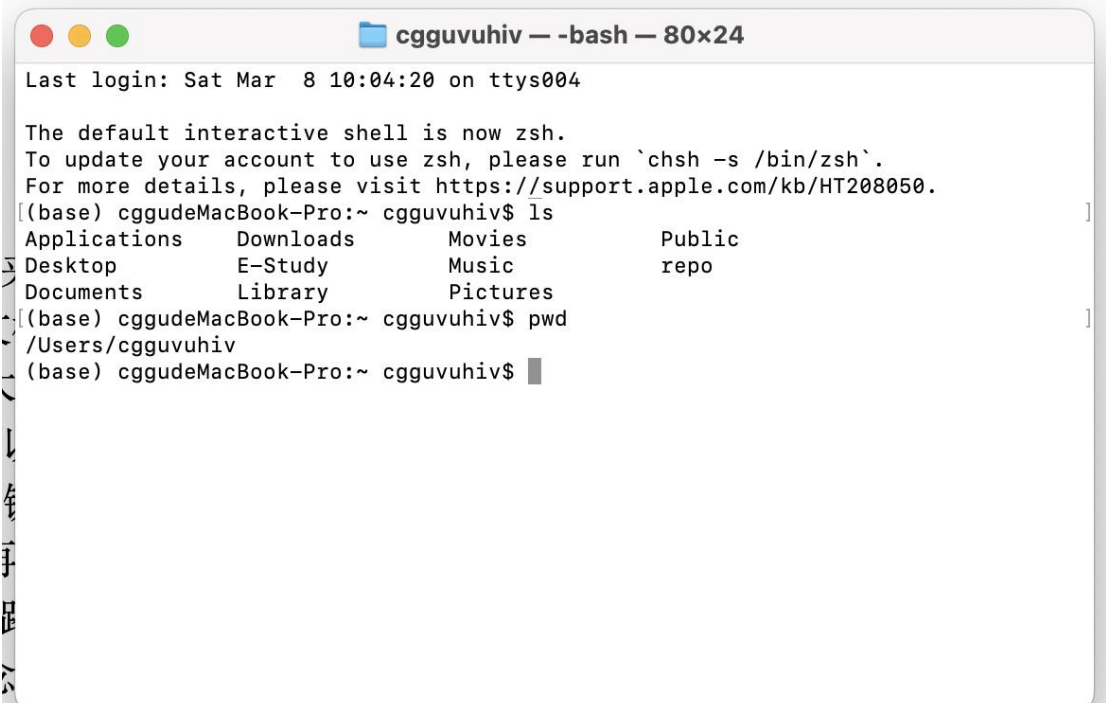


1.学习使用 `ls` 命令，检查文件夹的真实的文件系统路径，可以直接通过文件资源管理器的地址栏输入路径，快速定位到文档所在位置，无需在桌面或其他文件夹中逐个查找，尤其在桌面文件较多且杂乱时，能大大提高查找文件的效率。



```
cgguvuhiv — -bash — 80x24
Last login: Sat Mar  8 10:04:20 on ttys004

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
(base) cgguvuhiv$ ls
Applications  Downloads      Movies          Public
Desktop       E-Study        Music           repo
Documents     Library        Pictures
(base) cgguvuhiv$ pwd
/Users/cgguvuhiv
(base) cgguvuhiv$
```

2.学会了 `Tab` 键，“`Tab`” 键可以实现命令、文件名或路径的自动补全功能。当输入部分命令或文件名后，按下 “`Tab`” 键，如果系统能够匹配到唯一的选项，就会自动补全剩余的部分；如果存在多个匹配项，再次按下 “`Tab`” 键可以列出所有可能的选项供用户选择。3.`cat abc.txt` 可以看到文件中内容

