

Week 2 学习报告、

Unix 路径

目录树：计算机以分层树形结构管理资源，便于组织和访问文件及目录。

ls：用于列出当前工作目录下的文件和文件夹，帮助用户查看目录内容。

pwd：执行后显示当前工作目录的完整路径，让用户明确所在位置。

~：代表用户主目录（如 /home/用户名），后续可接具体目录路径，表示在主目录下的位置。

cd：用于切换目录。cd.. 表示返回上一级目录（两个点）；cd. 表示当前目录（一个点，实际使用较少），通过 cd 后接目录路径可切换到指定目录。

```
ntuser-OP1705198167562.dat.LOG2
ntuser-OP1705198167562.dat{57248c10-3854-11ee-a459-04e8b9be937f}.TM.blf
ntuser-OP1705198167562.dat{57248c10-3854-11ee-a459-04e8b9be937f}.TMContainer000000000000000001.regtrans-ms
ntuser-OP1705198167562.dat{57248c10-3854-11ee-a459-04e8b9be937f}.TMContainer000000000000000002.regtrans-ms
ntuser-OP1705643262136.dat.LOG1
ntuser-OP1705643262136.dat.LOG2
ntuser-OP1705643262136.dat{57248c10-3854-11ee-a459-04e8b9be937f}.TM.blf
ntuser-OP1705643262136.dat{57248c10-3854-11ee-a459-04e8b9be937f}.TMContainer000000000000000001.regtrans-ms
ntuser-OP1705643262136.dat{57248c10-3854-11ee-a459-04e8b9be937f}.TMContainer000000000000000002.regtrans-ms
ocrcookie.json
OneDrive
Pictures
PrintHood
Recent
repo
'Saved Games'
screen_1695915247810.png
Searches
SendTo
Templates
Videos
week01
Zomboid

wyhl1@hikari MINGW64 ~
$ pwd
/c/Users/wyhl1

wyhl1@hikari MINGW64 ~
```

```
MB Windows PowerShell x MINGW64/c/Users/wyhl1/rep x + v
wyhl1@hikari MINGW64 ~
$ cd App
bash: cd: App: No such file or directory

wyhl1@hikari MINGW64 ~
$ cd Appdata
wyhl1@hikari MINGW64 ~/Appdata
$ cd
wyhl1@hikari MINGW64 ~
$ cd repo
wyhl1@hikari MINGW64 ~/repo
$ ls
txt1.txt week01/ week02/

wyhl1@hikari MINGW64 ~/repo
$ txt1.txt
bash: txt1.txt: command not found

wyhl1@hikari MINGW64 ~/repo
$ cat txt1.txt
hello world

wyhl1@hikari MINGW64 ~/repo
$ cat txt1.txt
hello world
goodbye

wyhl1@hikari MINGW64 ~/repo
$ |
```

绝对路径：以 “/” 开头，从根目录开始描述文件或目录的完整位置。例如 /home/user/documents/file.txt，明确指出从根目录依次经过 home、user、documents 目录，最终定位到 file.txt。

相对路径：相对于当前工作目录的路径，不包含根目录。例如当前目录是 /home/user，若要访问其中的 documents 目录，直接使用 documents 即可，无需写完整路径。

特殊符号：“.”：表示当前目录（如 ls 列出当前目录内容）。“..”：表示上一级目录（如 cd .. 切换到父目录）。连续多个斜杠（如 //）会被视为单个斜杠 “/”，用于简化路径书写。

ls -a：显示所有文件，包括以 “.” 开头的隐藏文件（a 代表 all）。

ls -l：以长格式列出文件详细信息，如权限、所有者、文件大小、修改时间等（l 代表 long format）。

ls -s：按文件大小排序显示（s 代表 size）。

ls -t：按文件修改时间排序显示（t 代表 time）。

ls -h：以人类易读的格式显示文件大小（如 KB、MB 等，h 代表 human - readable）。

```
$ cd
wyhl1@hikari MINGW64 ~
$ ls -a
./
../
.android/
.conda/
.gitconfig
.idlerc/
.ipython/
.leigod/
.lessgst
.local/
.matplotlib/
.ollama/
.spss/
.ssh/
.vscode/
「开始」菜单@
anaconda3/
AppData/
'Application Data'@
Contacts/
Cookies@
Desktop/
Documents/
Downloads/
Favorites/
Links/
ntuser-OP1705198102206.dat{57248c10-3854-11ee-a459-04e8b9be937f}.TMContainer000000000000000001.regtrans-ms
ntuser-OP1705198102206.dat{57248c10-3854-11ee-a459-04e8b9be937f}.TMContainer000000000000000002.regtrans-ms
ntuser-OP1705198167562.dat.LOG1
ntuser-OP1705198167562.dat.LOG2
ntuser-OP1705198167562.dat{57248c10-3854-11ee-a459-04e8b9be937f}.TM.blf
ntuser-OP1705198167562.dat{57248c10-3854-11ee-a459-04e8b9be937f}.TMContainer000000000000000001.regtrans-ms
ntuser-OP1705198167562.dat{57248c10-3854-11ee-a459-04e8b9be937f}.TMContainer000000000000000002.regtrans-ms
ntuser-OP1705643262136.dat.LOG1
ntuser-OP1705643262136.dat.LOG2
ntuser-OP1705643262136.dat{57248c10-3854-11ee-a459-04e8b9be937f}.TM.blf
ntuser-OP1705643262136.dat{57248c10-3854-11ee-a459-04e8b9be937f}.TMContainer000000000000000001.regtrans-ms
ntuser-OP1705643262136.dat{57248c10-3854-11ee-a459-04e8b9be937f}.TMContainer000000000000000002.regtrans-ms
ocrcookie.json
OneDrive/
Pictures/
PrintHood@
Recent@
repo/
'Saved Games'/
screen_1695915247810.png
Searches/
SendTo@
Templates@
Videos/
week01/
Zomboid/
wyhl1@hikari MINGW64 ~
$
```

