

第二周 命令行与文件系统（初级）

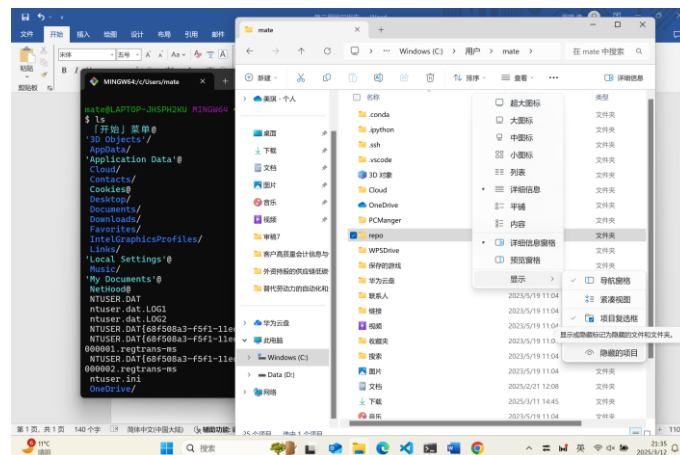
第二周学习基础知识,编程为了自动化,将专业知识转换为代码,让代码运行自动完成任务。

- 学习使用 ls 命令，检查自己计算机最常用的“桌面”、“下载”、“文档”等文件夹的真实的文件系统路径是什么

目录树组织文件系统资源

ls 显示当前工作目录下的文件列表

`pwd` 当前工作目录



```

Cookies@
Desktop/
Documents/
Downloads/
Favorites/
IntelGraphicsProfi
Links/
'Local Settings'@
Music/
'My Documents'@
NetHood@
NTUSER.DAT
ntuser.dat.LOG1
ntuser.dat.LOG2
NTUSER.DAT{68f508a
NTUSER.DAT{68f508a
000001.regtrans-ms
NTUSER.DAT{68f508a
000002.regtrans-ms
ntuser.ini
OneDrive/
PCManager/
Pictures/
PrintHood@
Recent@
repo/
'Saved Games' /
Searches/
SendTo@
Templates@

```

浅蓝色字是符号链接，蓝色字是文件夹

Ctrl+S 快捷键保存

Tab 键 命令行自动补全

提示符显示当前工作目录 ~用户主目录

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
```

cd 改变目录

```
mate@LAPTOP-JHSPH2KU MINGW64 ~
$ cd WPS
WPS Cloud Files/ WPSDrive/

mate@LAPTOP-JHSPH2KU MINGW64 ~
$ cd Desktop/

mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ pwd
/c/Users/mate/Desktop

mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ ls
'$二周学习报告.docx'
'~WRL0005.tmp'
abc.txt

mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ cat abc.txt
hello

mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ cat abc.txt
hello world

mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ cat abc.txt
hello world

mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ cat abc.txt
hello world

mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ cat abc.txt
goodbye

mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$
```

不保存在内存中不安全，保存在磁盘中安全。

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ cd ..
```

回到上一级文件夹。

pwd 展示当前工作目录

ls 列示具体内容

cd des(快捷键)打开文件

pwd 展示工作目录

ls 列示

cat abc.txt 读取

pwd 展示工作目录

cd ..返回上一级文件夹

ls 列示

cd dow(快捷键)打开文件

pwd 展示工作目录

ls 列示

cd ../doc(快捷键)打开上级文件夹的 doc 文件

ls 列示

pwd

cd /到根目录去（~用户主目录 /文件系统根目录）

路径分为绝对路径和相对路径

相对路径

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ cat abc.txt
hello world
goodbye
```

绝对路径

```
mate@LAPTOP-JHSPH2KU MINGW64 /
$ cat /c/Users/mate/Desktop/abc.txt
hello world
goodbye
mate@LAPTOP-JHSPH2KU MINGW64 /
$ pwd
/
```

绝对路径从根目录开始写

相对路径相对于当前工作目录而言，取决于当前工作目录

cd 回到用户主目录

/分隔符

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Downloads
$ cat Desktop/abc.txt
cat: Desktop/abc.txt: No such file or directory
```

Unix 路径-斜杠/分隔 /c/Users/mate 根目录是/

C:\Users\mate\Documents

Windows 路径-反斜杠\分隔

理解 shell 命令行-空格分隔、长选项、短选项、参数

```
mate@LAPTOP-JHSPH2KU MINGW64 ~
$ ls -a
./
../
.conda/
.emulator_console_auth_token
.gitconfig
.ipython/
.lessht
.ssh/
.vscode/
「开始」菜单@.隐藏文件
```

符号链接 cd Cookies -> /c/Users/mate/AppData/Local/Microsoft/Windows/INetCookies/

输入错误! ctrl+c 解决!!

