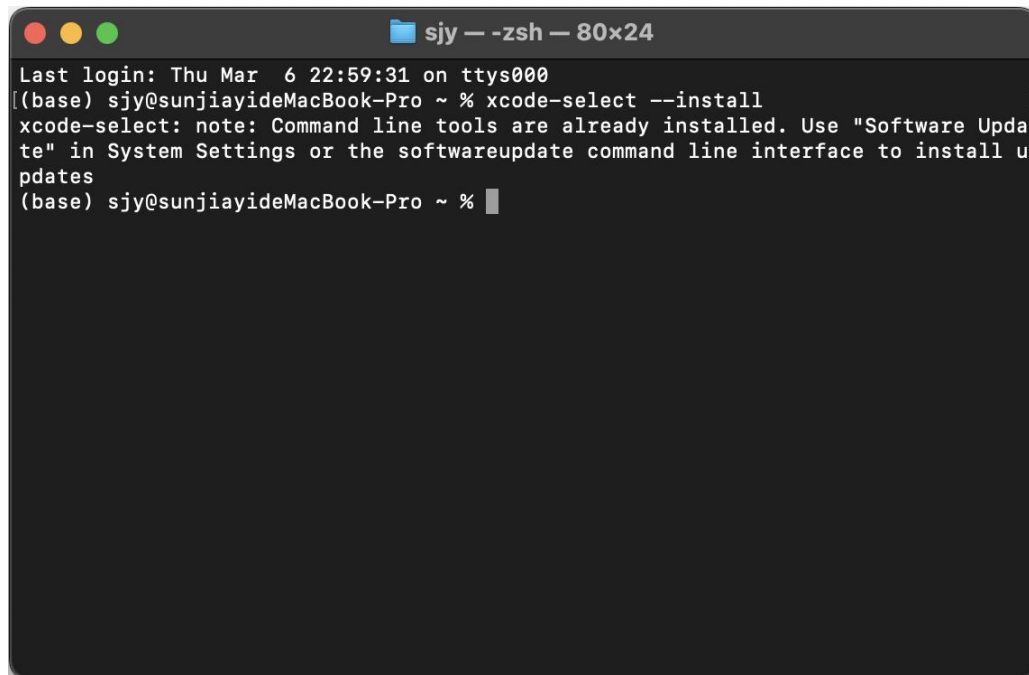


金融编程与计算

week 1——准备开发环境

访问网址：<https://gitcode.com/cueb-fintech/courses>

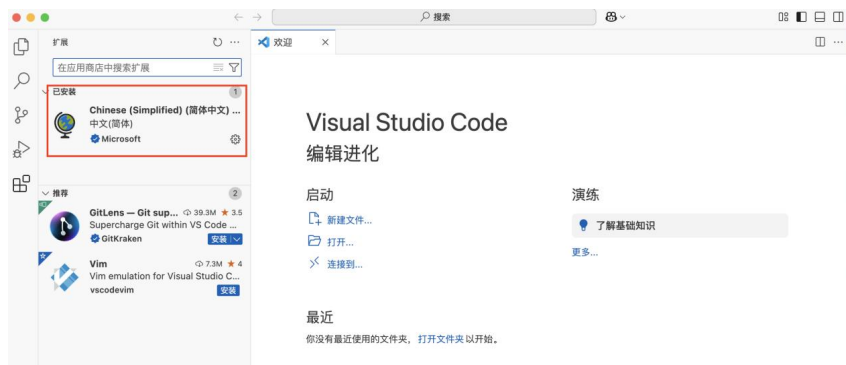
一、安装命令行工具：xcode-select --install



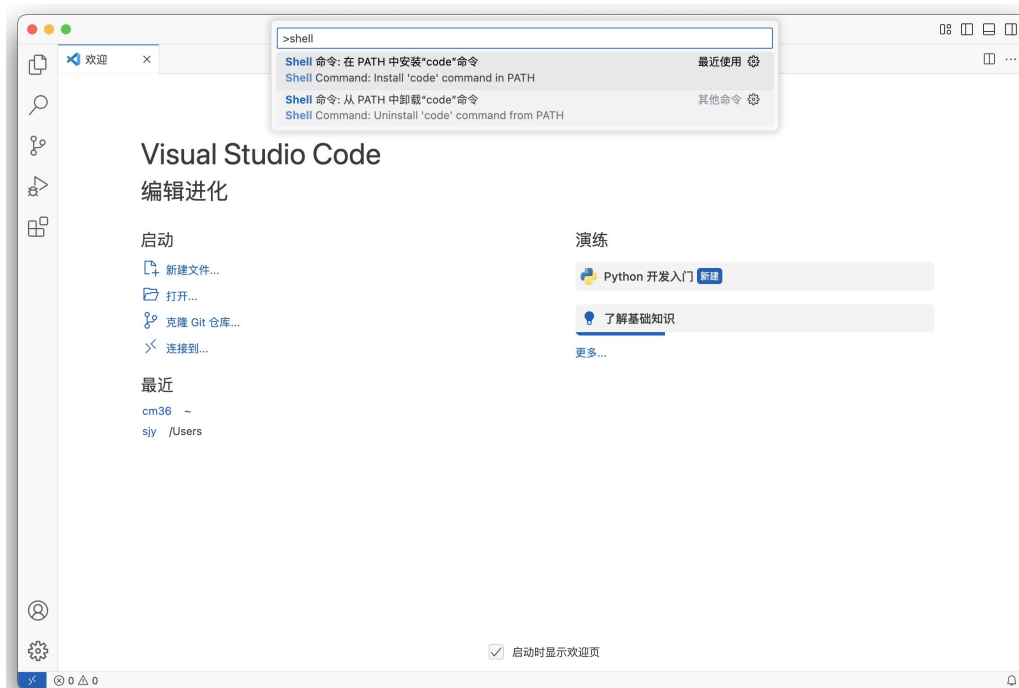
```
sjy — -zsh — 80x24
Last login: Thu Mar  6 22:59:31 on ttys000
(base) sjy@sunjiayideMacBook-Pro ~ % xcode-select --install
xcode-select: note: Command line tools are already installed. Use "Software Update" in System Settings or the softwareupdate command line interface to install updates
(base) sjy@sunjiayideMacBook-Pro ~ %
```

二、安装 VS Code

1. 安装：从官网中下载
2. 修改语言：从拓展坞中下载 Chinese，将语言转成中文



3. command+shift+p 打开命令面板，安装 code 命令



4. 试用——运行生成长度为 12 的随机代码

```
cm36 — zsh — 80x24

(base) sjy@sunjiayideMacBook-Pro cm36 % cat 11
(base) sjy@sunjiayideMacBook-Pro cm36 % cat 11.py
cat: 11.py: No such file or directory
(base) sjy@sunjiayideMacBook-Pro cm36 % ls
11
(base) sjy@sunjiayideMacBook-Pro cm36 % cat 11
import random
import string

def generate_password(length):
    # 定义所有可能的字符，包括字母（大写和小写）、数字和特殊字符
    all_characters = string.ascii_letters + string.digits + string.punctuation
    # 从所有可能的字符中随机选择指定长度的字符，组成密码
    password = ''.join(random.choice(all_characters) for i in range(length))
    return password

# 调用函数生成一个长度为 12 的随机密码
password = generate_password(12)
print("生成的随机密码是:", password)
(base) sjy@sunjiayideMacBook-Pro cm36 % python 11
生成的随机密码是: dfwqmQEL3R/2
(base) sjy@sunjiayideMacBook-Pro cm36 % python 11
生成的随机密码是: GdYR6o*>#s/O
(base) sjy@sunjiayideMacBook-Pro cm36 %
```

