小 tips:

1.命令行下的自动补全功能：Tab 键

1.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

pwd：当前工作目录

ls：工作目录下的所有文件

Cat：将文件路径作为参数传输过去，显示文件内容

Cd 空格..：回到上一级文件夹

```
Lenovo@LAPTOP-TH3BCRPU MINGW64 ~
$ ls
 「开始」菜单@              NTUSER.DAT
'3D Objects'/              ntuser.dat.LOG1
 ado/                      ntuser.dat.LOG2
 AppData/                  NTUSER.DAT{68569b3c-a0e0-1
'Application Data'@        NTUSER.DAT{68569b3c-a0e0-1
 Contacts/                 NTUSER.DAT{68569b3c-a0e0-1
 Cookies@                  ntuser.ini
 Desktop/                  OneDrive/
 Documents/                Pictures/
 Downloads/                PrintHood@
 Favorites/                Recent@
 Links/                    repo/
'Local Settings'@         'Saved Games'/
 MicrosoftEdgeBackups/     Searches/
 Music/                    SendTo@
'My Documents'@            Templates@
 NetHood@                  Videos/

Lenovo@LAPTOP-TH3BCRPU MINGW64 ~
$ pwd
/c/Users/Lenovo

Lenovo@LAPTOP-TH3BCRPU MINGW64 ~
$ ls
 「开始」菜单@              NTUSER.DAT
'3D Objects'/              ntuser.dat.LOG1
 ado/                      ntuser.dat.LOG2
```

```
Lenovo@LAPTOP-TH3BCRPU MINGW64 ~/Desktop
$ ls 气候政策不确定性/
环境政策不确定性对中国上市公司股价影响的实证研究_林伟源.caj
气候政策不确定性、能源市场与绿色债券市场泡沫传染_罗胜涛.caj
气候政策不确定性、数字化转型与企业投资效率_孙海波.caj
气候政策不确定性、投资者气候风险感知与系统性金融风险_胡丽宁.caj
气候政策不确定性对企业绿色创新的影响_张娜.caj
气候政策不确定性对全球能源...格和股票价格的动态影响研究_贺志芳.caj
气候政策不确定性下的企业E...于ESG文本语调的经验研究_汪顺.caj
气候政策不确定性与股价崩盘风险_夏欣雅.caj
气候政策不确定性与企业绿色...新闻媒体文本分析方法的测度_郭晶.caj
气候政策不确定性与企业绿色低碳创新_王沐丹.caj
气候政策不确定性与企业债券融资成本_翟鹏翔.caj
气候政策不确定性与银行系统性风险_王文蔚.caj
气候政策不确定性与中国企业升级困境_汪顺.caj
文献阅读.xlsx

Lenovo@LAPTOP-TH3BCRPU MINGW64 ~/Desktop
$ cat 气候政策不确定性/
cat: 气候政策不确定性/: Is a directory

Lenovo@LAPTOP-TH3BCRPU MINGW64 ~/Desktop
$ cd ..

Lenovo@LAPTOP-TH3BCRPU MINGW64 ~
$ ls
 「开始」菜单@            NTUSER.DAT
'3D Objects'/            ntuser.dat.LOG1
```

文件中显示文件扩展名才能找到

/：根目录

`/c/Users/qiang/Desktop` ：unix 路径

Windows 路径用:\

~：用户主目录

Cd 空格/：改变路径到根目录

绝对路径：从根目录开始一个完整的路径

相对路径：不是从根目录开始写的路径

ls 命令

ls - list directory contents   列举目录内容

【】内容表示可以有也可以没有

ls [*OPTION*]... [*FILE*]...   …表示可以有多个

List information about the FILEs (the current directory by
default).   Sort entr 如果没有给参数的话，默认按照字母顺序排序   or
--sort is specified.

Mandatory arguments to long options are mandatory for short
options too.

-a, --all 短选项 长选项 不会忽略以 . 开头的条目，显示所有文件
       do not ignore entries starting with .

```
qiang@gqm3win CLANGAR
$ ls -a
./
../
.conda/
.gitconfig
.git-for-windows-upd
.ssh/
.viminfo
.vscode/
anaconda3/
AppData/
'Application Data'@
Contacts/
Cookies@
Desktop/
Documents/
Downloads/
Favorites/
Links/
'Local Settings'@
Music/
'My Documents'@
NetHood@
NTUSER.DAT
ntuser.dat.LOG1
ntuser.dat.LOG2
```
蓝色的是文件夹，白色的是文件

-A, --almost-all
        do not list implied . and ..