一些常用指令

cp 指令（复制文件或目录）

常用选项及用法：

-r：递归复制目录及其内容。

-i：覆盖已有文件前询问，防止误覆盖。

-v：显示复制过程。

mv 指令（移动文件 / 目录或重命名）

移动文件 / 目录：将文件或目录移至指定位置。

示例：mv /home/user/file.txt /home/user/docs/，把 file.txt 移动到 docs 目录。

重命名：若目标位置为新文件名，则实现重命名。

示例：mv old_file.txt new_file.txt，将 old_file.txt 重命名为 new_file.txt。

常用选项 -i：覆盖前询问。

示例：mv -i file.txt /target/location/，若目标位置有同名文件，会提示确认是否覆盖。

```
The version control method may be selected via the --backup option or through
the VERSION_CONTROL environment variable. Here are the values:

    none, off       never make backups (even if --backup is given)
    numbered, t     make numbered backups
    existing, nil   numbered if numbered backups exist, simple otherwise
    simple, never   always make simple backups

As a special case, cp makes a backup of SOURCE when the force and backup
options are given and SOURCE and DEST are the same name for an existing,
regular file.

GNU coreutils online help: <https://www.gnu.org/software/coreutils/>
Report any translation bugs to <https://translationproject.org/team/>
Full documentation <https://www.gnu.org/software/coreutils/cp>
or available locally via: info '(coreutils) cp invocation'

wyh11@hikari MINGW64 ~/repo
$ cp -r \Users\wyh11\repo\1 C:\Users\wyh11\repo\2
cp: cannot stat 'Userswyh11repo1': No such file or directory

wyh11@hikari MINGW64 ~/repo
$ cp -r \Users\wyh11\repo\1 \Users\wyh11\repo\2
cp: cannot stat 'Userswyh11repo1': No such file or directory

wyh11@hikari MINGW64 ~/repo
$ cp -r "\Users\wyh11\repo\1" "\Users\wyh11\repo\2"

wyh11@hikari MINGW64 ~/repo
$ !
```

mkdir：用于创建新文件夹。

示例：mkdir my_folder，执行后创建名为 my_folder 的文件夹。

rm：删除文件或文件夹。

rm 文件名：删除单个文件。如 rm document.txt，删除 document.txt 文件。

rm -r 文件夹名：递归删除文件夹及其内部所有文件、子文件夹。如 rm -r my_dir，删除 my_dir 文件夹。

rm -rf 目标：强制删除（f 表示强制，r 表示递归），无需确认。如 rm -rf /path/to/target，直接删除目标，不提示。

注意事项：Unix/Linux 系统中，通过 rm 命令删除的文件或文件夹无法从回收站找回（不同于图形化界面的删除），操作前务必确认，避免误删重要数据。

新建私密代码仓

新建项目

需要导入已有项目?

使用模板

你可以选择一个模板项目来快速新建项目

仓库类型

☒ 代码仓 ☐ 模型仓 ☐ 数据集仓 ☐ Space

项目名称

mywork

项目路径

qingtianwawa / mywork

项目描述

请填写项目描述

0 / 2000

☐ 添加初始化 README 文件 (包括建议的项目的描述、安装指南、使用示例、贡献指南等信息)

☐ 添加 .gitignore 文件

选择 .gitignore 模板

☐ 添加 LICENSE 文件

选择 LICENSE 模板

☐ 公开项目

☒ 私密项目

取消

创建项目

操作解析：

克隆远程仓库到本地：

使用 `git clone [SSH 地址]`，通过 SSH 链接将远程仓库克隆到本地，创建本地仓库副本。

添加工作文件到仓库：

用 `cp` 源文件路径 本地仓库文件夹路径，将自己的工作文件（文本或二进制）复制到本地仓库目录。

查看提交日志：

`git log` 用于显示仓库的提交历史记录，包括提交者、时间、提交信息等。

查看工作目录状态：

`git status` 显示当前工作目录和暂存区的状态，红色文件表示有改动但未提交。

添加改动到暂存区：

`git add.` 将当前目录下所有改动（新增、修改等）添加到暂存区（“.”表示当前目录）。

提交改动到本地仓库：

`git commit -m “提交信息”`（图中未写 `-m`，实际需添加），将暂存区内容提交到本地仓库历史，此时改动仅在本地，未推送到远程网站。