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Cookies
$ cd 'my documents'
> pwd
>
> 
```

1. 使用 Ctrl+C:

- 按下 Ctrl 和 C 键可以中断当前正在执行的命令。如果命令还没有开始执行，它将清除命令行并允许你重新输入。

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Cookies
$ cd 'my documents'
> pwd
>
> ^C

mate@LAPTOP-JHSPH2KU MINGW64 ~/Cookies
$ 
```

文件类型-权限 rwx 可读可写可执行 owner group everyone -文件大小-修改时间-文件名

```
drwxr-xr-x 1 mate 197121      0  3月  8 14:18 ./
drwxr-xr-x 1 mate 197121      0  5月 19 2023 ../
drwxr-xr-x 1 mate 197121      0  3月  7 21:05 .conda/
-rw-r--r-- 1 mate 197121    16 10月 20 2022 .emulator_console_auth_token
-rw-r--r-- 1 mate 197121   144  3月  8 14:08 .gitconfig
drwxr-xr-x 1 mate 197121      0  3月  7 20:56 .ipython/
-rw-r--r-- 1 mate 197121    20  3月  8 14:18 .lessht
drwxr-xr-x 1 mate 197121      0  3月  7 21:36 .ssh/
drwxr-xr-x 1 mate 197121      0  3月  6 21:27 .vscode/
lrwxrwxrwx 1 mate 197121    58  5月 19 2023 「开始」菜单 -> '/c/Users/mate/AppD
drwxr-xr-x 1 mate 197121      0  9月 11 2021 '3D Objects'/
drwxr-xr-x 1 mate 197121      0  5月 19 2023 AppData/
lrwxrwxrwx 1 mate 197121    29  5月 19 2023 'Application Data' -> /c/Users/mate/
drwxr-xr-x 1 mate 197121      0 11月 26 2021 Cloud/
drwxr-xr-x 1 mate 197121      0  5月 19 2023 Contacts/
lrwxrwxrwx 1 mate 197121    57  5月 19 2023 Cookies -> /c/Users/mate/AppData/Lc
drwxr-xr-x 1 mate 197121      0  3月 14 13:30 Desktop/
drwxr-xr-x 1 mate 197121      0  3月 12 22:39 Documents/
```

```
-rwxr-xr-x 1 mate 197121 215027 2月 13 10:57 locate.exe*
-rwxr-xr-x 1 mate 197121 33855 2月 13 10:57 logname.exe*
-rwxr-xr-x 1 mate 197121 149479 2月 13 10:56 ls.exe*
-rwxr-xr-x 1 mate 197121 60928 2月 13 10:57 lsattr.exe*
-rwxr-xr-x 1 mate 197121 53862 2月 13 10:57 mac2unix.exe*
```

使用 ls 命令

2. 使用 cp 命令复制文件/文件夹, 使用 mv 命令移动 (重命名) 文件/文件夹

复制文件到文档里, 并更改名字

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ cat abc.txt
hello world
goodbye
mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ cp abc.txt ../Documents/

mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ cp abc.txt ../Documents/xyz.txt
```

复制文件夹到文档里

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ cp 审稿 ../Documents/
cp: -r not specified; omitting directory '审稿'

mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ cp -r 审稿 ../Documents/
```

移动文件、文件夹

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ mv ../Documents/xyz.txt ./

mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ mv ../Documents/shengao .
```

3. 使用 `mkdir` 命令创建一个名为 `myproject` 的新文件夹，然后从图形界面 (比如 Windows 的“文件资源管理器”、macOS 的“访达”) 复制粘贴某些文件/文件夹进去，用 `ls` 命令查看复制进去的文件/文件夹的大小和修改时间，最后用 `rm` 命令删除这些文件/文件夹

创建文件夹

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ mkdir myproject
```

查看具体信息

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ ls -alh myproject/
total 41K
drwxr-xr-x 1 mate 197121 0 3月 14 14:44 ./
drwxr-xr-x 1 mate 197121 0 3月 14 14:44 ../
drwxr-xr-x 1 mate 197121 0 3月 14 14:43 she
-rw-r--r-- 1 mate 197121 20 3月 14 14:26 xyz
drwxr-xr-x 1 mate 197121 0 3月 14 14:44 数字
```

删除文件、文件夹

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ rm xyz.txt

mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ rm -r myproject/

mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ rm store
rm: cannot remove 'store': No such file or directory

mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ rm -r shengao
```

