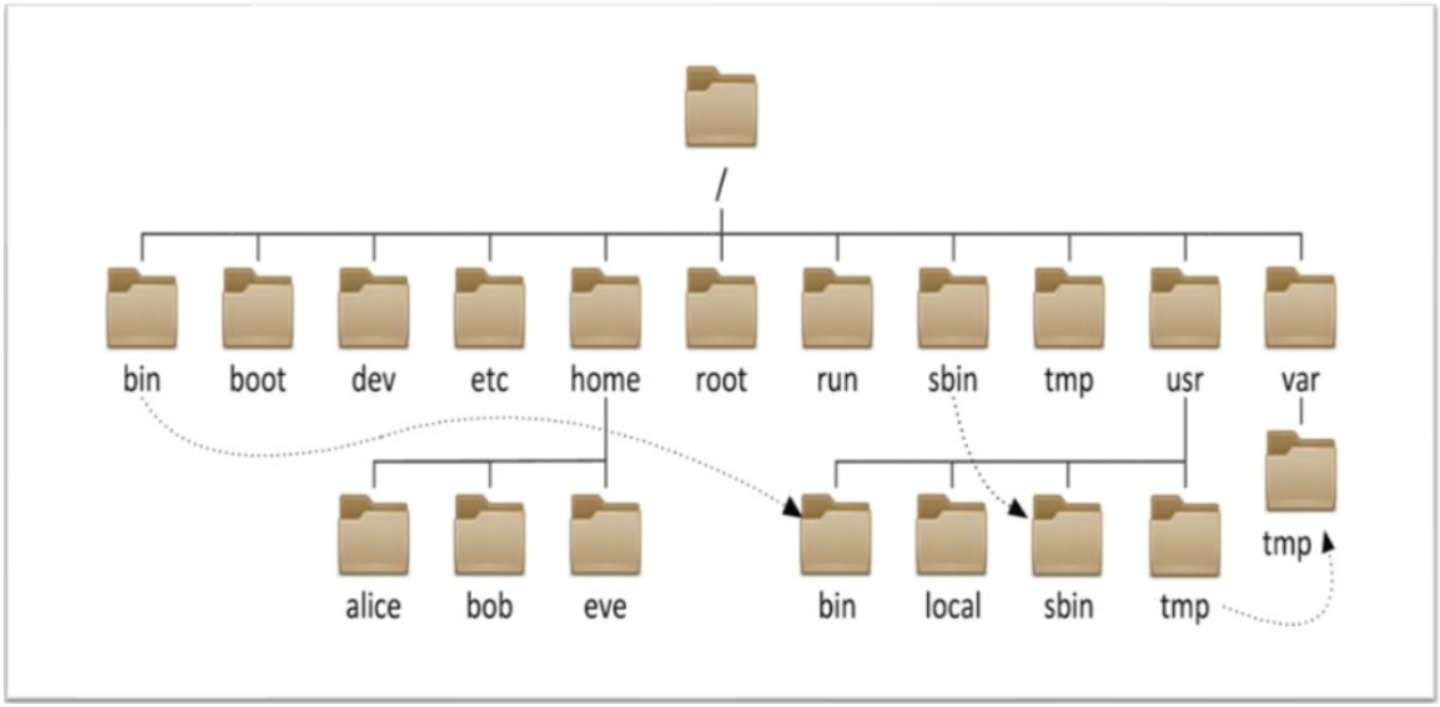
阶段二：Linux运维核心命令

1. Linux的目录结构

Linux的目录结构是一个树型结构，没有盘符这个概念, 只有一个根目录 /, 所有文件都在它下面，而不像Windows可以拥有多个盘符, 如 C盘、D盘、E盘



- Linux常见目录结构（高亮部分的要格外注意）

目录	作用
/bin	二进制命令所在的目录(普通命令 => 普通用户和超级管理员root)
/boot	系统引导程序所需要的文件目录，相当于Windows中的C盘
/dev	/device缩写，设备文件目录，磁盘，光驱 => /dev/sr0
/etc	系统配置文件目录，启动程序，几乎所有的软件都会把自己的配置文件安装在/etc中
/home	普通用户的家目录，默认用户数据存放目录，itheima账号 => /home/itheima文件夹
/lib	共享库文件和内核模块存放目录，软件安装、运行依赖库文件.a、.so文件
/mnt	临时挂载储存设备的挂载点，插入u盘、移动硬盘 => 先挂载 => /mnt中访问
/opt	额外的应用软件包， 安装qq、游戏、wps办公软件
/proc	process进程目录，操作系统运行时，进程信息和内核信息存放在这里

