解决端口占用：

netstat -tunlp | grep yourport

ps -ef | grep yourpid

kill -9 yourpid

查看日志：

从后向前看

less log.out shift+g跳到最后一行 ctrl+b,ctrl+f翻页

?word查找，n,shift+n上一个下一个 q退出

ls -al ll

cd /

当前目录：pwd

递归创建多个目录 :mkdir -p test2/test22 创建权限为777的目录:mkdir -m 777 test3

cat pkms.env | grep orcl75 -C 5 搜索pkms.env中"orcl75"的前后5行

ls -rt 根据时间排序

ps -ef | grep osa 查找osa进程

删除：rm [选项] 目标文件或目录

-f, --force    忽略不存在的文件，从不给出提示。

    -i, --interactive 进行交互式删除

    -r, -R, --recursive   指示rm将参数中列出的全部目录和子目录均递归地删除。

    -v, --verbose    详细显示进行的步骤

自定义回收站功能：

[root@localhost test]# myrm(){ D=/tmp/$(date +%Y%m%d%H%M%S); mkdir -p $D;  mv "$@" $D && echo "moved to $D ok"; }

[root@localhost test]# alias rm='myrm'

[root@localhost test]# rm 123.log

移动：mv [选项] 源文件或目录 目标文件或目录

-b ：若需覆盖文件，则覆盖前先行备份。

-f ：force 强制的意思，如果目标文件已经存在，不会询问而直接覆盖；

-i ：若目标文件 (destination) 已经存在时，就会询问是否覆盖！

-u ：若目标文件已经存在，且 source 比较新，才会更新(update)

复制：cp [选项] 源 目录 将aa文件夹下的所有文件复制到bb文件夹下：cp -r aa/\* bb

-b ： 为每个已存在的目标文件创建备份

-f, --force 不询问直接覆盖

-i, --interactive   覆盖前询问

-n, --no-clobber   不要覆盖已存在的文件

-R, -r, --recursive 递归复制目录及目录内的所有项目

touch:用来修改文件时间戳，或者新建一个不存在的文件

**创建不存在的文件** touch log2012.log log2013.log

**更新log.log的时间和log2012.log时间戳相同** touch -r log.log log2012.log

**设定文件的时间戳** touch -t 201211142234.50 log.log

cat:一次显示整个文件:cat filename

nl log2012.log

more**:从文件中查找第一个出现"day3"字符串的行，并从该处前两行开始显示输出** more +/day3 log2012.log

head：显示前面6行：head -n 6 log2014.log

tail: **循环查看文件内容**tail -f test.log

which: which命令的作用是，在PATH变量指定的路径中，搜索某个系统命令的位置，并且返回第一个搜索结果。也就是说，使用which命令，就可以看到某个系统命令是否存在，以及执行的到底是哪一个位置的命令 which pwd

whereis whereris tomcat

-b   定位可执行文件。

-m   定位帮助文件。

-s   定位源代码文件。

-u   搜索默认路径下除可执行文件、源代码文件、帮助文件以外的其它文件。

-B   指定搜索可执行文件的路径。

-M   指定搜索帮助文件的路径。

-S   指定搜索源代码文件的路径。

locate aa **查找和aa相关的所有文件**

find: find pathname -options [-print -exec -ok ...]

在当前目录查找 以.log结尾的文件。 "**.**"代表当前目录  find . -name "\*.log"

查找当目录，以.log结尾的普通文件 find . -type f -name "\*.log"

查找当前目录下\*.log并复制到test3目录 find . -name "\*.log" -exec cp {} test3 \

4．常见解压/压缩命令

tar

解包：tar xvf FileName.tar

打包：tar cvf FileName.tar DirName

（注：tar是打包，不是压缩！）

.gz

解压1：gunzip FileName.gz

解压2：gzip -d FileName.gz

压缩：gzip FileName

.tar.gz 和 .tgz

解压：tar zxvf FileName.tar.gz

压缩：tar zcvf FileName.tar.gz DirName

.bz2

解压1：bzip2 -d FileName.bz2

解压2：bunzip2 FileName.bz2

压缩： bzip2 -z FileName

.tar.bz2

解压：tar jxvf FileName.tar.bz2

压缩：tar jcvf FileName.tar.bz2 DirName

.bz

解压1：bzip2 -d FileName.bz

解压2：bunzip2 FileName.bz

压缩：未知

.tar.bz

解压：tar jxvf FileName.tar.bz

压缩：未知

.Z

解压：uncompress FileName.Z

压缩：compress FileName

.tar.Z

解压：tar Zxvf FileName.tar.Z

压缩：tar Zcvf FileName.tar.Z DirName

.zip

解压：unzip FileName.zip

压缩：zip FileName.zip DirName

.rar

解压：rar x FileName.rar

压缩：rar a FileName.rar DirName

chmod 更改文件权限

文件属主（u），文件属主同组用户（g），其他用户（o）

设定文件text的属性为：文件属主（u） 增加写权限;与文件属主同组用户（g） 增加写权限;其他用户（o） 删除执行权限：

chmod ug+w,o-x log2012.log

修改权限： chmod u=rwx,g=rx,o=x file  递归： chmod -R u+x test4

chgrp 更改文件群组

chgrp -R bin test6

chown 更改拥有者和群组

chown -R -v root:mail test6

jdk安装

**第一步　　卸载系统自带的OpenJDK以及相关的java文件**

查看 java -version

rpm -qa | grep java

删除openjdk

rpm -e --nodeps java-1.7.0-openjdk-1.7.0.111-2.6.7.8.el7.x86\_64
rpm -e --nodeps java-1.8.0-openjdk-1.8.0.102-4.b14.el7.x86\_64
rpm -e --nodeps java-1.8.0-openjdk-headless-1.8.0.102-4.b14.el7.x86\_64
rpm -e --nodeps java-1.7.0-openjdk-headless-1.7.0.111-2.6.7.8.el7.x86\_64

解压

tar -zxvf jdk-8u144-linux-x64.tar.gz

设置环境变量

vi /etc/profile

文件最后加上

#java environment

export JAVA\_HOME=/usr/java/jdk1.8.0\_144

export CLASSPATH=.:${JAVA\_HOME}/jre/lib/rt.jar:${JAVA\_HOME}/lib/dt.jar:${JAVA\_HOME}/lib/tools.jar

export PATH=$PATH:${JAVA\_HOME}/bin

让环境变量生效

source /etc/profile

tomcat安装

直接解压即可

sh startup.sh

sh shutdown.sh

vi编辑器：

vi filename

i 进入编辑模式

**:wq 保存并退出**

**:q! 不保存退出**

**crontab -u wmprodmy -l 查询wmprodmy的定时任务**

**crontab -u wmprodmy -e 编辑wmprodmy的定时任务**