--author
        with -l, print the author of each file

-b, --escape
        print C-style escapes for nongraphic characters

--block-size=*SIZE*
        with -l, scale sizes by SIZE when printing them; e.g.,
        '--block-size=M'; see SIZE format below

-B, --ignore-backups
        do not list implied entries ending with ˜

-c   with -lt: sort by, and show, ctime (time of last change of
        file status information); with -l: show ctime and sort by
        name; otherwise: sort by ctime, newest first

-C   list entries by columns

--color[=*WHEN*]
        color the output WHEN; more info below

-d, --directory
        list directories themselves, not their contents

-D, --dired
        generate output designed for Emacs' dired mode

```
-f         same as -a -U

-F, --classify[=WHEN]
           append indicator (one of */=>@|) to entries WHEN

--file-type
           likewise, except do not append '*'

--format=WORD
           across -x, commas -m, horizontal -x, long -l, single-column
           -1, verbose -l, vertical -C

--full-time
           like -l --time-style=full-iso

-g         like -l, but do not list owner

--group-directories-first
           group directories before files

-G, --no-group
           in a long listing, don't print group names

-h, --human-readable
           with -l and -s, print sizes like 1K 234M 2G etc.

--si       likewise, but use powers of 1000 not 1024

-H, --dereference-command-line
           follow symbolic links listed on the command line
```

-H, --dereference-command-line
    follow symbolic links listed on the command line

--dereference-command-line-symlink-to-dir
    follow each command line symbolic link that points to a
    directory

--hide=*PATTERN*
    do not list implied entries matching shell PATTERN
    (overridden by **-a** or **-A**)

--hyperlink[=*WHEN*]
    hyperlink file names WHEN

--indicator-style=*WORD*
    append indicator with style WORD to entry names: none
    (default), slash (**-p**), file-type (**--file-type**), classify
    (**-F**)

-i, --inode
    print the index number of each file

-I, --ignore=*PATTERN*
    do not list implied entries matching shell PATTERN

-k, --kibibytes
    default to 1024-byte blocks for file system usage; used
    only with **-s** and per directory totals

```
-1      use a long listing format

-L, --dereference
        when showing file information for a symbolic link, show
        information for the file the link references rather than
        for the link itself

-m      fill width with a comma separated list of entries

-n, --numeric-uid-gid
        like -l, but list numeric user and group IDs

-N, --literal
        print entry names without quoting

-o      like -l, but do not list group information

-p, --indicator-style=slash
        append / indicator to directories

-q, --hide-control-chars
        print ? instead of nongraphic characters

--show-control-chars
        show nongraphic characters as-is (the default, unless
        program is 'ls' and output is a terminal)
```

-Q, --quote-name

        enclose entry names in double quotes

--quoting-style=*WORD*

        use quoting style WORD for entry names: literal, locale,
        shell, shell-always, shell-escape, shell-escape-always, c,
        escape (overrides QUOTING_STYLE environment variable)

-r, --reverse

        reverse order while sorting

-R, --recursive

        list subdirectories recursively

-s, --size

        print the allocated size of each file, in blocks

-S   sort by file size, largest first大写，根据文件大小排序

--sort=*WORD*

        change default 'name' sort to WORD: none (-U), size (-S),
        time (-t), version (-v), extension (-X), name, width

--time=*WORD*

        select which timestamp used to display or sort; access time
        (**-u**): atime, access, use; metadata change time (**-c**): ctime,
        status; modified time (default): mtime, modification; birth
        time: birth, creation;

        with **-l**, WORD determines which time to show; with
        --sort=*time*, sort by WORD (newest first)

--time-style=_TIME_STYLE_
    time/date format with -l; see TIME_STYLE below

-t   sort by time, newest first; see **--time** 小写，根据时间排序

-T, --tabsize=_COLS_
    assume tab stops at each COLS instead of 8

-u   with -lt: sort by, and show, access time; with -l: show
    access time and sort by name; otherwise: sort by access
    time, newest first

