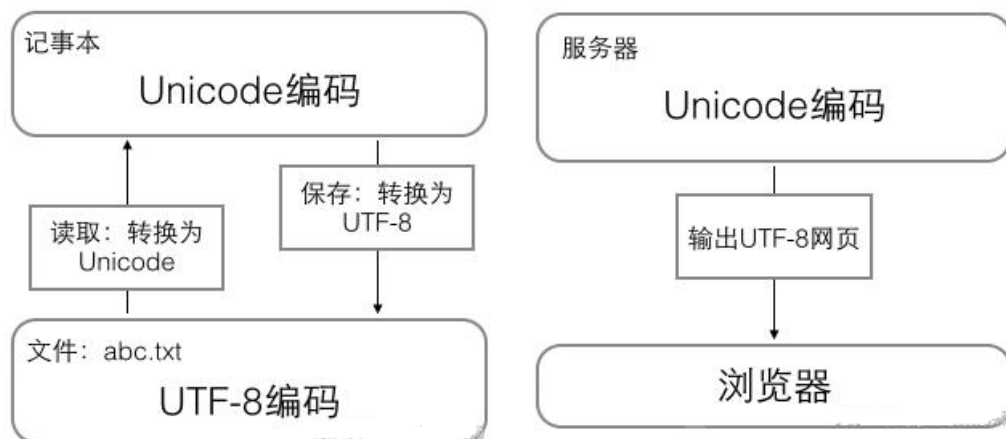
```
10.0.0.8 Xshell 5
文件(F) 编辑(E) 查看(V) 工具(T) 选项卡(B) 窗口(W) 帮助(H)

1 10.0.0.8

000 0 00 NUL '\0' (null character) 100 64 40 @
001 1 01 SOH (start of heading) 101 65 41 A
002 2 02 STX (start of text) 102 66 42 B
003 3 03 ETX (end of text) 103 67 43 C
004 4 04 EOT (end of transmission) 104 68 44 D
005 5 05 ENQ (enquiry) 105 69 45 E
006 6 06 ACK (acknowledge) 106 70 46 F
007 7 07 BEL '\a' (bell) 107 71 47 G
010 8 08 BS '\b' (backspace) 110 72 48 H
011 9 09 HT '\t' (horizontal tab) 111 73 49 I
012 10 0A LF '\n' (new line) 112 74 4A J
013 11 0B VT '\v' (vertical tab) 113 75 4B K
014 12 0C FF '\f' (form feed) 114 76 4C L
015 13 0D CR '\r' (carriage ret) 115 77 4D M
016 14 0E SO (shift out) 116 78 4E N
017 15 0F SI (shift in) 117 79 4F O
020 16 10 DLE (data link escape) 120 80 50 P
021 17 11 DC1 (device control 1) 121 81 51 Q
022 18 12 DC2 (device control 2) 122 82 52 R
023 19 13 DC3 (device control 3) 123 83 53 S
024 20 14 DC4 (device control 4) 124 84 54 T
025 21 15 NAK (negative ack.) 125 85 55 U
026 22 16 SYN (synchronous idle) 126 86 56 V
027 23 17 ETB (end of trans. blk) 127 87 57 W
030 24 18 CAN (cancel) 130 88 58 X
031 25 19 EM (end of medium) 131 89 59 Y
032 26 1A SUB (substitute) 132 90 5A Z
033 27 1B ESC (escape) 133 91 5B [
034 28 1C FS (file separator) 134 92 5C \
035 29 1D GS (group separator) 135 93 5D ]
036 30 1E RS (record separator) 136 94 5E ^
037 31 1F US (unit separator) 137 95 5F _
040 32 20 SPACE 140 96 60 `
041 33 21 ! 141 97 61 a
042 34 22 " 142 98 62 b
043 35 23 # 143 99 63 c
044 36 24 $ 144 100 64 d
045 37 25 % 145 101 65 e
046 38 26 & 146 102 66 f
047 39 27 ' 147 103 67 g
050 40 28 ( 150 104 68 h
051 41 29 ) 151 105 69 i
052 42 2A * 152 106 6A j
053 43 2B + 153 107 6B k
054 44 2C , 154 108 6C l
055 45 2D - 155 109 6D m
056 46 2E . 156 110 6E n
057 47 2F / 157 111 6F o

060 48 30 0 160 112 70 p
061 49 31 1 161 113 71 q
062 50 32 2 162 114 72 r
063 51 33 3 163 115 73 s
064 52 34 4 164 116 74 t
065 53 35 5 165 117 75 u
066 54 36 6 166 118 76 v
067 55 37 7 167 119 77 w
070 56 38 8 170 120 78 x
071 57 39 9 171 121 79 y
072 58 3A : 172 122 7A z
073 59 3B ; 173 123 7B {
074 60 3C < 174 124 7C [
075 61 3D = 175 125 7D ]
076 62 3E > 176 126 7E ~
077 63 3F ? 177 127 7F DEL

080 64 40 @ 180 128 80 P
081 65 41 A 181 129 81 Q
082 66 42 B 182 130 82 R
083 67 43 C 183 131 83 S
084 68 44 D 184 132 84 T
085 69 45 E 185 133 85 U
086 70 46 F 186 134 86 V
087 71 47 G 187 135 87 W
088 72 48 H 188 136 88 X
089 73 49 I 189 137 89 Y
090 74 4A J 190 138 8A Z
091 75 4B K 191 139 8B [
092 76 4C L 192 140 8C \
093 77 4D M 193 141 8D ]
094 78 4E N 194 142 8E ^
095 79 4F O 195 143 8F _
096 80 50 P 196 144 90 `
097 81 51 Q 197 145 91 a
098 82 52 R 198 146 92 b
099 83 53 S 199 147 93 c
100 84 54 T 200 148 94 d
101 85 55 U 201 149 95 e
102 86 56 V 202 150 96 f
103 87 57 W 203 151 97 g
104 88 58 X 204 152 98 h
105 89 59 Y 205 153 99 i
106 90 5A Z 206 154 100 j
107 91 5B [ 207 155 101 k
108 92 5C \ 208 156 102 l
109 93 5D ] 209 157 103 m
110 94 5E ^ 210 158 104 n
111 95 5F _ 211 159 105 o
112 96 60 ` 212 160 106 p
113 97 61 a 213 161 107 q
114 98 62 b 214 162 108 r
115 99 63 c 215 163 109 s
116 100 64 d 216 164 110 t
117 101 65 e 217 165 111 u
118 102 66 f 218 166 112 v
119 103 67 g 219 167 113 w
120 104 68 h 220 168 114 x
121 105 69 i 221 169 115 y
122 106 6A j 222 170 116 z
123 107 6B k 223 171 117 {
124 108 6C l 224 172 118 [
125 109 6D m 225 173 119 ]
126 110 6E n 226 174 120 ^
127 111 6F o 227 175 121 _
228 176 122 `
229 177 123 a
230 178 124 b
231 179 125 c
232 180 126 d
233 181 127 e
234 182 128 f
235 183 129 g
236 184 130 h
237 185 131 i
238 186 132 j
239 187 133 k
240 188 134 l
241 189 135 m
242 190 136 n
243 191 137 o
244 192 138 p
245 193 139 q
246 194 140 r
247 195 141 s
248 196 142 t
249 197 143 u
250 198 144 v
251 199 145 w
252 200 146 x
253 201 147 y
254 202 148 z
255 203 149 {
256 204 150 [
257 205 151 ]
258 206 152 ^
259 207 153 _
260 208 154 `
261 209 155 a
262 210 156 b
263 211 157 c
264 212 158 d
265 213 159 e
266 214 160 f
267 215 161 g
268 216 162 h
269 217 163 i
270 218 164 j
271 219 165 k
272 220 166 l
273 221 167 m
274 222 168 n
275 223 169 o
276 224 170 p
277 225 171 q
278 226 172 r
279 227 173 s
280 228 174 t
281 229 175 u
282 230 176 v
283 231 177 w
284 232 178 x
285 233 179 y
286 234 180 z
287 235 181 {
288 236 182 [
289 237 183 ]
290 238 184 ^
291 239 185 _
292 240 186 `
293 241 187 a
294 242 188 b
295 243 189 c
296 244 190 d
297 245 191 e
298 246 192 f
299 247 193 g
300 248 194 h
301 249 195 i
302 250 196 j
303 251 197 k
304 252 198 l
305 253 199 m