三、安装 Git

1、MAC 需先安装 Homebrew

(1) 在第一天安装 Homebrew 时，输错了密码。后直接重启终端，

再进行下载操作,在下载完成后,发现无法将桌面文件复制到硬盘中,找寻苹果技术人员在线排查后无果,最后将电脑恢复至出厂设置。(不知道什么原因导致的,希望可以得到解答)

****本次收获****

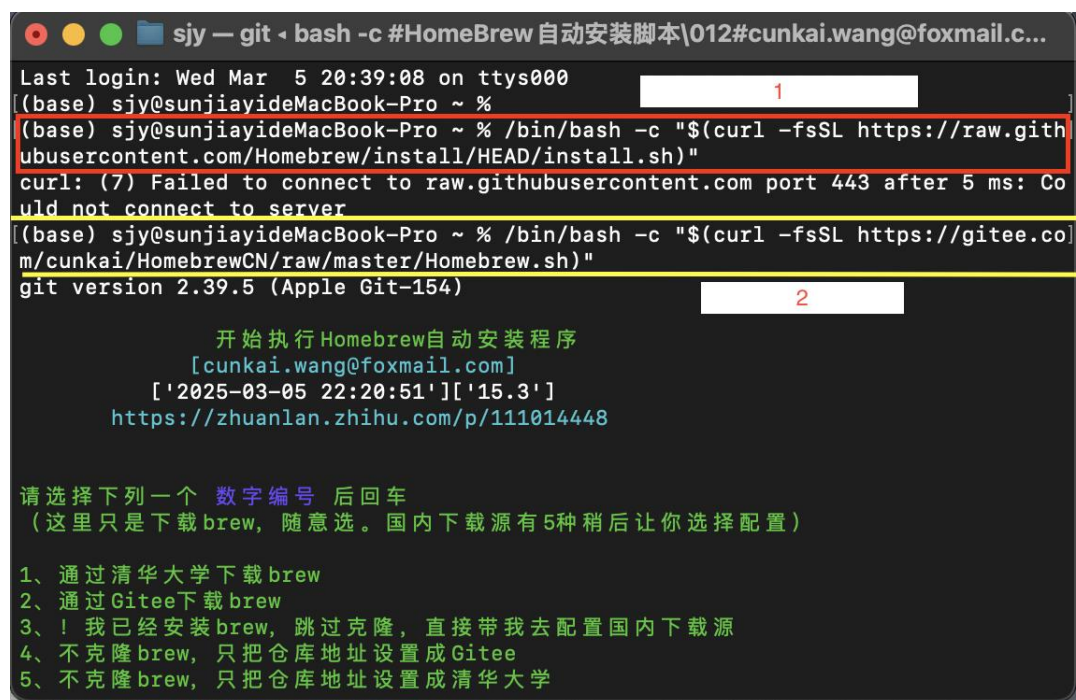
- a. 首先一定要随时并提前备份电脑文件,以防出现上述问题
- b. 在寻找解决办法时,发现不同电脑还有不同的硬盘匹配格式
(可以通过格式化硬盘进行修改)

ExFAT——可适用于两系统之间传输;

NTFS——微软文件系统;

APFS——苹果文件系统;