-U   do not sort directory entries

-v   natural sort of (version) numbers within text

-w, --width=_COLS_
    set output width to COLS. 0 means no limit

-x   list entries by lines instead of by columns

-X   sort alphabetically by entry extension

-Z, --context
    print any security context of each file

--zero end each output line with NUL, not newline

-1   list one file per line

--help display this help and exit

--version
    output version information and exit

第一个字母是 d 代表文件夹，是-代表普通文件，l 代表符号链接（快捷方式）
后面紧跟着的是文件权限，r 是可读，w 是可写，x 是执行
数字代表占了多少个字节
年月日是文件修改时间

2.cp 复制文件

复制并改名，前半部分为来源，后半部分为
target

-r 递归，复制整个文件夹
mv 移动文件

将文件从 download 移动到桌面

3.mkdir 创建文件夹，然后用 ls 查看文件夹里面内容

用 rm 删掉文件夹内容

4.df 查看剩余磁盘空间

du 查看文件或文件夹占用的磁盘空间

5.

Clone 本地文件时要选 SSH，第一次选 HTTP 出现以下错误

选 SSH 之后正确

```
Lenovo@LAPTOP-TH3BCRPU MINGW64 ~/repo
$ git clone git@gitcode.com:Alicecindy/mywork.git
Cloning into 'mywork'...
warning: You appear to have cloned an empty repository.

Lenovo@LAPTOP-TH3BCRPU MINGW64 ~/repo
$ ls -
ls: cannot access '-': No such file or directory

Lenovo@LAPTOP-TH3BCRPU MINGW64 ~/repo
$ ls -
ls: cannot access '-': No such file or directory

Lenovo@LAPTOP-TH3BCRPU MINGW64 ~/repo
$ ls -l
total 4
drwxr-xr-x 1 Lenovo 197121 0  3月 16 23:20 mywork/
-rw-r--r-- 1 Lenovo 197121 0  3月 10 19:02 script1.py
drwxr-xr-x 1 Lenovo 197121 0  3月 10 21:04 week01/

Lenovo@LAPTOP-TH3BCRPU MINGW64 ~/repo
$ cd mywork

Lenovo@LAPTOP-TH3BCRPU MINGW64 ~/repo/mywork (main)
$
```

提交作业

1. Fork，右上角 fork

项目名称 * (当你 fork 一个项目时，系统默认会使用和 fork 原始项目一致的名称，不过你也可以使用自定义的名称)

第02周打卡_1264

项目路径 *(当你 fork 一个项目时，系统默认会使用和 fork 原始项目一致的路径，不过你也可以使用自定义的路径)