4.`cd ..`回到上一级文件夹

5.理解 “根目录”，`cd /`可以到根目录

根目录是文件系统中顶层的目录，也被称为根文件夹，具有以下特点和作用：

- **层级结构的起点**：在文件系统的树形结构中，根目录是所有其他目录和文件的起始点，位于层级结构的最顶端。所有的文件和文件夹都直接或间接包含在根目录下，就像一棵树的树根，其他目录和文件是从根目录分支出来的枝干和叶子。例如，在 Windows 系统中，通常有多个盘符（如 C 盘、D 盘等），每个盘符都有自己的根目录，如 `C:\` 就是 C 盘的根目录；在 Linux 系统中，根目录用 `/` 表示。
- **系统文件存储**：根目录用于存放系统启动和运行所需的关键文件和文件夹，这些文件对于操作系统的正常运行至关重要。例如，在 Windows 系统的 C 盘根目录下，有 `boot.ini` 文件（用于系统启动配置）、`pagefile.sys`（虚拟内存文件）等系统文件；在 Linux 系统的根目录下，有 `/etc` 文件夹（存放系统配置文件）、`/bin` 文件夹（包含基本的系统命令）等重要的系统目录和文件。
- **文件路径的基准**：根目录是确定文件在文件系统中位置的基准。文件的完整路径是从根目录开始，沿着目录层级逐步到文件所在的具体位置。例如，在 Windows 系统中，文件 `C:\Program Files\Microsoft Office\Office16\WINWORD.EXE` 的路径就是以 C 盘根目录 `C:\` 为起点，依次经过 `Program Files`、`Microsoft Office`、`Office16` 等文件夹，最后到 `WINWORD.EXE` 文件；在 Linux 系统中，文件 `/home/user/Documents/file.txt` 的路径是以根目录 `/` 为起点，经过 `home`、`user`、`Documents` 等目录到达 `file.txt` 文件。

6. 理解绝对路径与相对路径（从根目录开始）

绝对路径

- **定义：**绝对路径是从文件系统的根目录开始，到目标文件或目录的完整路径，它明确地指定了文件或目录在整个文件系统中的唯一位置，不依赖于当前工作目录。
- **表示方法：**在不同的操作系统中，绝对路径的表示方式略有不同。
 - 在 Windows 系统中，绝对路径以盘符（如 C:、D: 等）开头，后面跟着反斜杠（\）和各级目录名称，例如 `C:\Program Files\Microsoft Office\Office16\WINWORD.EXE`。
 - 在 Linux 和 Unix 系统中，绝对路径以根目录（/）开头，后面是各级目录名称，用正斜杠（/）分隔，如 `/home/user/Documents/file.txt`。

相对路径

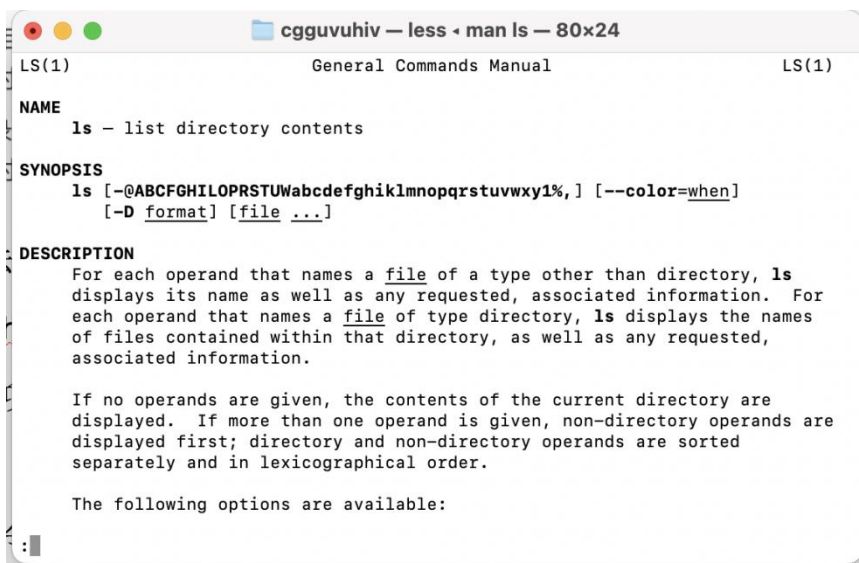
- **定义：**相对路径是相对于当前工作目录来表示文件或目录的位置。它不是从根目录开始，而是从当前所在的目录出发，描述目标文件或目录与当前目录的相对位置关系。
- **表示方法：**
 - `.` 表示当前目录。例如，当前目录是 `/home/user`，要访问当前目录下的 `Documents` 文件夹中的 `file.txt` 文件，可以使用相对路径 `./Documents/file.txt`。
 - `..` 表示当前目录的上一级目录。如果当前目录是 `/home/user/Documents`，要访问 `user` 目录下的 `file.txt` 文件，可以使用相对路径 `../file.txt`。