306 254 200 n
307 255 201 o
308 256 202 p
309 257 203 q
310 258 204 r
311 259 205 s
312 260 206 t
313 261 207 u
314 262 208 v
315 263 209 w
316 264 210 x
317 265 211 y
318 266 212 z
319 267 213 {
320 268 214 [
321 269 215 ]
322 270 216 ^
323 271 217 _
324 272 218 `
325 273 219 a
326 274 220 b
327 275 221 c
328 276 222 d
329 277 223 e
330 278 224 f
331 279 225 g
332 280 226 h
333 281 227 i
334 282 228 j
335 283 229 k
336 284 230 l
337 285 231 m
338 286 232 n
339 287 233 o
340 288 234 p
341 289 235 q
342 290 236 r
343 291 237 s
344 292 238 t
345 293 239 u
346 294 240 v
347 295 241 w
348 296 242 x
349 297 243 y
350 298 244 z
351 299 245 {
352 300 246 [
353 301 247 ]
354 302 248 ^
355 303 249 _
356 304 250 `
357 305 251 a
358 306 252 b
359 307 253 c
360 308 254 d
361 309 255 e
362 310 256 f
363 311 257 g
364 312 258 h
365 313 259 i
366 314 260 j
367 315 261 k
368 316 262 l
369 317 263 m
370 318 264 n
371 319 265 o
372 320 266 p
373 321 267 q
374 322 268 r
375 323 269 s
376 324 270 t
377 325 271 u
378 326 272 v
379 327 273 w
380 328 274 x
381 329 275 y
382 330 276 z
383 331 277 {
384 332 278 [
385 333 279 ]
386 334 280 ^
387 335 281 _
388 336 282 `
389 337 283 a
390 338 284 b
391 339 285 c
392 340 286 d
393 341 287 e
394 342 288 f
395 343 289 g
396 344 290 h
397 345 291 i
398 346 292 j
399 347 293 k
400 348 294 l
401 349 295 m
402 350 296 n
403 351 297 o
404 352 298 p
405 353 299 q
406 354 300 r
407 355 301 s
408 356 302 t
409 357 303 u
410 358 304 v
411 359 305 w
412 360 306 x
413 361 307 y
414 362 308 z
415 363 309 {
416 364 310 [
417 365 311 ]
418 366 312 ^
419 367 313 _
420 368 314 `
421 369 315 a
422 370 316 b
423 371 317 c
424 372 318 d
425 373 319 e
426 374 320 f
427 375 321 g
428 376 322 h
429 377 323 i
430 378 324 j
431 379 325 k
432 380 326 l
433 381 327 m
434 382 328 n
435 383 329 o
436 384 330 p
437 385 331 q
438 386 332 r
439 387 333 s
440 388 334 t
441 389 335 u
442 390 336 v
443 391 337 w
444 392 338 x
445 393 339 y
446 394 340 z
447 395 341 {
448 396 342 [
449 397 343 ]
450 398 344 ^
451 399 345 _
452 400 346 `
453 401 347 a
454 402 348 b
455 403 349 c
456 404 350 d
457 405 351 e
458 406 352 f
459 407 353 g
460 408 354 h
461 409 355 i
462 410 356 j
463 411 357 k
464 412 358 l
465 413 359 m
466 414 360 n
467 415 361 o
468 416 362 p
469 417 363 q
470 418 364 r
471 419 365 s
472 420 366 t
473 421 367 u
474 422 368 v
475 423 369 w
476 424 370 x
477 425 371 y
478 426 372 z
479 427 373 {
480 428 374 [
481 429 375 ]
482 430 376 ^
483 431 377 _
484 432 378 `
485 433 379 a
486 434 380 b
487 435 381 c
488 436 382 d
489 437 383 e
490 438 384 f
491 439 385 g
492 440 386 h
493 441 387 i
494 442 388 j
495 443 389 k
496 444 390 l
497 445 391 m
498 446 392 n
499 447 393 o
500 448 394 p
501 449 395 q
502 450 396 r
503 451 397 s
504 452 398 t
505 453 399 u
506 454 400 v
507 455 401 w
508 456 402 x
509 457 403 y
510 458 404 z
511 459 405 {
512 460 406 [
513 461 407 ]
514 462 408 ^
515 463 409 _
516 464 410 `
517 465 411 a
518 466 412 b
519 467 413 c
520 468 414 d
521 469 415 e
522 470 416 f
523 471 417 g
524 472 418 h
525 473 419 i
526 474 420 j
527 475 421 k
528 476 422 l
529 477 423 m
530 478 424 n
531 479 425 o
532 480 426 p
533 481 427 q
534 482 428 r
535 483 429 s
536 484 430 t
537 485 431 u
538 486 432 v
539 487 433 w
540 488 434 x
541 489 435 y
542 490 436 z
543 491 437 {
544 492 438 [
545 493 439 ]
546 494 440 ^
547 495 441 _
548 496 442 `
549 497 443 a
550 498 444 b
551 499 445 c
552 500 446 d
553 501 447 e
554 502 448 f
555 503 449 g
556 504 450 h
557 505 451 i
558 506 452 j
559 507 453 k
560 508 454 l
561 509 455 m
562 510 456 n
563 511 457 o
564 512 458 p
565 513 459 q
566 514 460 r
567 515 461 s
568 516 462 t
569 517 463 u
570 518 464 v
571 519 465 w
572 520 466 x
573 521 467 y
574 522 468 z
575 523 469 {
576 524 470 [
577 525 471 ]
578 526 472 ^
579 527 473 _
580 528 474 `
581 529 475 a
582 530 476 b
583 531 477 c
584 532 478 d
585 533 479 e
586 534 480 f
587 535 481 g
588 536 482 h
589 537 483 i
590 538 484 j
591 539 485 k
592 540 486 l
593 541 487 m
594 542 488 n
595 543 489 o
596 544 490 p
597 545 491 q
598 546 492 r
599 547 493 s
600 548 494 t
601 549 495 u
602 550 496 v
603 551 497 w
604 552 498 x
605 553 499 y
606 554 500 z
607 555 501 {
608 556 502 [
609 557 503 ]
610 558 504 ^
611 559 505 _
612 560 506 `
613 561 507 a
614 562 508 b
615 563 509 c
616 564 510 d
617 565 511 e
618 566 512 f
619 567 513 g
620 568 514 h
621 569 515 i
622 570 516 j
623 571 517 k
624 572 518 l
625 573 519 m
626 574 520 n
627 575 521 o
628 576 522 p
629 577 523 q
630 578 524 r
631 579 525 s
632 580 526 t
633 581 527 u