强制删除

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ rm -rf
```

4. 使用 `df` 命令查看磁盘剩余空间，使用 `du` 命令查看文件/文件夹占用的磁盘空间

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ df
Filesystem      1K-blocks      Used Available Use% Mounted on
D:/Git          355705852    66502168 289203684 19% /
C:              125829116 122808688  3020428 98% /c

mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ df -h
Filesystem      Size  Used Avail Use% Mounted on
D:/Git          340G   64G  276G  19% /
C:              120G  118G   2.9G  98% /c
```

查看磁盘空间

```
C:              120G  118G   2.9G  98% /c

mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ ls -l /c
total 9128956
drwxr-xr-x 1 mate 197121 0 9月 27 2021 '$Re
drwxr-xr-x 1 mate 197121 0 2月 9 11:47 '$W
drwxr-xr-x 1 mate 197121 0 2月 19 10:33 '$W
drwxr-xr-x 1 mate 197121 0 8月 5 2023 add
drwxr-xr-x 1 mate 197121 0 3月 9 13:14 Co
lrwxrwxrwx 1 mate 197121 8 9月 11 2021 'Doc
-rw-r--r-- 1 mate 197121 12288 9月 30 2022 Dur
-rw-r--r-- 1 mate 197121 12288 2月 16 12:59 Dur
-rw-r--r-- 1 mate 197121 6781059072 3月 14 13:26 hi
```

查看 c 盘内容

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ ls -lh
total 8.7M
-rw-r--r-- 1 mate 197121 162 3月 14 13:30 '~$二屏
-rw-r--r-- 1 mate 197121 851K 3月 13 21:40 '~WRL0
-rw-r--r-- 1 mate 197121 20 3月 12 22:00 abc.t
-rwxr-xr-x 1 mate 197121 813 4月 21 2024 'Adobe
-rw-r--r-- 1 mate 197121 1.2M 3月 5 13:41 'ai an
-rw-r--r-- 1 mate 197121 282 5月 19 2023 desk
-rw-r--r-- 1 mate 197121 17K 3月 13 15:07 stata
-rwxr-xr-x 1 mate 197121 1.6K 2月 19 10:54 'Stata
-rw-r--r-- 1 mate 197121 1.3M 3月 14 14:59 第二屏
-rw-r--r-- 1 mate 197121 3.2M 3月 10 08:23 第一屏
```

查看文件占用空间

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ du .
```

查看当前文件夹里子文件夹占用空间

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ du -d 1 .
```

查看当前文件夹 深度主要 1 总的磁盘使用量

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ du -h -d 1 .
17M ./
4.6M ./
2.1G ./
119M ./
2.7G ./
0 ./
4.9G .

mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ du -h -d 0 .
4.9G .
```

查看桌面磁盘使用量

5. 使用某个 AI 大模型 (比如豆包, 或 DeepSeek) 解释下面的 Bash 命令,

```
du -s * | sort -nr > ~/report.txt
```

理解其含义后, 适当修改, 尝试在自己的计算机上应用。

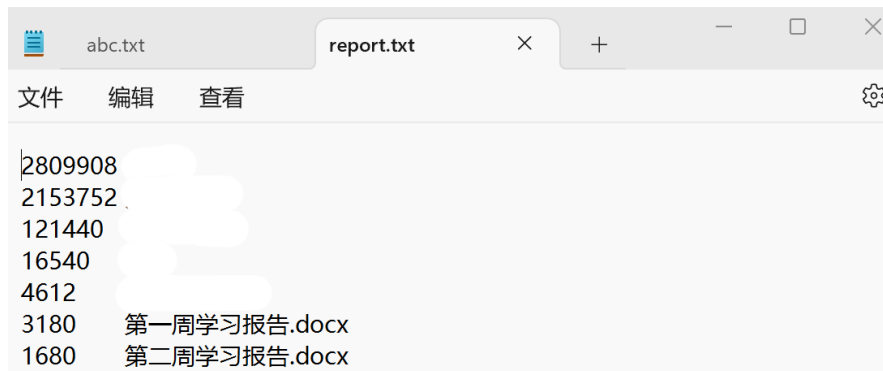
```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ du -h -d 0 .
4.9G .

mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ du -hs .
4.9G .
```

1. `du -s *` :
 - `du` 是一个用于估算文件空间占用的命令。
 - `-s` 参数表示对每个参数 (这里是 `*`, 代表当前目录下的所有文件和目录) 进行汇总, 只显示每个参数的总大小, 而不是显示其内部文件的大小。
 - `*` 是一个通配符, 表示当前目录下的所有文件和目录。
2. `| sort -nr` :
 - `|` 是管道符, 用于将前一个命令的输出作为后一个命令的输入。
 - `sort` 是一个用于对文本进行排序的命令。
 - `-n` 参数表示按照数值大小进行排序。
 - `-r` 参数表示反向排序 (从大到小)。
3. `> ~/report.txt` :
 - `>` 是重定向符号, 用于将命令的输出内容写入到指定的文件中。如果文件已存在, 则会覆盖原有内容。
 - `~/report.txt` 表示将结果保存到用户主目录下的 `report.txt` 文件中。