(2) 在第二次安装 Homebrew 时,显示连接不上,通过上网查询+老师文件中提示发现可以通过镜像站的方式下载。命令见下方



```
sjy — git - bash -c #HomeBrew 自动安装脚本\012#cunkai.wang@foxmail.c...
Last login: Wed Mar  5 20:39:08 on ttys000
(base) sjy@sunjiayideMacBook-Pro ~ %
(base) sjy@sunjiayideMacBook-Pro ~ % /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
curl: (7) Failed to connect to raw.githubusercontent.com port 443 after 5 ms: Could not connect to server
(base) sjy@sunjiayideMacBook-Pro ~ % /bin/bash -c "$(curl -fsSL https://gitee.com/cunkai/HomebrewCN/raw/master/Homebrew.sh)"
git version 2.39.5 (Apple Git-154)

开始执行 Homebrew 自动安装程序
[cunkai.wang@foxmail.com]
['2025-03-05 22:20:51']['15.3']
https://zhuanlan.zhihu.com/p/111014448

请选择下列一个 数字编号 后回车
(这里只是下载 brew, 随意选。国内下载源有5种稍后让你选择配置)

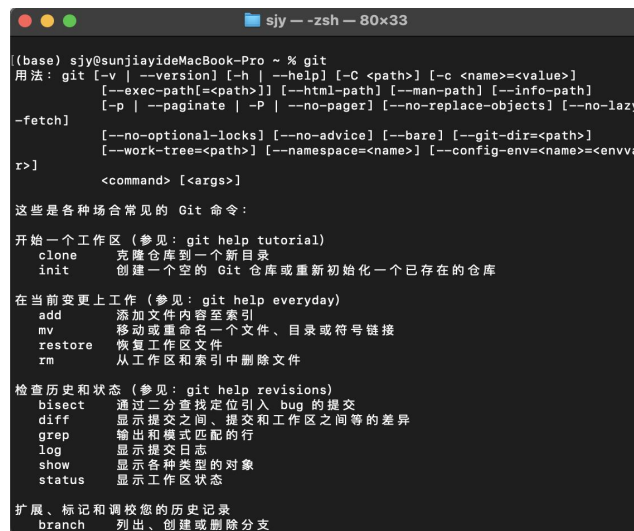
1、通过清华大学下载 brew
2、通过 Gitee 下载 brew
3、! 我已经安装 brew, 跳过克隆, 直接带我去配置国内下载源
4、不克隆 brew, 只把仓库地址设置成 Gitee
5、不克隆 brew, 只把仓库地址设置成清华大学
```

2、下载 Git

(1) 下载命令: `brew install git`

(2) 检验是否安装成功: `git --version` (`--version` 可用通用于检验其他软件的版本号)

(3) 配置 Git 的用户名和电子邮件: `git config --global user.name ""`



```
(base) sjy@sunjiayideMacBook-Pro ~ % git
用法: git [-v | --version] [-h | --help] [-C <path>] [-c <name>=<value>]
      [--exec-path=<path>] [--html-path] [--man-path] [--info-path]
      [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--no-lazy
~fetch]
      [--no-optional-locks] [--no-advice] [--bare] [--git-dir=<path>]
      [--work-tree=<path>] [--namespace=<name>] [--config-env=<name>=<envva
r>]
      <command> [<args>]

这些是各种场合常见的 Git 命令:

开始一个工作区 (参见: git help tutorial)
  clone 克隆仓库到一个新目录
  init  创建一个空的 Git 仓库或重新初始化一个已存在的仓库

在当前变更上工作 (参见: git help everyday)
  add  添加文件内容至索引
  mv   移动或重命名一个文件、目录或符号链接
  restore 恢复工作区文件
  rm   从工作区和索引中删除文件

检查历史和状态 (参见: git help revisions)
  bisect 通过二分查找定位引入 bug 的提交
  diff   显示提交之间、提交和工作区之间等的差异
  grep   输出和模式匹配的行
  log    显示提交日志
  show   显示各种类型的对象
  status 显示工作区状态

扩展、标记和调校您的历史记录
  branch 列出、创建或删除分支
```