634 582 528 v
635 583 529 w
636 584 530 x
637 585 531 y
638 586 532 z
639 587 533 {
640 588 534 [
641 589 535 ]
642 590 536 ^
643 591 537 _
644 592 538 `
645 593 539 a
646 594 540 b
647 595 541 c
648 596 542 d
649 597 543 e
650 598 544 f
651 599 545 g
652 600 546 h
653 601 547 i
654 602 548 j
655 603 549 k
656 604 550 l
657 605 551 m
658 606 552 n
659 607 553 o
660 608 554 p
661 609 555 q
662 610 556 r
663 611 557 s
664 612 558 t
665 613 559 u
666 614 560 v
667 615 561 w
668 616 562 x
669 617 563 y
670 618 564 z
671 619 565 {
672 620 566 [
673 621 567 ]
674 622 568 ^
675 623 569 _
676 624 570 `
677 625 571 a
678 626 572 b
679 627 573 c
680 628 574 d
681 629 575 e
682 630 576 f
683 631 577 g
684 632 578 h
685 633 579 i
686 634 580 j
687 635 581 k
688 636 582 l
689 637 583 m
690 638 584 n
691 639 585 o
692 640 586 p
693 641 587 q
694 642 588 r
695 643 589 s
696 644 590 t
697 645 591 u
698 646 592 v
699 647 593 w
700 648 594 x
701 649 595 y
702 650 596 z
703 651 597 {
704 652 598 [
705 653 599 ]
706 654 600 ^
707 655 601 _
708 656 602 `
709 657 603 a
710 658 604 b
711 659 605 c
712 660 606 d
713 661 607 e
714 662 608 f
715 663 609 g
716 664 610 h
717 665 611 i
718 666 612 j
719 667 613 k
720 668 614 l
721 669 615 m
722 670 616 n
723 671 617 o
724 672 618 p
725 673 619 q
726 674 620 r
727 675 621 s
728 676 622 t
729 677 623 u
730 678 624 v
731 679 625 w
732 680 626 x
733 681 627 y
734 682 628 z
735 683 629 {
736 684 630 [
737 685 631 ]
738 686 632 ^
739 687 633 _
740 688 634 `
741 689 635 a
742 690 636 b
743 691 637 c
744 692 638 d
745 693 639 e
746 694 640 f
747 695 641 g
748 696 642 h
749 697 643 i
750 698 644 j
751 699 645 k
752 700 646 l
753 701 647 m
754 702 648 n
755 703 649 o
756 704 650 p
757 705 651 q
758 706 652 r
759 707 653 s
760 708 654 t
761 709 655 u
762 710 656 v
763 711 657 w
764 712 658 x
765 713 659 y
766 714 660 z
767 715 661 {
768 716 662 [
769 717 663 ]
770 718 664 ^
771 719 665 _
772 720 666 `
773 721 667 a
774 722 668 b
775 723 669 c
776 724 670 d
777 725 671 e
778 726 672 f
779 727 673 g
780 728 674 h
781 729 675 i
782 730 676 j
783 731 677 k
784 732 678 l
785 733 679 m
786 734 680 n
787 735 681 o
788 736 682 p
789 737 683 q
790 738 684 r
791 739 685 s
792 740 686 t
793 741 687 u
794 742 688 v
795 743 689 w
796 744 690 x
797 745 691 y
798 746 692 z
799 747 693 {
800 748 694 [
801 749 695 ]
802 750 696 ^
803 751 697 _
804 752 698 `
805 753 699 a
806 754 700 b
807 755 701 c
808 756 702 d
809 757 703 e
810 758 704 f
811 759 705 g
812 760 706 h
813 761 707 i
814 762 708 j
815 763 709 k
816 764 710 l
817 765 711 m
818 766 712 n
819 767 713 o
820 768 714 p
821 769 715 q
822 770 716 r
823 771 717 s
824 772 718 t
825 773 719 u
826 774 720 v
827 775 721 w
828 776 722 x
829 777 723 y
830 778 724 z
831 779 725 {
832 780 726 [
833 781 727 ]
834 782 728 ^
835 783 729 _
836 784 730 `
837 785 731 a
838 786 732 b
839 787 733 c
840 788 734 d
841 789 735 e
842 790 736 f
843 791 737 g
844 792 738 h
845 793 739 i
846 794 740 j
847 795 741 k
848 796 742 l
849 797 743 m
850 798 744 n
851 799 745 o
852 800 746 p
853 801 747 q
854 802 748 r
855 803 749 s
856 804 750 t
857 805 751 u
858 806 752 v
859 807 753 w
860 808 754 x
861 809 755 y
862 810 756 z
863 811 757 {
864 812 758 [
865 813 759 ]
866 814 760 ^
867 815 761 _
868 816 762 `
869 817 763 a
870 818 764 b
871 819 765 c
872 820 766 d
873 821 767 e
874 822 768 f
875 823 769 g
876 824 770 h
877 825 771 i
878 826 772 j
879 827 773 k
880 828 774 l
881 829 775 m
882 830 776 n
883 831 777 o
884 832 778 p
885 833 779 q
886 834 780 r
887 835 781 s
888 836 782 t
889 837 783 u
890 838 784 v
891 839 785 w
892 840 786 x
893 841 787 y
894 842 788 z
895 843 789 {
896 844 790 [
897 845 791 ]
898 846 792 ^
899 847 793 _
900 848 794 `
901 849 795 a
902 850 796 b
903 851 797 c
904 852 798 d
905 853 799 e
906 854 800 f
907 855 801 g
908 856 802 h
909 857 803 i
910 858 804 j
911 859 805 k
912 860 806 l
913 861 807 m
914 862 808 n
915 863 809 o
916 864 810 p
917 865 811 q
918 866 812 r
919 867 813 s
920 868 814 t
921 869 815 u
922 870 816 v
923 871 817 w
924 872 818 x
925 873 819 y
926 874 820 z
927 875 821 {
928 876 822 [
929 877 823 ]
930 878 824 ^
931 879 825 _
932 880 826 `
933 881 827 a
934 882 828 b
935 883 829 c
936 884 830 d
937 885 831 e
938 886 832 f
939 887 833 g
940 888 834 h
941 889 835 i
942 890 836 j
943 891 837 k
944 892 838 l
945 893 839 m
946 894 840 n
947 895 841 o
948 896 842 p
949 897 843 q
950 898 844 r
951 899 845 s
952 900 846 t
953 901 847 u
954 902 848 v
955 903 849 w
956 904 850 x
957 905 851 y
958 906 852 z
959 907 853 {
960 908 854 [
961 909 855 ]
```