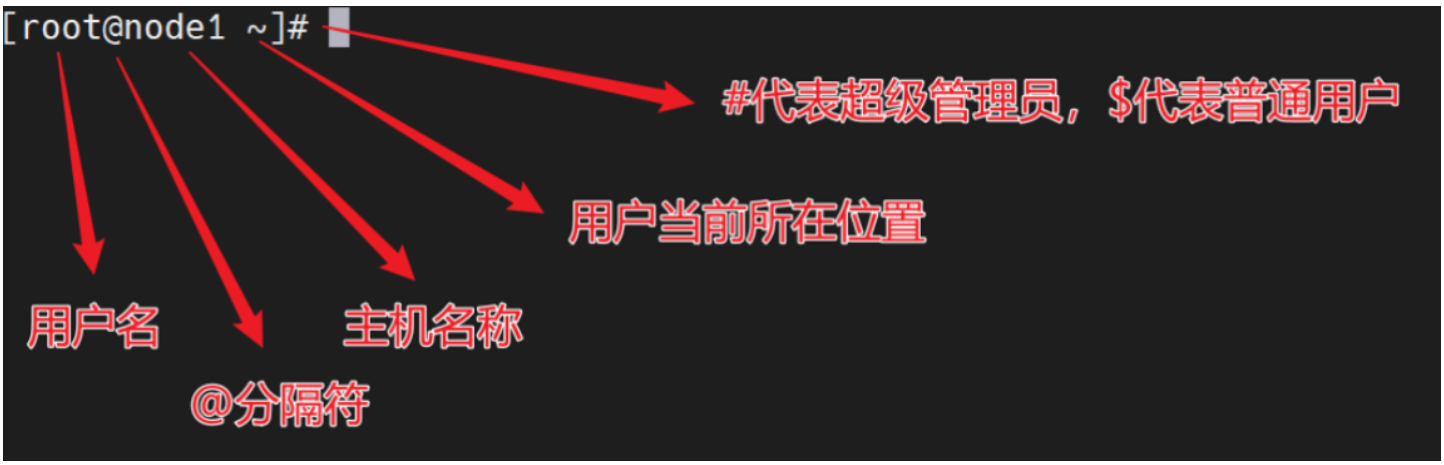
/root	Linux超级权限用户root的家目录，超级管理员root => /root
/sbin	和管理系统相关的命令，【超级管理员用】，s = super超级
/tmp	临时文件目录，这个目录被当作回收站使用
/usr	用户或系统软件应用程序目录，类似Windows中的Program files

2. Linux与目录相关命令

Linux目录的相关命令主要包括有pwd、ls、mkdir、cd、rmdir等， 如果把Linux的目录结构比作一个“文件柜”：

- `pwd` = 当前在文件柜的哪一层抽屉（显示当前位置）
- `ls` = 打开抽屉看看里面有什么（列出内容）
- `mkdir` = 增加了一个新的抽屉（创建目录）
- `cd` = 走进某个抽屉或文件夹（切换位置）
- `rmdir` = 扔掉一个空文件夹（删除空目录）

2.1 终端说明

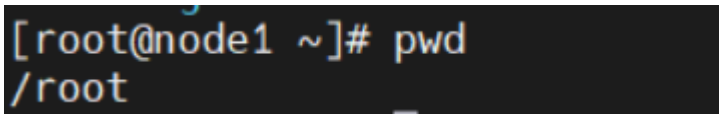


2.2 基本导航命令

2.2.1 定位与查看（pwd、ls）

pwd（print working directory）：

- 格式： pwd
- 作用： 查看当前所在位置



ls（list show）：

- 格式： ls [选项][路径]
- 作用： ls 是英文单词list show的简写, 其功能为列出目录的内容，是用户最常用的命令之一

ls常用选项

选项	含义
-a	all所有, 显示指定目录下所有子目录与文件, 包含隐藏文件
-l	list，以列表方式显示文件的详细信息
-h	配合 -l 以人性化的方式显示文件大小（文件大小 + 单位）

ls -l与ll显示信息说明：

2.2.2 切换与目录结构（cd、tree）

cd（change directory）：

- 作用：更改当前的工作目录
- 格式：

命令	含义
cd	切换到用户主目录（root用户主目录是/root,其他用户是/home/用户名）等价于 cd ~
cd 目录	切换到指定目录下 => cd /etc
cd ..	切换到上级目录

```
普及：路径有两种写法

绝对路径 ：从根目录一级一级向下移动，不能越级。如/home/itheima
例如：访问根目录下的usr目录下local目录下的nginx文件夹
cd /usr/local/nginx => 路径不需要记忆，用到的时候，直接按Tab键（自动补全）
```

```
6
7   相对路径 ：顾名思义，有一个参考点 => 以当前位置作为参考
```

tree：树形结构可视化辅助工具

- 安装 `tree` 命令

```
1   dnf -y install tree
2   # 查看目录结构
3   tree game_saves
```

输出示例:

```
game_saves
├── level1
├── level2
└── backup
    └── 2024
```

2.3 目录管理命令

2.3.1 创建目录（mkdir）

mkdir（make directory）：

- 作用：mkdir 命令用于创建目录
- 格式：

```
1   mkdir [-p] dirName
2
3   参数:
4   -p：一次创建多级目录
```