7. Unix 路径的标准写法：用/（斜杠，区别去反斜杠）

8. 理解 Shell (Bash、Zsh) 的基本语法结构:空格分隔（）、短选项、长选项、参数

(例: anaconda3)

9. 阅读 man ls



```
LS(1)                                General Commands Manual                                LS(1)

NAME
  ls - list directory contents

SYNOPSIS
  ls [-@ABCFGHILOPRSTUWabcdefghiklmnopqrstuvwxy1%,] [--color=when]
    [-D format] [file ...]

DESCRIPTION
  For each operand that names a file of a type other than directory, ls
  displays its name as well as any requested, associated information.  For
  each operand that names a file of type directory, ls displays the names
  of files contained within that directory, as well as any requested,
  associated information.

  If no operands are given, the contents of the current directory are
  displayed.  If more than one operand is given, non-directory operands are
  displayed first; directory and non-directory operands are sorted
  separately and in lexicographical order.

  The following options are available:
```

```
cgguvuhiv -- -bash -- 80x24
date: illegal time format
usage: date [-jnRu] [-I[date|hours|minutes|seconds]] [-f input_fmt]
        [-r filename|seconds] [-v[+|-]val[y|m|w|d|H|M|S]]
        [[[[mm]dd]HH]MM[[cc]yy][.SS] | new_date] [+output_fmt]
-bash: file://cgguvuhiv: No such file or directory
-bash: Saving: command not found

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
(base) cgguvuhiv$ ls -l
total 0
drwx-----  5 cgguvuhiv  staff  160  1 15  2024 Applications
drwx-----@ 32 cgguvuhiv  staff 1024  3 14 14:39 Desktop
drwx-----@ 17 cgguvuhiv  staff  544  9 16 16:03 Documents
drwx-----+ 73 cgguvuhiv  staff 2336  3 10 14:51 Downloads
drwxr-xr-x@  4 cgguvuhiv  staff  128  3 20  2024 E-Study
drwx-----@ 103 cgguvuhiv  staff 3296  4 10  2024 Library
drwx-----+  8 cgguvuhiv  staff  256  6 20  2023 Movies
drwx-----+  6 cgguvuhiv  staff  192 11 24  2020 Music
drwx-----+  9 cgguvuhiv  staff  288  3  8 10:21 Pictures
drwxr-xr-x+  4 cgguvuhiv  staff  128  1 28  2020 Public
drwxr-xr-x  4 cgguvuhiv  staff  128  3  8 10:32 repo
(base) cgguvuhiv$
```

ls -a 显示所有文件

ls -l 符号连接

10.zsh 与 bash

兼容性和性能

兼容性

- **Bash**: 是 Unix 和 Linux 系统中最常用的 Shell 之一，具有广泛的兼容性，大多数系统脚本和命令都能在 Bash 下很好地运行。
- **Zsh**: 与 Bash 有一定的兼容性，许多 Bash 命令和脚本可以在 Zsh 中直接运行，但在某些特殊情况下，可能会因为语法或特性的差异而导致一些脚本需要进行少量修改才能正常运行。

性能

- **Bash**: 启动速度相对较快，占用系统资源较少，适合在资源有限的环境中使用。
- **Zsh**: 由于功能强大，启动时需要加载更多的配置和插件，因此启动速度可能会稍慢一些，但在日常使用中，这种差异通常不太明显，特别是在交互使用时，Zsh 的高性能表现往往能弥补启动时的微小劣势。
- **Bash**: 补全功能相对基础，主要补全命令、文件名和目录名等。
- **Zsh**: 拥有更强大的补全功能，除了基本的补全外，还能根据历史命令和上下文进行智能补全，支持补全命令的参数、选项等，补全速度也更快。