UTF-8 是目前互联网上使用最广泛的一种 Unicode 编码方式，可变长存储。使用 1 - 4 个字节表示一个字符，根据字符的不同变换长度。编码规则如下：

对于单个字节的字符，第一位设为 0，后面的 7 位对应这个字符的 Unicode 码。因此，对于英文中的 0 - 127 号字符，与 ASCII 码完全相同。这意味着 ASCII 码的文档可用 UTF-8 编码打开

对于需要使用 N 个字节来表示的字符 (N > 1)，第一个字节的前 N 位都设为 1，第 N + 1 位设为 0，剩余的 N - 1 个字节的前两位都设位 10，剩下的二进制位则使用这个字符的 Unicode 码来填充

编码转换和查询参考链接：

<https://home.unicode.org/>

<https://unicode.yunser.com/unicode>

<http://www.chi2ko.com/tool/CJK.htm>

https://www.bejson.com/convert/unicode_chinese/

<https://javawind.net/tools/native2ascii.jsp?action=transform>

<http://tool.oschina.net/encode>

<http://web.chacuo.net/charsetscape>

Unicode和UTF-8

Unicode符号范围(十六进制)	UTF-8编码方式二进制)
0000 0000-0000 007F	0xxxxxxx
0000 0080-0000 07FF	110xxxxx 10xxxxxx
0000 0800-0000 FFFF	1110xxxx 10xxxxxx 10xxxxxx
0001 0000-0010 FFFF	11110xxx 10xxxxxx 10xxxxxx 10xxxxxx

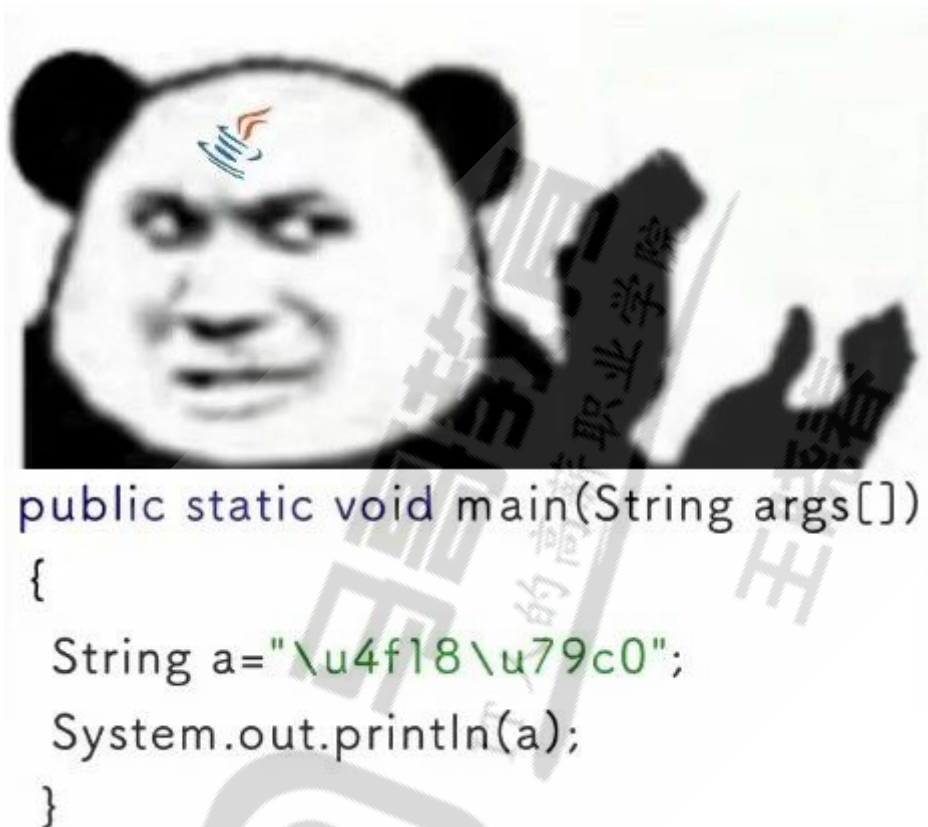
范例：Unicode 转换为 UTF-8

“汉”的 Unicode 码 0x6C49 (110 110001 001001)，需要三个字节存储，格式为： 1110xxxx 10xxxxxx 10xxxxxx，从后向前依次填充对应格式中的 x，多出的 x 用 0 补，得出UTF-8 编码为 11100110 10110001 10001001

“马”的 Unicode 码 0x9A6C (1001 101001 101100)，需要三个字节存储，格式为： 1110xxxx 10xxxxxx 10xxxxxx，从后向前依次填充对应格式中的 x，多出的 x 用 0 补，得出UTF-8 编码为 11101001 10101001 10101100

范例：修改LANG变量实现中文语言提示

```
[root@centos7 ~]#echo $LANG
en_US.UTF-8
[root@centos7 ~]#magedu
-bash: magedu: command not found
[root@centos7 ~]#LANG=zh_CN.UTF-8
[root@centos7 ~]#echo $LANG
zh_CN.UTF-8
[root@centos7 ~]#magedu
-bash: magedu: 未找到命令
```



1.9 命令行扩展和被括起来的集合

1.9.1 命令行扩展：`` 和 \$()

把一个命令的输出打印给另一个命令的参数,放在``中的一定是有输出信息的命令

```
$(COMMAND) 或 `COMMAND`
```

范例：比较“”，“”，``三者区别

```
[root@centos8 ~]#echo "echo $HOSTNAME"
echo centos8.localdomain
[root@centos8 ~]#echo 'echo $HOSTNAME'
echo $HOSTNAME
[root@centos8 ~]#echo `echo $HOSTNAME`
centos8.localdomain
```