2.3.2 删除空目录（rmdir）【了解】

作用：rmdir命令用于删除空目录

```
1   rmdir dirName
```

2.4 综合练习

2.4.1 搭建一个【游戏存档】目录

任务：

1. 在根目录创建 `game_saves` 目录，进入它
2. 创建子目录 `level1`、`level2`、`backup`
3. 在 `backup` 中创建 `2023` 和 `2024` 目录
4. 删除 `backup/2023`，再尝试删除整个 `game_saves`（观察错误）

2.4.2 搭建一个family目录

结构要求：

```
family
├── parents
│   ├── dad
│   └── mom
└── children
    ├── son
    └── daughter
```

要求：用 `tree` 命令验证结构，最后删除整个目录（只用已学命令）。

3. Linux与文件相关命令

3.1 文件操作命令

3.1.1 创建文件（touch）

作用：touch命令创建文件

格式：touch 文件名

案例演示：

```
需求一：  在家目录下，创建一个hello.java
cd
touch hello.java

需求二：  在家目录下，创建一个a.txt
cd
touch a.txt
```

> 注：touch命令不仅可以创建.txt文本文档，还可以创建python/java/go等脚本程序。如`touch main.py`

3.1.2 复制 cp

作用：cp（copy）命令用来实现文件或者目录的复制

格式：cp [-r] 源文件位置 目标路径

参数	说明
-r	递归,复制文件夹必须添加

3.1.3 移动 mv

作用：mv（move）命令用于文件、目录的移动和重命名

格式：mv 源文件路径 目标路径

> 注意：mv没有任何选项，移动文件和文件夹都可以

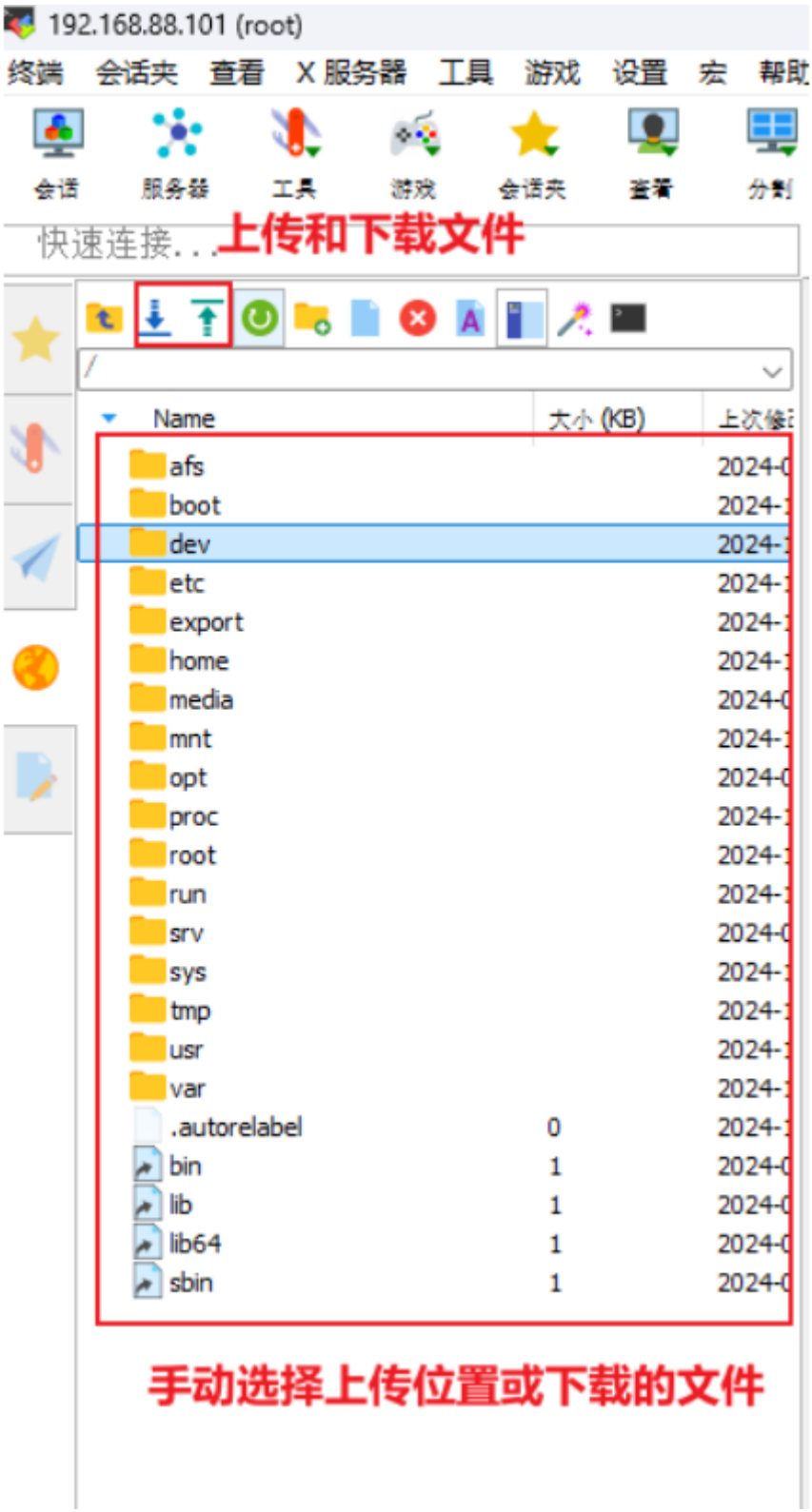
案例演示：

```
需求一：将 /root目录下的anaconda-ks.cfg文件,移动到 /export/data 目录下
cd ~
mv anaconda-ks.cfg /export/data

需求二：将 /export目录移动到 /root目录下
mv /export /root
```