Alicecindy ∨ / week02_1264

在此命名空间中具有相同名称或路径的项目已经存在，选择 fork 项目时我们将自动为你使用随机的项目路径和名称

项目描述

学生 Fork 此仓库并通过 PR 提交第 2 周学习报告

main 全部

只复制 main 分支，你可以在 fork 的项目中创建你自己的分支版本，并通过Pull Request将你写的代码贡献给 cueb-fintech/week0

取消 创建Fork项目

同步源项目

> C ⌂ ○ gitcode.com/Alicecindy/week02

签 📱 手机书签 ⬛ Lenovo Quick Fix... 🔵 创客贴 🟠 热门游戏

A Alicecindy / **第02周打卡** 🔍

k 源：首经贸金融科技实验室 / 第02周打卡 🔘 同步源项目 >

Clone

⌂ ○ gitcode.com/Alicecindy/week02

几书签 ⬛ Lenovo Quick Fix... 🔵 创客贴 🟠 热门游戏 🔵 AI助手 🟢 万能小in 🎵 抖音 🟥 爱淘宝

Alicecindy / **第02周打卡** 🔍 按下 / 快速开启搜索

超金融科技实验室 / 第02周打卡 🔘 同步源项目 >

Issues 🔀 Pull Requests 💬 讨论 👥 项目成员 1 📈 分析 > 🔳 免费领取云主机

🔀 1 🏷 0 🔍 搜索文件 + 🔳 ⬇ 下载zip ▶ Clone

进行配置

 # 请确保本地完成了 Git 的全局配置

```
git config --global user.name Alicecindy
git config --global user.email Alicecindy@noreply.gitcode.com
```
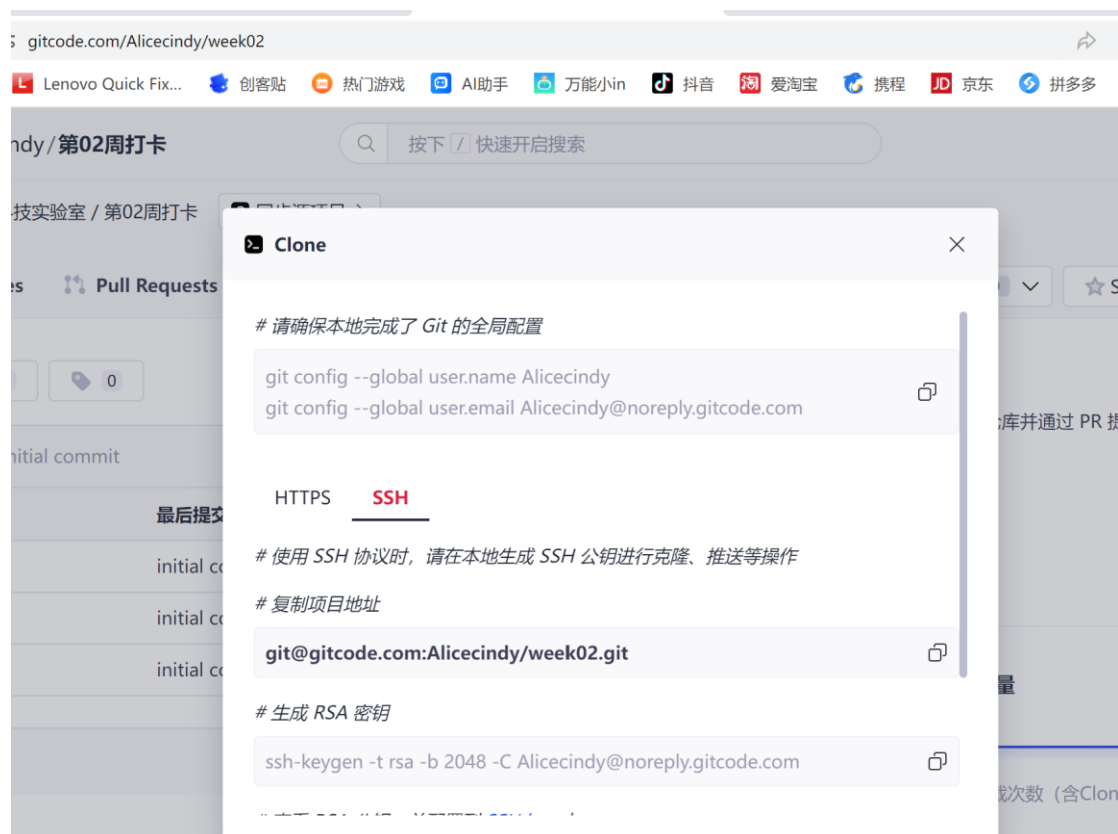
进行查看

```
Lenovo@LAPTOP-TH3BCRPU MINGW64 ~
$ git config --global user.name Alicecindy
git config --global user.email Alicecindy@noreply.gitcode.com

Lenovo@LAPTOP-TH3BCRPU MINGW64 ~
$ git config --list --global
core.editor="D:\Microsoft VS Code\bin\code" --wait
core.quotepath=false
user.name=Alicecindy
user.email=Alicecindy@noreply.gitcode.com
credential.https://gitcode.com.provider=generic
```

复制 SSH 公钥

```
Lenovo@LAPTOP-TH3BCRPU MINGW64 ~
$ cd repo

Lenovo@LAPTOP-TH3BCRPU MINGW64 ~/repo
$ pwd
/c/Users/Lenovo/repo

Lenovo@LAPTOP-TH3BCRPU MINGW64 ~/repo
$ git clone git@gitcode.com:Alicecindy/week02.git
Cloning into 'week02'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 5 (delta 0), reused 5 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (5/5), 8.45 KiB | 2.82 MiB/s, done.

Lenovo@LAPTOP-TH3BCRPU MINGW64 ~/repo
$ ll
total 4
drwxr-xr-x 1 Lenovo 197121 0  3月  16 23:52 mywork/
-rw-r--r-- 1 Lenovo 197121 0  3月  10 19:02 script1.py
drwxr-xr-x 1 Lenovo 197121 0  3月  10 21:04 week01/
drwxr-xr-x 1 Lenovo 197121 0  3月  17 19:53 week02/

Lenovo@LAPTOP-TH3BCRPU MINGW64 ~/repo
$ code week02/
```