`git config --global user.email ""`

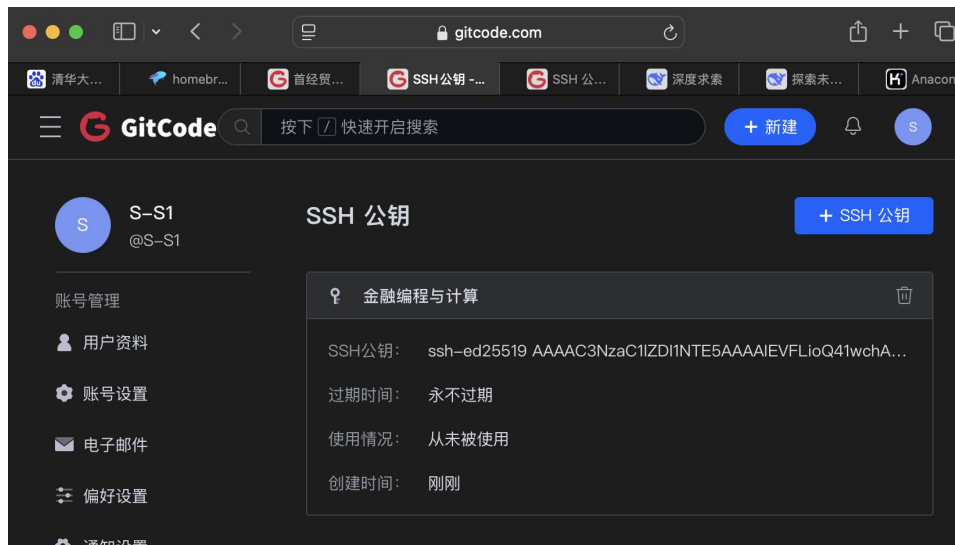
四、安装 Anaconda

1、Anaconda 和 python 区别

Python 本身是一个基础的编程工具,用户需要根据具体需求安装各种库来扩展其功能。Anaconda 是一个基于 Python 的数据科学平台,它包含了一系列用于数据处理、分析和可视化的工具和库。具体来说 Anaconda 包含 python, 是一个基于 Python 的集成环境。在安装 Anaconda 后,就可以直接使用 Python,而无需单独安装。

五、生成 SSH 密钥

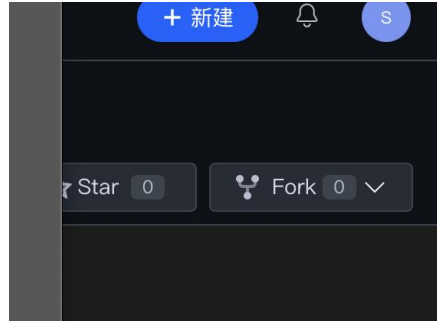
公钥可以发给别人，私钥需要自己保存好！！！！



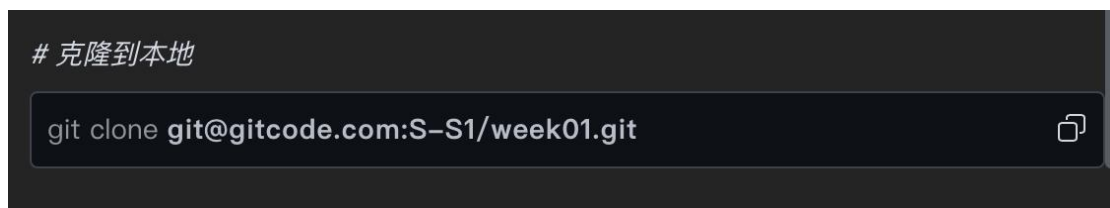
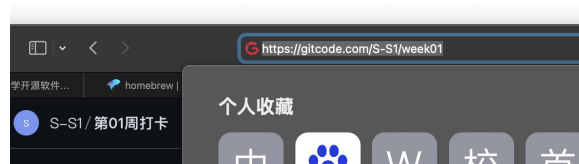
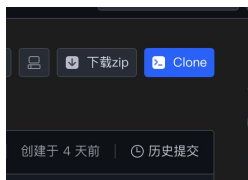
```
.ssh — -zsh — 80x24
[(base) sjy@sunjiayideMacBook-Pro cm36 % pwd
/Users/sjy/cm36
[(base) sjy@sunjiayideMacBook-Pro cm36 % quit
zsh: command not found: quit
[(base) sjy@sunjiayideMacBook-Pro cm36 % quit()
[function>
[function>
[(base) sjy@sunjiayideMacBook-Pro cm36 % cd
[(base) sjy@sunjiayideMacBook-Pro ~ % pwd
/Users/sjy
[(base) sjy@sunjiayideMacBook-Pro ~ % cd .ssh
[(base) sjy@sunjiayideMacBook-Pro .ssh % ls
id_ed25519      id_ed25519.pub
[(base) sjy@sunjiayideMacBook-Pro .ssh % cat id_ed25519.pub
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIEVFLioQ41wchA0/sorl/EOp7SD0CLh4SeHf0/xVduAN
1819202951@qq.com
[(base) sjy@sunjiayideMacBook-Pro .ssh % ssh -T git@gitcode.com
The authenticity of host 'gitcode.com (116.205.2.91)' can't be established.
RSA key fingerprint is SHA256:aTlsy+4ARMC7nWyy5eKIqUkotk8yv7Jd+XXoP4EXj1Y.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'gitcode.com' (RSA) to the list of known hosts.
remote: Welcome to GitCode, S-S1
[(base) sjy@sunjiayideMacBook-Pro .ssh %
```