重命名：

```
需求：将root用户家目录的a.txt文件，更名为 test.txt
cd ~
mv a.txt test.txt

需求二：将test.txt 移动到 / 目录下,并改名a.txt
cd ~
mv test.txt /a.txt
```

3.1.4 文件上传与下载



3.1.5 删除 rm

格式：rm [选项] 文件/目录路径

选项	说明
-r	是否启动递归删除方案（适用于删除目录的时候）
-f	是否需要强制删除，如果不添加， 会有一个询问的操作

3.1.6 文本查找相关命令 find

作用: 用于查找文档

语法: find 路径范围 选项1 选项1的值 [选项2 选项2 的值…]

选项	说明
-name	按照文档名称进行搜索（支持模糊搜索） * : 通配符，匹配任意个任意字符
-type	按照文档的类型进行搜索，文档类型的值，f（file)表示文件，d(directory)表示文件夹
-size	用于根据文件大小查找文件。格式为 `[+/-]n[单位]` `+` 和 `-` 前缀: • `+n`：查找大于 `n` 的文件。 • `-n`：查找小于 `n` 的文件。 单位说明: • `c`：字节（默认单位） • `k`：千字节（KB） • `M`：兆字节（MB） • `G`：吉字节（GB） • `b`：512字节块（不常用）
-mtime	用于根据文件的修改时间（modification time）来筛选文件,参数值是一个整数，表示天数 • `-mtime +n`：查找修改时间超过 `n` 天前的文件 • `-mtime -n`：查找修改时间在 `n` 天以内的文件 • `-mtime n`：查找修改时间正好是 `n` 天前的文件
-exec	用于对 `find` 查找到的文件执行指定的命令. 它的语法比较特殊，需要用 `{}` 表示查找到的文件，并用 `\;` 结束命令 例如: 查找所有 `.log` 文件并删除 find /var/log -name "*.log" -exec rm -f {} \;

3.2 文件操作命令综合练习

3.2.1 日志文件管理与备份

某服务器日志目录 /var/log/app 需要定期清理和备份。要求：

- 创建 5 个不同日期的日志文件（如 access_20231001.log、access_20231002.log ...）
- 将 3 天前的日志文件复制到 /backup/old_logs 目录
- 删除 7 天前的日志文件

3.2.2 批量重命名备份文件

某配置目录 /etc/nginx/conf.d 下有多个 .conf 文件，要求：

- 将所有 .conf 文件备份到 /etc/nginx/backup，备份文件名添加 .bak 后缀（如 site1.conf.bak）
- 将原目录中所有 .conf 文件重命名为 .conf.old

3.2.3 清理临时文件

某临时目录 /tmp/userdata 中存在大量临时文件（文件名包含 tmp_ 前缀），要求：

- 查找所有 24 小时内修改过的 tmp_* 文件并移动到 /tmp/recent
- 删除超过 30 天未修改的 tmp_* 文件

3.2.4 查找特定文件并归档

某服务器 /home 目录下可能存在用户遗留的大文件（超过 100MB），要求：

- 查找所有大小超过 100MB 的文件。
- 将这些文件的路径记录到 /var/log/large_files.list。
- 将这些文件移动到 /archive/large_files 目录。