#结论:

单引号: 强引用, 六亲不认, 变量和命令都不识别, 都当成了普通的字符串, "最傻"

双引号: 弱引用, 不能识别命令, 可以识别变量, "半傻不精"

反向单引号: 里面的内容必须是能执行的命令并且有输出信息, 变量和命令都识别, 并且会将反向单引号的内容当成命令进行执行后, 再交给调用反向单引号的命令继续, "最聪明"

范例:

```
[09:18:04 root@centos8 data]#echo "This system's name is $(hostname)"
This system's name is centos8.magedu.org
[09:18:55 root@centos8 data]#echo "I am `whoami`"
I am root
[09:15:37 root@centos8 data]#touch $(date +%F).log
[09:16:29 root@centos8 data]#ls
2019-12-13.log
[09:16:31 root@centos8 data]#ll
total 0
-rw-r--r--. 1 root root 0 Dec 13 09:16 2019-12-13.log
[09:16:34 root@centos8 data]#touch `date +%F`.txt
[09:17:14 root@centos8 data]#ll
total 0
-rw-r--r--. 1 root root 0 Dec 13 09:16 2019-12-13.log
-rw-r--r--. 1 root root 0 Dec 13 09:17 2019-12-13.txt
[09:17:15 root@centos8 data]#touch `hostname`-`date +%F`.log
[09:18:02 root@centos8 data]#ll
total 0
-rw-r--r--. 1 root root 0 Dec 13 09:16 2019-12-13.log
-rw-r--r--. 1 root root 0 Dec 13 09:17 2019-12-13.txt
-rw-r--r--. 1 root root 0 Dec 13 09:18 centos8.magedu.org-2019-12-13.log

[root@centos8 ~]#touch `date +%F_%H-%M-%S`.log
[root@centos8 ~]#touch `date -d '-1 day' +%F`.log
```

范例: \$() 和 ``

```
[root@centos8 ~]#ll `echo `date +%F`.txt`
-bash: .txt: command not found
ls: cannot access 'date': No such file or directory
ls: cannot access '+%F': No such file or directory

[root@centos8 ~]#ll $(echo $(date +%F).txt)
-rw-r--r-- 1 root root 0 Mar 20 09:55 2020-03-20.txt

[root@centos8 ~]#ll `echo $(date +%F).txt`
-rw-r--r-- 1 root root 0 Mar 20 09:55 2020-03-20.txt

[root@centos8 ~]#ll $(echo `date +%F`.txt)
-rw-r--r-- 1 root root 0 Mar 20 09:55 2020-03-20.txt
```

1.9.2 括号扩展: {}

{ } 可以实现打印重复字符串的简化形式

```
{元素1,元素2,元素3}
```

```
{元素1..元素2}
```

范例:

```
echo file{1,3,5} 结果为: file1 file3 file5
rm -f file{1,3,5}
echo {1..10}
echo {a..z}
echo {1..10}..2}
echo {000..20..2}
```

范例:

```
[root@centos8 ~]#echo {000..20..2}
000 002 004 006 008 010 012 014 016 018 020

[root@centos8 ~]#echo {a..z}..2}
a c e g i k m o q s u w y

[root@centos8 ~]#echo {A..z}
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [ ] ^ _ ` a b c d e f g h i
j k l m n o p q r s t u v w x y z
```

范例: 关闭和启用{}的扩展功能

```
[root@centos8 ~]#echo $-
himBHS
[root@centos8 ~]#echo {1..10}
1 2 3 4 5 6 7 8 9 10
[root@centos8 ~]#set +B
[root@centos8 ~]#echo $-
himHs
[root@centos8 ~]#echo {1..10}
{1..10}
[root@centos8 ~]#set -B
[root@centos8 ~]#echo $-
himBHS
[root@centos8 ~]#echo {1..10}
1 2 3 4 5 6 7 8 9 10
```

1.10 tab 键补全

tab 键可以实现命令及路径等补全, 提高输入效率, 避免出错

1.10.1 命令补全

- 内部命令：
- 外部命令：bash根据PATH环境变量定义的路径，自左而右在每个路径搜寻以给定命令命名的文件，第一次找到的命令即为要执行的命令
- 命令的子命令补全，需要安装 `bash-completion`

注意：用户给定的字符串只有一条唯一对应的命令，直接补全，否则，再次Tab会给出列表

范例：

```
[root@centos8 ~]#nmcli connection 2TAB
add      delete  edit    help    load    monitor show
clone    down    export  import  modify  reload  up
```

1.10.2 路径补全

把用户给出的字符串当做路径开头，并在其指定上级目录下搜索以指定的字符串开头的文件名

如果惟一：则直接补全

否则：再次Tab给出列表

1.10.3 双击Tab键

- command 2Tab 所有子命令或文件补全
- string 2Tab 以string开头命令
- /2Tab 显示所有根目录下一级目录，包括隐藏目录
- ./2Tab 当前目录下子目录，包括隐藏目录
- *2Tab 当前目录下子目录，不包括隐藏目录
- ~2Tab 所有用户列表
- \$2Tab 所有变量
- @2Tab /etc/hosts记录（centos7 不支持）
- =2Tab 相当于ls -A（centos7不支持）

1.11 命令行历史

当执行命令后，系统默认会在内存记录执行过的命令

当用户正常退出时，会将内存的命令历史存放对应历史文件中，默认是 `~/.bash_history`

登录shell时，会读取命令历史文件中记录下的命令加载到内存中

登录进shell后新执行的命令只会记录在内存的缓存区中；这些命令会用户正常退出时“追加”至命令历史文件中

利用命令历史。可以用它来重复执行命令，提高输入效率

命令：history

```
history [-c] [-d offset] [n]
history -anrw [filename]
history -ps arg [arg...]
```

常用选项：

-c: 清空命令历史
-d offset: 删除历史中指定的第offset个命令
n: 显示最近的n条历史
-a: 追加本次会话新执行的命令历史列表至历史文件
-r: 读历史文件附加到历史列表
-w: 保存历史列表到指定的历史文件
-n: 读历史文件中未读过的行到历史列表
-p: 展开历史参数成多行, 但不存在历史列表中
-s: 展开历史参数成一行, 附加在历史列表后