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ du -s * | sort -nr > ~/report.txt
```

运行命令，结果如下



尝试新命令

1. 查看文件夹大小并排序
2. 查找大文件
3. 统计文件数量
4. 列出最近修改的文件

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ du -sh * | sort -h > ~/daxiao.txt

mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ find . -type f -exec du -sh {} + | sort -h > ~/large_files.txt

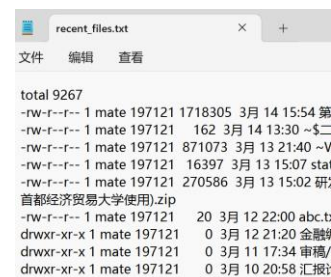
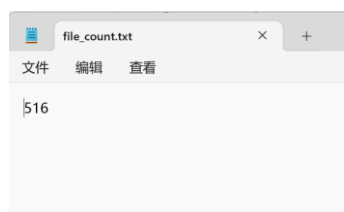
mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ find . -type f | wc -l > ~/file_count.txt

mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$ ls -lt | head -n 10 > ~/recent_files.txt


mate@LAPTOP-JHSPH2KU MINGW64 ~/Desktop
$
```

```
0 金融编
1.0K ~$二周
1.0K abc.txt
1.0K desкто
4.0K Adobe
4.0K StataM
20K stata.d
268K 研
812K 新
852K ~\
1.2M ai
1.2M 数
1.7M 第
3.2M 第
4.6M 数
17M 审稿
119M 研
2.1G 汇报论
2.7G 资料
```

```
1.0K ./~$
1.0K ./ab
1.0K ./de
4.0K ./Ac
4.0K ./Sta
4.0K ./汇
4.0K ./研
4.0K ./研
4.0K ./研
4.0K ./资
8.0K ./汇
8.0K ./汇
8.0K ./汇
8.0K ./汇
文件.do
8.0K ./汇
8.0K ./研
12K ./汇
12K ./研
12K ./资
12K ./资
12K ./资
12K ./资
```



6. 在 GitCode 平台新建一个你个人的私密的代码仓库 (非公开, 别人看不到), `clone` 到本地, 将一些你自己的工作文件 (文本文件或二进制文件都可以) 添加到仓库里, `push` 到平台上托管



The screenshot shows the GitCode web interface for a repository named 'readlijie/mywork'. The repository is private. The commit history table shows a single commit by 'readlijie' with the message 'added some files for test' at 13 minutes ago. The project introduction section is visible on the right.

文件	最后提交记录	最后更新时间
.gitconfig	added some files for test	13 分钟前

```
mate@LAPTOP-JHSPH2KU MINGW64 ~/repo/mywork (main)
$ git commit -m 'added some files for test'
[main (root-commit) 46ee29f] added some files for test
1 file changed, 6 insertions(+)
create mode 100644 .gitconfig

mate@LAPTOP-JHSPH2KU MINGW64 ~/repo/mywork (main)
$ cat ~/.gitconfig
[core]
    editor = "D:\\Microsoft VS Code\\bin\\code" --wait
    quotePath = false
[user]
    name = readlijie
    email = readlijie@noreply.gitcode.com

mate@LAPTOP-JHSPH2KU MINGW64 ~/repo
$ git clone git@gitcode.com:readlijie/mywork.git
Cloning into 'mywork'...
warning: You appear to have cloned an empty repository.

mate@LAPTOP-JHSPH2KU MINGW64 ~/repo
$ ls -l
total 4
drwxr-xr-x 1 mate 197121 0 3月 14 16:42 mywork/
-rw-r--r-- 1 mate 197121 0 3月 7 21:04 script1.py
drwxr-xr-x 1 mate 197121 0 3月 8 13:50 week01/

mate@LAPTOP-JHSPH2KU MINGW64 ~/repo/mywork (main)
$ ls -al
total 5
drwxr-xr-x 1 mate 197121 0 3月 14 16:45 ./
drwxr-xr-x 1 mate 197121 0 3月 14 16:42 ../
drwxr-xr-x 1 mate 197121 0 3月 14 16:49 .git/
-rw-r--r-- 1 mate 197121 144 3月 14 16:45 .gitconfig

mate@LAPTOP-JHSPH2KU MINGW64 ~/repo/mywork (main)
$ git push
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 346 bytes | 346.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Start Git Hooks Checking
To gitcode.com:readlijie/mywork.git
* [new branch]      main -> main

[PASSED]
```