3.3 编辑文件 vi、vim

vi是visual interface的简称, 是Linux中最经典的文本编辑器（Windows中的记事本）

vi的核心设计思想：让程序员的手指始终保持在键盘的核心区域, 就能完成所有编辑操作

vi的特点：

- 只能是编辑文本内容, 不能对字体段落进行排版
- 不支持鼠标操作
- 没有菜单
- 只有命令

vim 是从vi发展出来的文本编辑器, 支持代码补全、编译及显示效果等方面编程的功能提别丰富, 在程序员中被广泛使用, 被称为编辑器之神。

在CentOS Stream 9系统中，我们可以通过dnf命令安装vim编辑器：

```
dnf -y install vim
```

3.3.1 打开文件

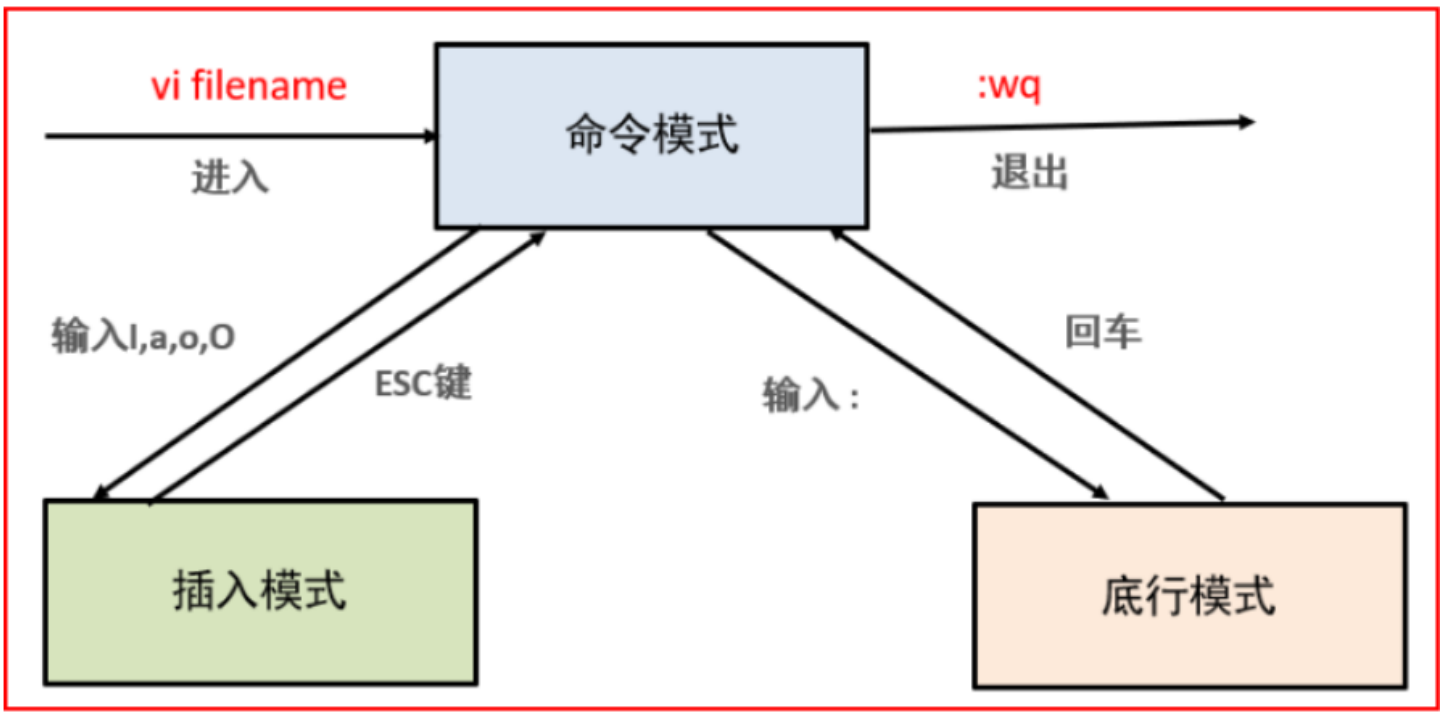
```
vi a.txt          # 直接打开文件
vim a.txt         # vim是vi的增强版
vim +10 a.txt     # 直接打开文件，并定位到第10行
```

vi/vim三种工作模式： 命令模式、插入模式(编辑模式)、末行模式(底行模式)

命令模式：复制、粘贴、移动光标、撤销与恢复

插入模式（编辑模式）：只能编辑文件内容（写字）

底行模式（末行模式）：保存文件、退出文件、显示行号、搜索关键词



3.3.2 命令模式

当我们通过vi/vim命令打开文件时，默认就处于==命令模式==

> 小技巧：进入vim编辑器，先查看左下角有没有提示信息，如果没有任何信息，代表你当前位于命令模式！

命令	功能

o	在当前行后面插入一空行
O	在当前行前面插入一空行
dd	剪切或删除光标所在行
ndd	从光标位置向下连续剪切或删除 n 行
yy	复制光标所在行
nyy	从光标位置向下连续复制n行
p	粘贴
u	撤销上一次命令，相当于Windows中的Ctrl + Z
gg	回到文件顶部
G	回到文件末尾
Ctrl + R	恢复，与u相对应

3.3.3 编辑模式

如何进入编辑模式呢?

答:

- `i`：在当前光标的前面插入内容
- `a`：在当前光标的后面插入内容
- `o`：在光标的后一行插入内容
- `O`：在光标的前一行插入内容

问题：如何从编辑模式回到命令模式

答：按`Esc`键

3.3.4 底行模式

核心重点记住3个字母即可，:w、:q、！ :x

在Linux操作系统中，文件必须先保存后退出!

！叹号代表强制，强制保存、强制退出、强制保存并退出!