六、学习报告提交

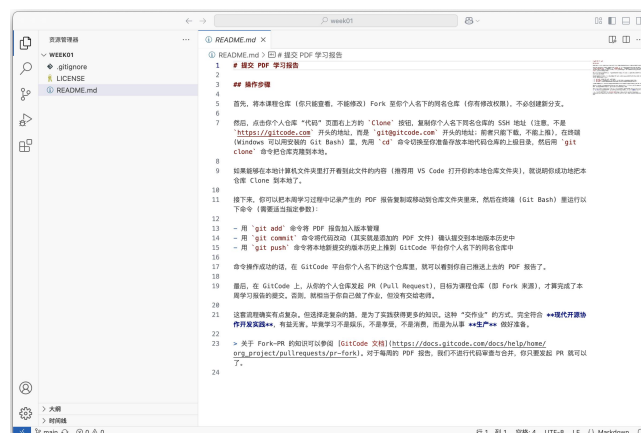
1、先 fork 一个分支到自己的账号下



2、再在自己的库内进行克隆，**一定要注意网址是自己的库**，并将代码输入至终端，放置在 cm36 文件夹中



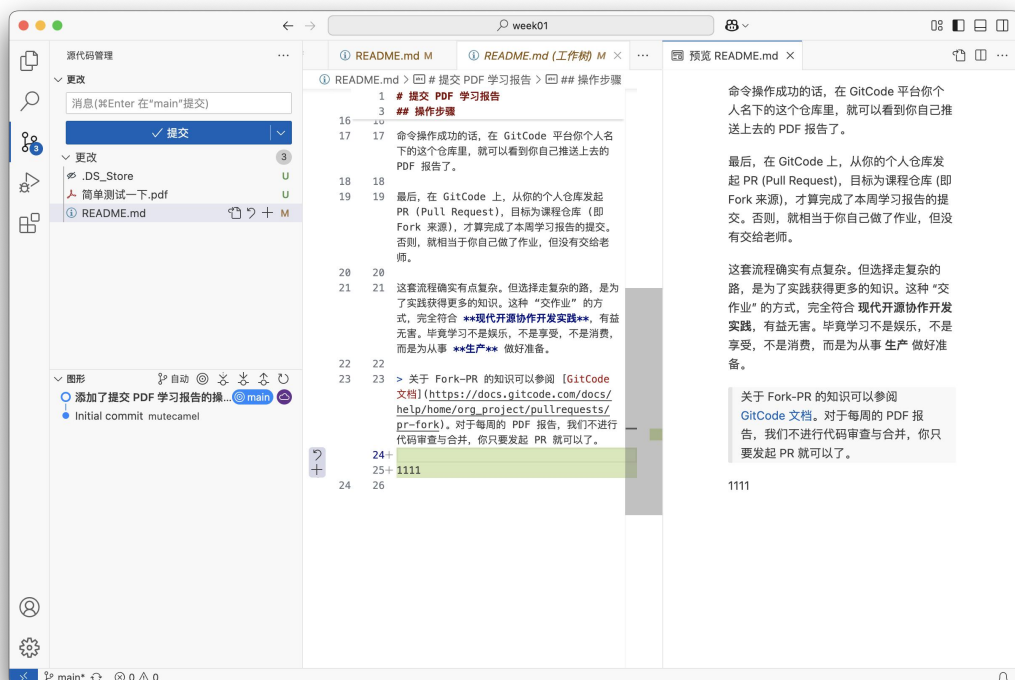
3、code week01/: 用 vs code 打开



4、将学习笔记文档存到 week01 中，即可用 vs code 打开



：可以查看修改记录，绿色的是有修改部分，红色是被删除的部分



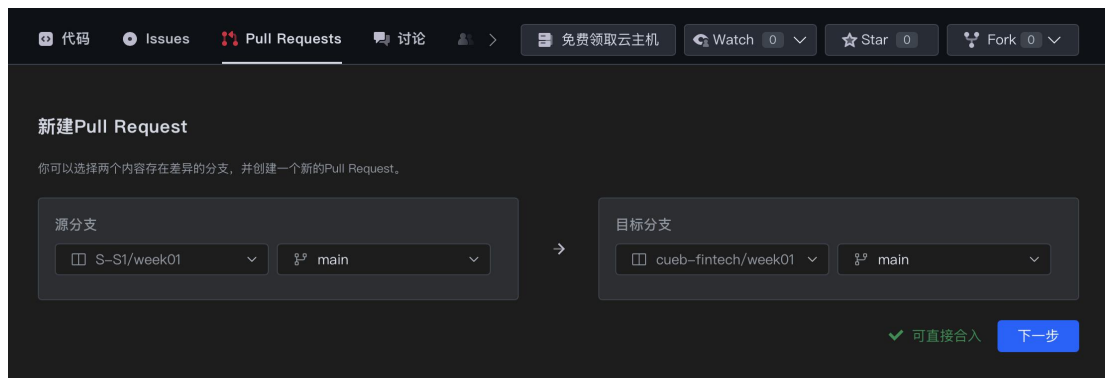
5、提交

- 用 `git add` 命令将 PDF 报告加入版本管理
- 用 `git commit` 命令将代码改动（其实就是添加的 PDF 文件）确认提交到本地版本历史中
- 用 `git push` 命令将本地新提交的版本历史上推到 GitCode 平台你个人名下的同名仓库中