历史命令

- **Bash**: 通过 `history` 命令查看历史命令，可使用 `!` (`n` 为历史命令编号) 来重复执行特定命令。
- **Zsh**: 不仅支持类似 Bash 的历史命令操作，还提供了更灵活的搜索和筛选功能，例如可以使用 `Ctrl + R` 进行反向搜索历史命令，能更方便地找到并重复使用之前的命令。

别名和函数

- **Bash**: 可以定义别名和函数，但在语法和功能上相对简单。
- **Zsh**: 在别名和函数的定义与使用上更加灵活强大，支持更多的参数扩展和高级特性，例如可以在别名中使用通配符等。

可定制性

主题和提示符

- **Bash**: 可以通过修改环境变量来定制提示符，但可定制的程度有限，外观相对较为简单。
- **Zsh**: 有大量的主题可供选择，能够轻松地定制出美观、个性化的命令行提示符，显示更多的信息，如当前目录、用户名、主机名、命令执行状态等。

插件系统

- **Bash**: 也有一些插件可供使用，但插件生态系统不如 Zsh 丰富。
- **Zsh**: 拥有丰富的插件系统，涵盖自动补全、语法高亮、版本控制集成等各种功能，如 `zsh-autosuggestions` 和 `zsh-syntax-highlighting` 等插件，能极大地扩展 Zsh 的功能。

11.cp 复制文件/文件夹，mv 命令移动 (重命名) 文件/文件夹

12.mkdir 创建文件；rm 删除文件

```
drwxr-xr-x+ 4 cgguvuhiv staff 128 1 28 2020 Public
drwxr-xr-x 4 cgguvuhiv staff 128 3 8 10:32 repo
(base) cggudeMacBook-Pro:~ cgguvuhiv$ mkdir myproject
(base) cggudeMacBook-Pro:~ cgguvuhiv$ rm myproject
rm: myproject: is a directory
(base) cggudeMacBook-Pro:~ cgguvuhiv$ df
Filesystem      512-blocks      Used Available Capacity iused      ifree %iused M
ounted on
/dev/disk1s5s1  489620264    25781312    67613456    28%    404475 338067280    0% /
devfs           381          381          0    100%     660      0    100% /
dev
/dev/disk1s2    489620264    7711696    67613456    11%     3170 338067280    0% /
System/Volumes/Preboot
/dev/disk1s4    489620264   10486344    67613456    14%        5 338067280    0% /
System/Volumes/VM
/dev/disk1s6    489620264    134288    67613456     1%      859 338067280    0% /
System/Volumes/Update
/dev/disk1s1    489620264   375193536    67613456    85%   2393577 338067280    1% /
System/Volumes/Data
map auto_home   0            0            0    100%      0      0      - /
System/Volumes/Data/home
/dev/disk1s5    489620264    25781312    67613456    28%   404507 338067280    0% /
System/Volumes/Update/mnt1
(base) cggudeMacBook-Pro:~ cgguvuhiv$
```

```
System/Volumes/Update
/dev/disk1s1    489620264   375193536    67613456    85%   2393577 338067280    1% /
System/Volumes/Data
map auto_home   0            0            0    100%      0      0      - /
System/Volumes/Data/home
/dev/disk1s5    489620264    25781312    67613456    28%   404507 338067280    0% /
System/Volumes/Update/mnt1
(base) cggudeMacBook-Pro:~ cgguvuhiv$ df -h
Filesystem      Size  Used Avail Capacity iused ifree %iused  Mounted on
/dev/disk1s5s1  233Gi  12Gi   32Gi    28%    404k  338M     0%      /
devfs          191Ki  191Ki   0Bi   100%     660      0    100%    /dev
/dev/disk1s2    233Gi   3.7Gi   32Gi    11%    3.2k  338M     0%    /System/Volu
mes/Preboot
/dev/disk1s4    233Gi   5.0Gi   32Gi    14%        5  338M     0%    /System/Volu
mes/VM
/dev/disk1s6    233Gi   66Mi   32Gi     1%      859  338M     0%    /System/Volu
mes/Update
/dev/disk1s1    233Gi  179Gi   32Gi    85%    2.4M  338M     1%    /System/Volu
mes/Data
map auto_home   0Bi    0Bi    0Bi   100%      0      0      - /System/Volu
mes/Data/home
/dev/disk1s5    233Gi  12Gi   32Gi    28%    405k  338M     0%    /System/Volu
mes/Update/mnt1
(base) cggudeMacBook-Pro:~ cgguvuhiv$
```