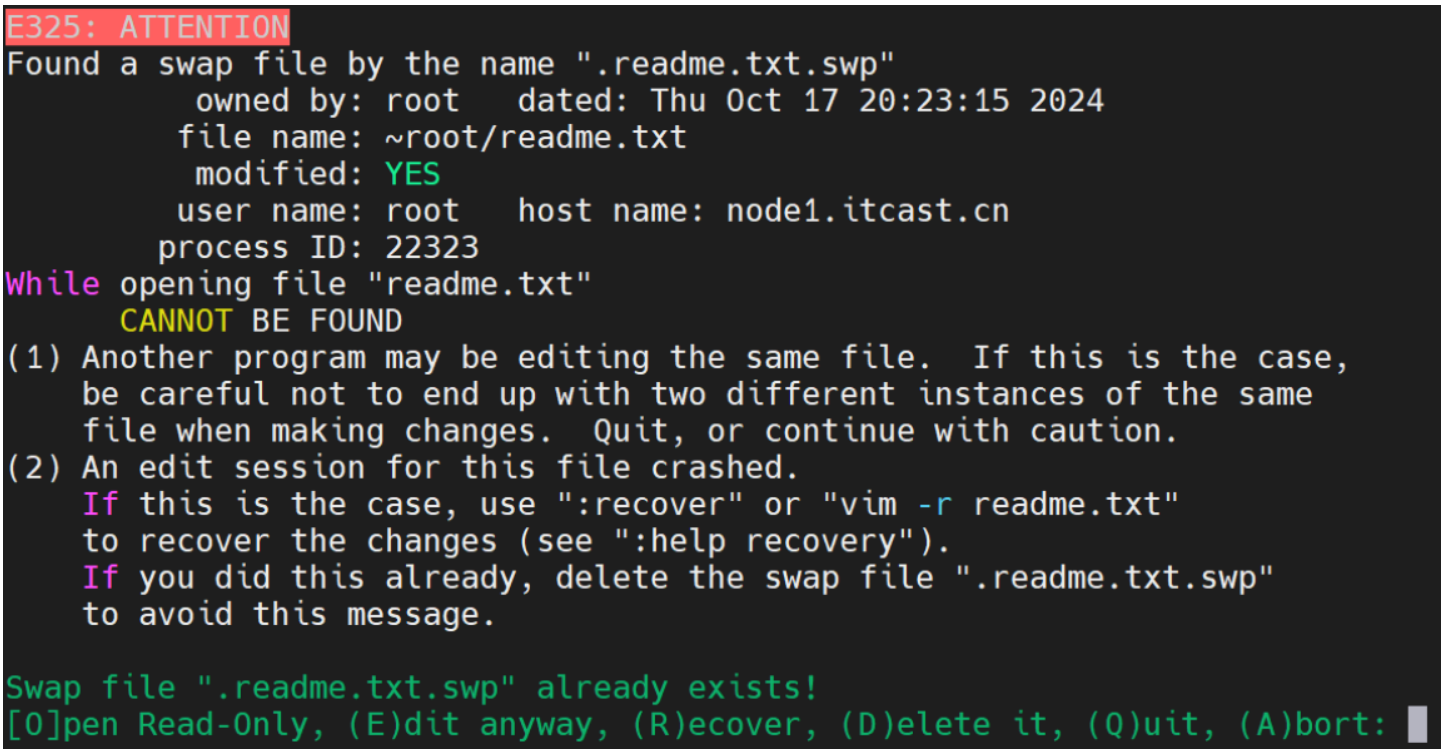
命令	**功能**
:w 文件	另存为

:w	保存(ctrl + s)
:q	退出, 如果没有保存,不允许退出
:q!	强行退出, 不保存退出
:wq	保存并退出
:x	保存并退出 【建议运维选择】
:set nu	设置行号，取消行号使用:set nonu
:%s/旧关键词/新关键词/g	文本替换
:noh	取消高亮
/关键词	搜索某一关键词

:wq和:x区别?

> 答：如果文件内容有改变，两者的效果是一样的。如果文件内容没有改变，:x不会改变文件的最后修改时间，但是:wq会更新文件的最后修改时间。
>
> **运维工程师,强烈推荐使用:x**

3.3.5 编辑常见问题



如果在打开某个文件时，弹出以上提示，那代表你这个文件之前没有保存就强制退出了，触发了Vim的备份机制，产生了一个`.文件名称.swp`交换文件。

以后每次打开之前的文件就会产生上面的提示，解决方案：

① 之前的修改不重要，可以直接删除的情况

- 直接回车，切换到错误的底部，然后按q，直接退出，然后执行删除操作

```
rm  .源文件名称.swp
```

② 之前的修改很重要，需要先恢复内容，然后再解决报错问题

- 第一步：直接回车，切换到错误的底部，找到回复菜单，一般是R，恢复文件内容
- 第二步：针对找回的内容进行:wq保存并退出
- 第三步：删除刚才产生的交换文件

```
rm  .源文件名称.swp
```