`git commit -m "可以描述一下在干什么"`

`git push origin main`

6、最终发起 PR



一些终端小命令：

- 1、control+C=打断命令
- 2、pwd 显示根目录
- 3、cat 抓取内容
- 4、command+n 打开另一个界面
- 5、which+文件名 找到所在路径
- 6、mkdir+名字=新建目录
- 7、ls -l 查看当前路径下文档
- 8、git status 查看当前状态, config --global core.quotepath false 将文件改成中文名

```
(base) sjy@sunjiayideMacBook-Pro week01 % git status
位于分支 main
您的分支与上游分支 'origin/main' 一致。

尚未暂存以备提交的变更：
  (使用 "git add <文件>..." 更新要提交的内容)
  (使用 "git restore <文件>..." 丢弃工作区的改动)
    修改：      README.md

未跟踪的文件：
  (使用 "git add <文件>..." 以包含要提交的内容)
    .DS_Store
    "\347\256\200\345\215\225\346\265\213\350\257\225\344\270\200\344\270\213.pdf"

修改尚未加入提交 (使用 "git add" 和/或 "git commit -a")
(base) sjy@sunjiayideMacBook-Pro week01 %
```

```
week01 -- zsh -- 90x24

    修改：      README.md

未跟踪的文件：
  (使用 "git add <文件>..." 以包含要提交的内容)
    .DS_Store
    "\347\256\200\345\215\225\346\265\213\350\257\225\344\270\200\344\270\213.pdf"

[(base) sjy@sunjiayideMacBook-Pro week01 % git config --global core.quotepath false ]
[(base) sjy@sunjiayideMacBook-Pro week01 % git status ]
位于分支 main
您的分支与上游分支 'origin/main' 一致。

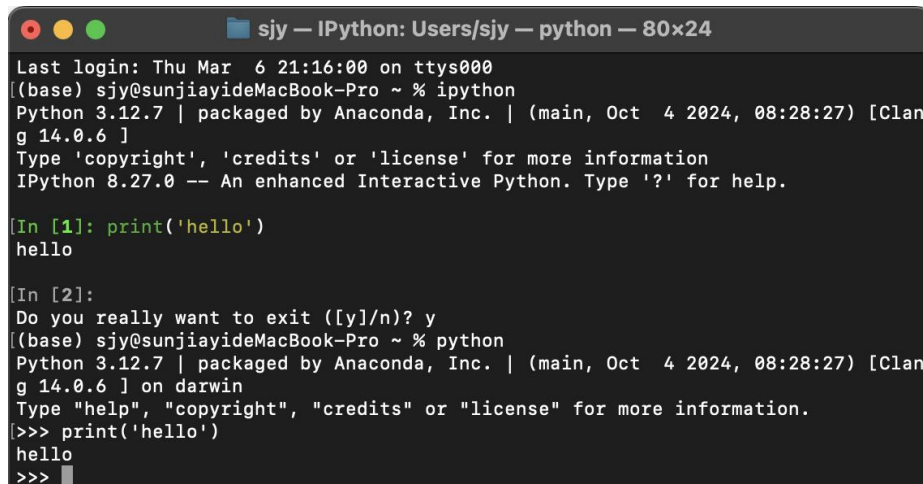
尚未暂存以备提交的变更：
  (使用 "git add <文件>..." 更新要提交的内容)
  (使用 "git restore <文件>..." 丢弃工作区的改动)
    修改：      README.md

未跟踪的文件：
  (使用 "git add <文件>..." 以包含要提交的内容)
    .DS_Store
    简单测试一下.pdf

修改尚未加入提交 (使用 "git add" 和/或 "git commit -a")
(base) sjy@sunjiayideMacBook-Pro week01 %
```

一些计算机新知识:

1、ipython 和 python 的区别: IPython 基于 Python, 扩展了其交互功能, 例子如下:

A terminal window titled 'sjy — IPython: Users/sjy — python — 80x24'. The first part shows IPython 8.27.0 being launched from a base environment. It displays the last login time, the IPython version, and the Python version (3.12.7) packaged by Anaconda. It prompts for help information. Then, the user enters '[In [1]: print('hello')]' and the output is 'hello'. Next, the user enters '[In [2]: Do you really want to exit ([y]/n)? y' and then '[In [2]: python]'. This switches the prompt to a standard Python shell. The user enters '>>> print('hello')' and the output is 'hello'. Finally, the user enters '>>>' and the prompt remains.

```

Last login: Thu Mar  6 21:16:00 on ttys000
(base) sjy@sunjiayideMacBook-Pro ~ % ipython
Python 3.12.7 | packaged by Anaconda, Inc. | (main, Oct  4 2024, 08:28:27) [Clang 14.0.6 ]
Type 'copyright', 'credits' or 'license' for more information
IPython 8.27.0 -- An enhanced Interactive Python. Type '?' for help.

[In [1]: print('hello')]
hello

[In [2]:
Do you really want to exit ([y]/n)? y
(base) sjy@sunjiayideMacBook-Pro ~ % python
Python 3.12.7 | packaged by Anaconda, Inc. | (main, Oct  4 2024, 08:28:27) [Clang 14.0.6 ] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> print('hello')
hello
>>>

```

2、pass 的作用: 是指环境变量