- 13. df 看磁盘空间; df -h
- 14. Control c 中断命令
- 15. du -d 1。看文件大小
- 16. du -s * | sort -nr > ~/report.txt

du -s *: du 是一个用于估算文件或目录磁盘使用空间的命令; -s 是 du 命令的一个选项, 代表 summarize, 意思是只显示每个参数的总计信息, 也就是只显示每个文件或目录占用磁盘空间的总量, 而不会列出子目录和文件的详细信息; * 是一个通配符, 代表当前目录下的所有文件和目录。因此, du -s * 这部分命令的作用是计算当前目录下每个文件和目录占用磁盘空间的总量, 并输出结果。

sort -nr: sort 是一个用于对文本行进行排序的命令; -n 是 sort 命令的一个选项, 表示按照数值大小进行排序, 而不是按照字典顺序排序; -r 也是 sort 命令的一个选项, 代表 reverse, 即反转排序结果, 实现降序排列; 这部分命令的作用是对 du -s * 输出的结果按照数值大小进行降序排序。

> ~/report.txt: > 是重定向符号, 它的作用是将命令的输出结果覆盖写入到指定的文件中。

如果文件不存在, 会创建该文件; 如果文件已经存在, 会清空文件原有内容并写入新的内容; ~/ 代表用户的主目录, 不同用户的主目录路径可能不同, 比如 root 用户的主目录是 /root, 普通用户 test 的主目录可能是 /home/test; report.txt 是指定的文件名; 这部分命令的作用是将 sort -nr 排序后的结果覆盖写入到用户主目录下的 report.txt 文件中。

整体命令作用: 这条命令的整体作用是计算当前目录下每个文件和目录占用磁盘空间的总量, 然后按照占用空间大小进行降序排序, 最后将排序结果保存到用户主目录下的 report.txt 文件中。通过查看这个文件, 你可以快速了解当前目录下哪些文件或目录占用的磁盘空间较大。

17. 建立私密仓库

```
repo — -bash — 80x24
(base) cggudeMacBook-Pro:~ cgguvuhiv$ cd mywork
-bash: cd: mywork: No such file or directory
(base) cggudeMacBook-Pro:~ cgguvuhiv$ cd repo
(base) cggudeMacBook-Pro:repo cgguvuhiv$ pwd
/Users/cgguvuhiv/repo
(base) cggudeMacBook-Pro:repo cgguvuhiv$ git clone git@gitcode.com:qqwwyy/mywork
.git
正克隆到 'mywork'...
警告: 您似乎克隆了一个空仓库。
提示: 使用 'master' 作为初始分支的名称。这个默认分支名称可能会更改。要在新仓库
中
提示: 配置使用初始分支名, 并消除这条警告, 请执行:
提示:
提示: git config --global init.defaultBranch <名称>
提示:
提示: 除了 'master' 之外, 通常选定的名字有 'main'、'trunk' 和 'development'。
提示: 可以通过以下命令重命名刚创建的分支:
提示:
提示: git branch -m <name>
(base) cggudeMacBook-Pro:repo cgguvuhiv$ ls -l
total 0
drwxr-xr-x  3 cgguvuhiv  staff   96  3  14 15:16 mywork
drwxr-xr-x  7 cgguvuhiv  staff  224  3  8 14:27 week01
(base) cggudeMacBook-Pro:repo cgguvuhiv$
```

项目列表 2

mywork 私有

我自己的代码和数据

☆ 0 🍴 0 • 14 分钟前

第01周打卡 Fork

学生 Fork 此仓库并通过 PR 提交第 1 周学习报告

☆ 0 🍴 0 • 6 天前