3.4 文本查看命令

- 1- cat命令【重要】
  - 作用：从上往下查看文件内容，一次性加载所有数据，适合小文件查看

```
格式:        cat 文件

案例说明:
需求:  查看家目录中anaconda-ks.cfg文件
cat anaconda-ks.cfg
```

- 2- more命令
  - 用于分页查看文件内容的命令

```
语法:
        more  filename

进入文件后，可使用的相关操作:

空格键：向下滚动一页。
Enter键：向下滚动一行。
b：向上翻页（可能不支持所有系统）。
q：退出查看

```

```
案例：
        more anaconda-ks.cfg
```

- 3- less命令【重要】
  - `less` 命令与 `more` 类似，但功能更强大，它允许用户**向前和向后滚动文件**，同时提供了更多的操作选项

```
语法：  less  filename

进入文件后，可使用的相关操作：

空格键：向下翻页。
b：向上翻页。
Page Up/Page Down：向上/向下滚动。  【通过方向键 上下来完成滚动】
q：退出查看。
/搜索词：搜索某个关键词，并跳转到该词的位置。  【可以检索操作】
n：跳到下一个匹配的搜索结果。
h：查看帮助信息

案例演示：
        less anaconda-ks.cfg
```

【退出操作】

```
强制退出：  ctrl + c

其他退出方式：q  quit exit
```

- 4- head命令【了解】
  - 作用：查看文件的前N行内容，默认显示前10行
  - 格式：head -N 文件

```
N：  表示从前往后看N行

案例说明：
需求：  查看家目录中anaconda-ks.cfg文件的前10行
head anaconda-ks.cfg

需求：  查看家目录中anaconda-ks.cfg文件的前20行
```

```
head -20 anaconda-ks.cfg
```

- 5- tail命令【重要】
  - 作用： 用于从后往前看文件，并支持监控文件变化
  - 语法：tail -[N][f] 文件

```
N:  表示从后往前看N行
f:  是否持续显示文件末尾内容

例如:
    tail -100 xx.log  查看某个文件最后100行的内容
    tail -f xx.log  实施查看该文件的输出内容
    tail -100f xx.log  默认显示文件末尾100行的内容，并实时监测最新的数据
```

- 6- wc命令
  - 作用: 用于统计文件或输入中的**行数**、**单词数**和**字节数**
  - 语法: wc [选项] [文件]

选项	说明
-l	统计行数（lines）
-w	统计单词数（words）
-c	统计字节数（bytes）
-m	统计字符数（characters）
-L	统计最长行的长度（length）

- 7- grep命令
  - 作用: 在文件中直接找到包含指定信息的那些行，并把这些信息显示出来
  - 语法: grep 要查找的内容 文件名

```
示例:
    grep network boot.log
    含义：在boot.log文件中，查找包含network的行

    grep network /var/log/*
```

```
6       含义：在/var/log目录下的所有文件中，查找包含network的行
```

3.5 文件查看案例练习

3.5.1 日志文件分析

- 需求1: 使用 cat 查看 /var/log/syslog 的内容。
- 需求2: 使用 head 查看该日志文件的前20行。
- 需求3: 使用 tail 查看该日志文件的最新50行。
- 需求4: 使用 less 浏览该日志文件，查找 "error" 关键字。
- 需求5: 使用 wc 统计 /var/log/syslog 文件的行数、单词数和字符数。

3.5.2 查找和统计文件中的特定内容

- 需求1: 在 /var/log 目录中找到一个大文件（如 dmesg 或 syslog）。
- 需求2: 使用 grep 命令查找所有包含 "timeout" 的行。
- 需求3: 使用 wc -l 统计包含 "timeout" 的行数。
- 需求4: 使用 less 浏览包含 "timeout" 的日志，确保能查看到整个内容。

3.6 文件解压缩

`tar`（tape archive）是一个用于打包和压缩文件的命令行工具。它可以将多个文件和目录打包成一个文件，也可以对这些文件进行压缩，减少存储空间。`tar` 命令广泛用于 Unix 和类 Unix 系统，包括 Linux 和 macOS。

- tar命令:
  - 格式: tar [选项] 文件 ...
  - 选项说明:

选项	说明
-c	创建一个新的归档文件; 这个选项表示将指定的文件或目录打包成一个新的归档文件。
-x	解压缩归档文件 ; 使用该选项从归档文件中提取文件。
-f	指定归档文件的名称;

	该选项必须紧跟着归档文件名，指定要创建或提取的归档文件。
-z	使用 gzip 压缩或解压归档文件; 该选项表示将归档文件压缩成 .tar.gz 格式，或者解压 .tar.gz 格式的归档文件。
-v	在创建或解压归档时显示详细过程信息（列出文件名）; 这个选项会显示每个文件在归档过程中的详细信息，通常用于调试或跟踪进度。
-j	使用 bzip2 压缩或解压归档文件; 该选项表示将归档文件压缩成 .tar.bz2 格式，或者解压 .tar.bz2 格式的归档文件。
-J	-J (xz) 使用 xz 压缩或解压归档文件; 该选项表示将归档文件压缩成 .tar.xz 格式，或者解压 .tar.xz 格式的归档文件。
-C	用于指向新的解压的目录