命令历史相关环境变量

HISTSIZE: 命令历史记录条数

HISTFILE: 指定历史文件, 默认为 ~/.bash_history

HISTFILESIZE: 命令历史文件记录历史的条数

HISTTIMEFORMAT="%F %T `whoami` " 显示时间和用户

HISTIGNORE="str1:str2*:..." 忽略str1命令, str2开头的历史

HISTCONTROL: 控制命令历史的记录方式

ignoredups 是默认值, 可忽略重复的命令, 连续且相同为“重复”

ignorespace 忽略所有以空白开头的命令

ignoreboth 相当于ignoredups, ignorespace的组合

erasedups 删除重复命令

持久保存变量

以上变量可以 export 变量名="值" 形式存放在 /etc/profile 或 ~/.bash_profile

范例:

```
[root@centos8 ~]#cat .bash_profile
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
    . ~/.bashrc
fi

# User specific environment and startup programs

PATH=$PATH:$HOME/bin

export PATH
export HISTCONTROL=ignoreboth
export HISTTIMEFORMAT="%F %T "

[root@centos8 ~]#history
1 2019-12-13 08:39:05 ls /data
2 2019-12-13 08:39:05 date
3 2019-12-13 08:39:05 vie0
4 2019-12-13 08:39:05 nano .bash_profile
5 2019-12-13 08:39:05 exit
```


1.12 调用命令行历史

#重复前一个命令方法

重复前一个命令使用上方向键，并回车执行

按 **!!** 并回车执行

输入 **!-1** 并回车执行

按 **Ctrl+p** 并回车执行

!:0 执行前一条命令（去除参数）

!n 执行**history**命令输出对应序号**n**的命令

!-n 执行**history**历史中倒数第**n**个命令

!string 重复前一个以“**string**”开头的命令

!?string 重复前一个包含**string**的命令

!string:p 仅打印命令历史，而不执行

!\$:p 打印输出 **!\$**（上一条命令的最后一个参数）的内容

!*:p 打印输出 **!***（上一条命令的所有参数）的内容

^string 删除上一条命令中的第一个**string**

^string1^string2 将上一条命令中的第一个**string1**替换为**string2**

!:gs/string1/string2 将上一条命令中所有的**string1**都替换为 **string2**

使用**up**（向上）和**down**（向下）键来上下浏览从前输入的命令

ctrl-r来在命令历史中搜索命令

（**reverse-i-search**）**``**:

Ctrl+g: 从历史搜索模式退出

#要重新调用前一个命令中最后一个参数

!\$ 表示前一个命令中最后一个参数

Esc, . 点击**Esc**键后松开，然后点击 **.** 键

Alt+ . 按住**Alt**键的同时点击 **.** 键

command !^ 利用上一个命令的第一个参数做**command**的参数

command !\$ 利用上一个命令的最后一个参数做**command**的参数

command !* 利用上一个命令的全部参数做**command**的参数

command !:n 利用上一个命令的第**n**个参数做**command**的参数

command !n:^ 调用第**n**条命令的第一个参数

command !n:\$ 调用第**n**条命令的最后一个参数

command !n:m 调用第**n**条命令的第**m**个参数

command !n:* 调用第**n**条命令的所有参数

command !string:^ 从命令历史中搜索以 **string** 开头的命令，并获取它的第一个参数

command !string:\$ 从命令历史中搜索以 **string** 开头的命令，并获取它的最后一个参数

command !string:n 从命令历史中搜索以 **string** 开头的命令，并获取它的第**n**个参数

command !string:* 从命令历史中搜索以 **string** 开头的命令，并获取它的所有参数

1.13 bash的快捷键

Ctrl + l 清屏，相当于**clear**命令

Ctrl + o 执行当前命令，并重新显示本命令

Ctrl + s 阻止屏幕输出，锁定

Ctrl + q 允许屏幕输出，解锁

Ctrl + c 终止命令

Ctrl + z 挂起命令

Ctrl + a 光标移到命令行首，相当于**Home**

Ctrl + e 光标移到命令行尾，相当于**End**

Ctrl + f 光标向右移动一个字符

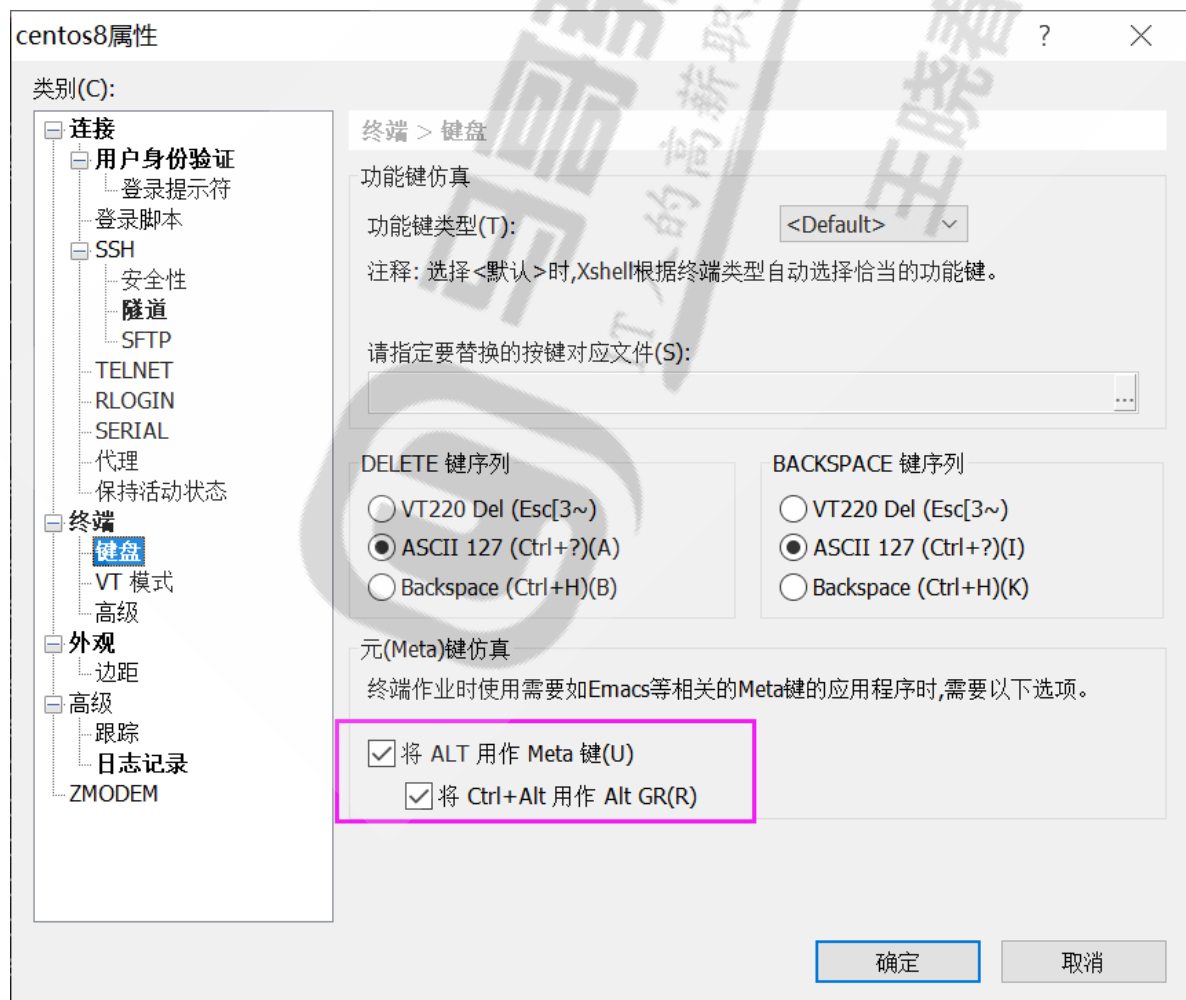
Ctrl + b 光标向左移动一个字符

Ctrl + xx 光标在命令行首和光标之间移动

Alt + f	光标向右移动一个单词尾
Alt + b	光标向左移动一个单词首
Ctrl + u	从光标处删除至命令行首
Ctrl + k	从光标处删除至命令行尾
Alt + r	删除当前整行
Ctrl + w	从光标处向左删除至单词首
Alt + d	从光标处向右删除至单词尾
Alt + Backspace	删除左边单词
Ctrl + d	删除光标处的一个字符
Ctrl + h	删除光标前的一个字符
Ctrl + y	将删除的字符粘贴至光标后
Alt + c	从光标处开始向右更改为首字母大写的单词
Alt + u	从光标处开始，将右边一个单词更改为大写
Alt + l	从光标处开始，将右边一个单词更改为小写
Ctrl + t	交换光标处和之前的字符位置
Alt + t	交换光标处和之前的单词位置
Alt + #	提示输入指定字符后，重复显示该字符#次

注意：Alt 组合快捷键经常和其它软件冲突

范例：xshell中启动 alt 键



2 获得帮助

获取帮助的能力决定了技术的能力！

多层次的帮助

- whatis
- command --help
- man and info
- /usr/share/doc/
- Red Hat documentation 、 Ubuntu documentation
- 软件项目网站
- 其它网站
- 搜索

2.1 whatis

whatis 使用数据库来显示命令的简短描述

此工具在系统刚安装后，不可立即使用，需要制作数据库后才可使用

执行下面命令生成数据库

```
#CentOS 7 版本以后
mandb
#CentOS 6 版本之前
makewhatis
```

范例：

```
[root@centos8 ~]#whatis cal
cal (1)          - display a calendar
cal (1p)         - print a calendar
[root@centos8 ~]#man -f cal
cal (1)          - display a calendar
cal (1p)         - print a calendar
```

范例：

```
[root@centos8 ~]#whatis ls
ls: nothing appropriate.
#生成man相关数据库
[root@centos8 ~]#mandb
Processing manual pages under /usr/share/man...
Updating index cache for path `/usr/share/man/mann'. wait...done.
Checking for stray cats under /usr/share/man...
...省略...
0 old database entries were purged.
[root@centos8 ~]#whatis ls
ls (1)          - list directory contents
```

2.2 查看命令的帮助

2.2.1 内部命令帮助

- help COMMAND
- man bash

范例：

```
[root@centos8 ~]#type history
history is a shell builtin
[root@centos8 ~]#help history
history: history [-c] [-d offset] [n] or history -anrw [filename] or history -ps
arg [arg...]
```

Display or manipulate the history list.

Display the history list with line numbers, prefixing each modified entry with a `*'. An argument of N lists only the last N entries.

Options:

- c clear the history list by deleting all of the entries
- d offset delete the history entry at position OFFSET.
- a append history lines from this session to the history file
- n read all history lines not already read from the history file and append them to the history list
- r read the history file and append the contents to the history list
- w write the current history to the history file
- p perform history expansion on each ARG and display the result without storing it in the history list
- s append the ARGS to the history list as a single entry

If FILENAME is given, it is used as the history file. Otherwise, if HISTFILE has a value, that is used, else ~/.bash_history.

If the HISTTIMEFORMAT variable is set and not null, its value is used as a format string for strftime(3) to print the time stamp associated with each displayed history entry. No time stamps are printed otherwise.

Exit Status:

Returns success unless an invalid option is given or an error occurs.

2.2.2 外部命令及软件帮助

- COMMAND --help 或 COMMAND -h
- 使用 man 手册(manual): man COMMAND
- 信息页: info COMMAND
- 程序自身的帮助文档: README、INSTALL、ChangeLog
- 程序官方文档
- 相关网站, 如: 技术论坛
- 搜索引擎

2.3 外部命令的--help 或 -h 选项

显示用法总结和参数列表, 大多数命令使用, 但并非所有的

范例:

```
[root@centos8 ~]#date --help
Usage: date [OPTION]... [+FORMAT]
or: date [-u|--utc|--universal] [MMDDhhmm[[CC]YY][.ss]]
Display the current time in the given FORMAT, or set the system date.
```

```
[root@centos8 ~]#cal -h
Usage:
cal [options] [[[day] month] year]
cal [options] <timestamp|monthname>

[root@centos8 ~]#strace --help
strace: invalid option -- '-'
Try 'strace -h' for more information.

[root@centos8 ~]#openssl --help
Invalid command '--help'; type "help" for a list.

[root@centos8 ~]#date -h
date: invalid option -- 'h'
Try 'date --help' for more information.

[root@centos8 ~]#shutdown -h
Shutdown scheduled for Fri 2020-03-20 08:26:27 CST, use 'shutdown -c' to cancel.
```

格式说明:

- [] 表示可选项
- CAPS或 <> 表示变化的数据
- ... 表示一个列表
- x |y| z 的意思是“x 或 y 或 z”
- -abc 的意思是 -a -b -c
- {} 表示分组

练习:

- 1、显示当前时间, 格式: 2016-06-18 10:20:30
- 2、显示前天是星期几
- 3、设置当前日期为2019-08-07 06:05:10

2.4 man 命令

man 提供命令帮助的文件,手册页存放在/usr/share/man

几乎每个命令都有man的“页面”

中文man需安装包

- man-pages
- man-pages-zh-CN

man 页面分组

不同类型的帮助称为不同的“章节”, 统称为Linux手册, man 1 man

- 1: 用户命令
- 2: 系统调用
- 3: C库调用
- 4: 设备文件及特殊文件
- 5: 配置文件格式
- 6: 游戏
- 7: 杂项

- 8: 管理类的命令
- 9: Linux 内核API

man命令的配置文件:

```
#CentOS 6 之前版 man 的配置文件
/etc/man.config
#CentOS 7 之后版 man 的配置文件
/etc/man_db.conf
#ubuntu man 的配置文件
/etc/manpath.config
```

格式:

```
MANPATH /PATH/TO/SOMEWHERE #指明man文件搜索位置
```

也可以指定位置下搜索COMMAND命令的手册页并显示

```
man -M /PATH/TO/SOMEWHERE COMMAND
```

查看man手册页

```
man [OPTION...] [SECTION] PAGE...
man [章节] keyword
```

man 帮助段落说明

- NAME 名称及简要说明
- SYNOPSIS 用法格式说明
- [] 可选内容
- <> 必选内容
- a|b 二选一
- {} 分组
- ... 同一内容可出现多次
- DESCRIPTION 详细说明
- OPTIONS 选项说明
- EXAMPLES 示例
- FILES 相关文件
- AUTHOR 作者
- COPYRIGHT 版本信息
- REPORTING BUGS bug信息
- SEE ALSO 其它帮助参考

man命令的操作方法: 使用less命令实现

- space, ^v, ^f, ^F: 向文件尾翻页
- b, ^b: 向文件首部翻页
- d, ^d: 向文件尾部翻半屏
- u, ^u: 向文件首部翻半屏
- RETURN, ^N, e, ^E or j or ^J: 向文件尾部翻一行
- y or ^Y or ^P or k or ^K: 向文件首部翻一行
- q: 退出
- #: 跳转至第#行
- 1G: 回到文件首部

- G: 翻至文件尾部
- /KEYWORD
以KEYWORD指定的字符串为关键字, 从当前位置向文件尾部搜索; 不区分字符大小写
n: 下一个
N: 上一个
- ?KEYWORD
以KEYWORD指定的字符串为关键字, 从当前位置向文件首部搜索; 不区分字符大小写
n: 跟搜索命令同方向, 下一个
N: 跟搜索命令反方向, 上一个

常用选项

- 列出所有帮助

```
man -a keyword
```

- 搜索man手册

```
#列出所有匹配的页面, 使用 whatis 数据库  
man -k keyword
```

- 相当于 whatis

```
man -f keyword
```

- 打印man帮助文件的路径

```
man -w [章节] keyword
```

范例:

```
[root@centos8 ~]#dnf install man-pages  
[root@centos8 ~]#man -w 1 passwd  
/usr/share/man/man1/passwd.1.gz  
[root@centos8 ~]#whatis passwd  
openssl-passwd (1ssl) - compute password hashes  
passwd (1) - update user's authentication tokens  
[root@centos8 ~]#man 1ssl openssl-passwd  
[root@centos8 ~]#man 7 ascii  
[root@centos8 ~]#man 7 utf8
```

范例: 查看passwd相关命令和文件,man帮助文件路径

```
[root@centos8 ~]#whereis passwd  
passwd: /usr/bin/passwd /etc/passwd /usr/share/man/man1/passwd.1.gz  
/usr/share/man/man5/passwd.5.gz
```

练习:

- 1、在本机字符终端登录时, 除显示原有信息外, 再显示当前登录终端号, 主机名和当前时间
- 2、今天18:30自动关机, 并提示用户

2.5 info

man常用于命令参考，GNU工具 info 适合通用文档参考

没有参数,列出所有的页面

info 页面的结构就像一个网站

每一页分为“节点”

链接节点之前 *

info 命令格式

```
info [ 命令 ]
```

导航info页

- 方向键, PgUp, PgDn 导航
- Tab键 移动到下一个链接
- d 显示主题目录
- Home 显示主题首部
- Enter进入 选定链接
- n/p/u/l 进入下/前/上一层/最后一个链接
- s 文字 文本搜索
- q 退出 info

2.6 命令自身提供的官方使用指南

```
/usr/share/doc
```

多数安装了的软件包的子目录,包括了这些软件的相关原理说明

常见文档: README INSTALL CHANGES

不适合其它地方的文档的位置

配置文件范例

HTML/PDF/PS 格式的文档

授权书详情

2.7 系统及第三方应用官方文档

2.7.1 通过在线文档获取帮助

<http://www.github.com>

<https://www.kernel.org/doc/html/latest/>

<http://httpd.apache.org>

<http://www.nginx.org>

<https://mariadb.com/kb/en>

<https://dev.mysql.com/doc/>

<http://tomcat.apache.org>

<https://jenkins.io/zh/doc/>

<https://kubernetes.io/docs/home/>

<https://docs.openstack.org/train/>

<http://www.python.org>

<http://php.net>

2.7.2 Linux官方在线文档和知识库

通过发行版官方的文档光盘或网站可以获得安装指南、部署指南、虚拟化指南等

<http://kbase.redhat.com>

<http://www.redhat.com/docs>

<http://access.redhat.com>

<https://help.ubuntu.com/lts/serverguide/index.html>

<http://tldp.org>

2.7.3 红帽全球技术支持服务

rhn.redhat.com或者本地卫星服务器/代理服务器

RHN账户为及其注册和基于网络管理的RHN用户

sosreport 收集所有系统上的日志信息的工具，并自动打成压缩包，方便技术支持人员和红帽全球支持提供分析问题依据

范例：

```
[root@centos8 ~]#dnf -y install sos
[root@centos8 ~]#sosreport
sosreport (version 3.6)
This command will collect diagnostic and configuration information from
this CentOS Linux system and installed applications.

An archive containing the collected information will be generated in
/var/tmp/sos.8fs1kg3w and may be provided to a CentOS support
representative.

Any information provided to CentOS will be treated in accordance with
the published support policies at:

    https://wiki.centos.org/

The generated archive may contain data considered sensitive and its
content should be reviewed by the originating organization before being
passed to any third party.

No changes will be made to system configuration.

Press ENTER to continue, or CTRL-C to quit.

Please enter the case id that you are generating this report for []: 2

Setting up archive ...
Setting up plugins ...
Running plugins. Please wait ...

    Finishing plugins                [Running: processor]
    Finished running plugins
    Creating compressed archive...

Your sosreport has been generated and saved in:
```

```
/var/tmp/sosreport-centos8-2-2020-03-20-x11ummc.tar.xz
```

The checksum is: ffbbe6fa9f009f4024e272fbce500bff

Please send this file to your support representative.

```
[root@centos8 sysconfig]#ll /var/tmp/sosreport-centos8-2-2020-03-20-x11ummc.tar.xz
-rw----- 1 root root 4409596 Mar 20 18:57 /var/tmp/sosreport-centos8-2-2020-03-20-x11ummc.tar.xz
[root@centos8 sysconfig]#cd
[root@centos8 ~]#ll /var/tmp/sosreport-centos8-2-2020-03-20-x11ummc.tar.xz
```

2.8 相关网站和搜索

<http://www.google.com>

```
openstack filetype:pdf
rhca site:redhat.com/docs
```

<http://bing.com>

<http://www.baidu.com>

<http://www.slideshare.net>

 马哥教育
IT人的高薪职业学院

祝大家学业有成

谢 谢

讲师：王晓春
邮箱：29308620@qq.com
电话：400-080-6560