- 常用组合

参数组合	说明
-cvf	创建归档文件,俗称 "打包",打包后的文件后缀名为.tar
-xvf	解压归档文件, 俗称 "拆包"
-czvf	创建归档文件，并采用gzip进行**压缩**, 压缩后的文件后缀名为.tar.gz
-xzvf	**解压**被gzip压缩的归档文件
-cjvf	创建归档文件，并采用bzip2进行**压缩**, 压缩后的文件后缀名为.tar.bz2
-xjvf	**解压**被bzip2压缩的归档文件

```
案例:
    需求1:  请 /root目录下所有内容，全部压缩为一个文件，文件名字: root.tar.gz
    cd ~
    tar -czvf root.tar.gz a.txt data data.bat export hello.java

    需求2: 将 /root目录下 除了压缩包，其余的都删除后，尝试进行解压操作
    tar -xzvf root.tar.gz

    注意:  如果需要解压到指定的位置  后面跟上 -C 指定新的解压路径
    例如解压到/export:  tar -xzvf root.tar.gz -C /export
```

- zip和unzip命令
  - 作用: zip用于压缩, unzip进行解压
  - 格式:
    - zip [-r] 归档文件名.zip 需要归档的文件列表

选项	说明
-r	递归压缩（压缩文件夹）

    - unzip 归档文件名.zip [-d 解压目录]

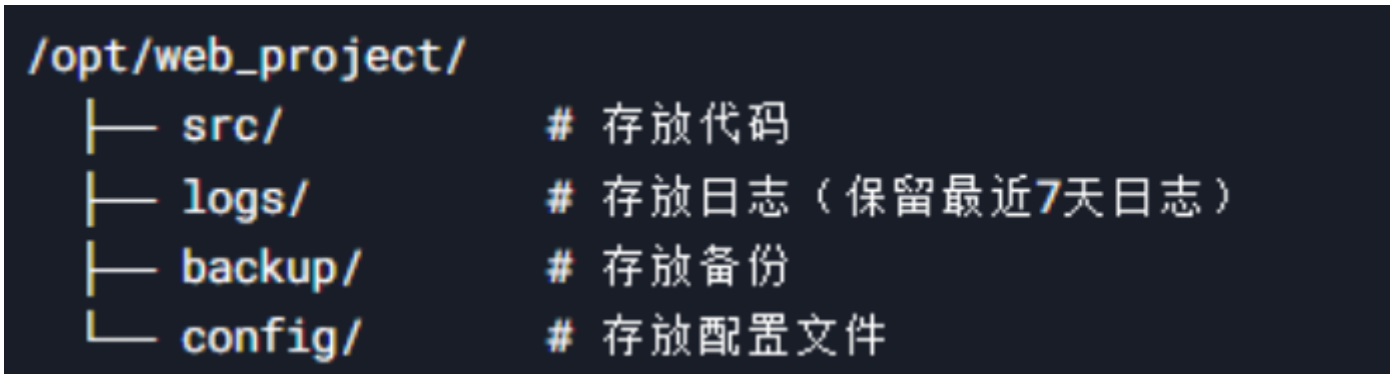
选项	说明
-d	如果不指定 表示解压到当前路径

```
示例代码:
zip 1.zip 1.txt
含义: 将1.txt 压缩成1.zip

zip 1dao4.zip 1.txt 2.txt 3.txt 4.txt
含义: 将1.txt,2.txt,3.txt,4.txt四个文件压缩成一个1dao4.zip文件


示例代码:
zip 1.zip
含义: 解压1.zip到当前目录下

zip 1dao4.zip -d /tmp/
含义: 将个1dao4.zip文件解压到/tmp目录下
```

3.7 综合练习

3.7.1 创建项目目录与日志管理

1- 创建目录结构:



2- 生成模拟日志

- 需求一: 在logs/目录下创建 3个日志文件，命名格式为 access_2023-08-0{1..3}.log
- 需求二: 在每个日志文件中写入 5 行文本，内容为 "[INFO] Request processed at 日期"

3- 备份与清理

- 需求一: 将logs/目录下所有以 2023-08-0 开头的日志文件，复制到backup/目录
- 需求二: 保留logs/目录中仅最近3天的日志（文件名日期最大的3个文件），其余删除

3.7.2 文件搜索与统计

使用find命令完成以下操作:

- 需求一: 在/etc目录下查找所有后缀为 .conf 的文件，将结果保存到 /opt/config_list.txt
- 需求二: 在用户主目录(~)下查找所有超过 1KB 的 .sh 脚本文件

统计信息:

- 需求一: 统计/opt/config_list.txt文件的总行数（即找到的.conf文件数量）
- 需求二: 查看/var/log/syslog的最后20行内容，并统计其中包含 "error" 关键词的行数