**第四周学习笔记**

**一、任务回顾**

数据存储与处理：数据在不通电时存储在磁盘或磁带中，需要呈现、计算加工或编解码时，通过 CPU 和内存，以进程为单元在操作系统中处理。

Python 解释器：启动后是一个进程，按照流程执行 Python 代码，调用操作系统或其他软件完成 “读取 —— 计算 —— 写入” 等工作。

本周目标：理解 Python 程序开发中的基本概念，如变量、函数、对象、类型等，并运用 pdb 调试器结合案例进行学习。

**二、任务步骤记录**

环境搭建

Fork 第 04 周打卡仓库至自己名下，并 Clone 到本地计算机。

使用 VS Code 打开项目目录，新建 environment.yml 文件，指定安装 Python 3.12，运行 conda env create 命令创建 Conda 环境。

新建 contacts.txt 文件，按格式写入联系人信息。

新建 main.py 文件用于编写 Python 代码。

工具安装与配置

在 VS Code 扩展商店安装 Python 扩展，以显示和选择 Python 解释器。

安装 Ruff 扩展并按文档配置，实现保存 .py 文件时自动规范化 Python 代码。

**三、Python 基本概念理解**

Python 语法保留字：如 if、for、while、def 等，具有特定语法功能，不能用作变量名等标识符。

语句和表达式

语句：执行特定操作，如 print('Hello, World!') 是一个输出语句。

表达式：计算产生一个值，如 2 + 3 是一个算术表达式，返回值为 5。

缩进：Python 使用缩进来表示代码块，同一级别的代码块缩进量必须一致，否则会导致语法错误。

局部变量、全局变量、LEGB 规则

局部变量：在函数内部定义的变量，作用域仅限于函数内部。

全局变量：在函数外部定义的变量，可在多个函数中访问，但在函数内部修改全局变量需要使用 global 关键字。

LEGB 规则：Python 查找变量的顺序，即 Local（局部）、Enclosing（嵌套作用域）、Global（全局）、Built-in（内置）。

函数的定义和调用

定义：使用 def 关键字，如 def add(a, b): return a + b。

调用：通过函数名和参数来执行函数，如 result = add(3, 5)。

字面值

字符串（str）：如 'Hello'、"World"。

整数（int）：如 10、-5。

列表（list）：如 [1, 2, 3]、['a', 'b', 'c']。

字典（dict）：如 {'name': 'Alice', 'age': 25}。

元组（tuple）：如 (1, 2, 3)，元组是不可变的。

运算符：如算术运算符 +、-、\*、/，比较运算符 ==、>、< 等。

形参、实参、返回值

形参：函数定义时括号内的参数，如 def func(a, b) 中的 a 和 b。

实参：函数调用时传递的实际参数，如 func(3, 4) 中的 3 和 4。

返回值：函数执行结束后返回的结果，使用 return 语句指定。

对象、类型、属性、方法

对象：Python 中一切皆对象，如字符串、列表、函数等都是对象。

类型：表示对象的种类，如 type('Hello') 返回 <class'str'>。

属性：对象所具有的特征，如字符串对象的 length 属性（在 Python 中通过 len() 函数获取）。

方法：对象可以执行的操作，如列表对象的 append() 方法用于添加元素。

**四、pdb 调试器使用**

启动调试：运行 python -m pdb main.py 命令以调试模式启动 Python 解释器。

常用命令

l（list）：显示当前代码行及前后几行代码。

n（next）：执行当前行代码，不进入函数调用内部。

p（print）：打印表达式的值，如 p contacts 可打印 contacts 列表的内容。

s（step）：步入函数调用内部，跟踪函数的执行。

pp（pretty print）：美观打印，用于格式化输出复杂数据结构。

c（continue）：继续执行代码，直到遇到下一个断点或程序结束。

**五、调试过程截图描述**

显示代码截图：使用 l 命令后，截图显示当前代码行及前后代码，清晰展示代码结构。

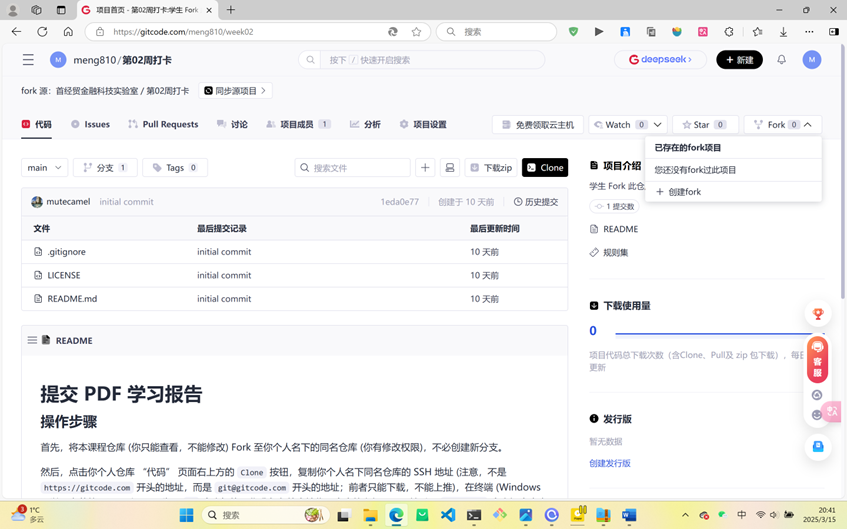
打印变量截图：使用 p 命令打印 contacts 变量，截图显示变量的内容，帮助理解数据在程序运行过程中的状态。

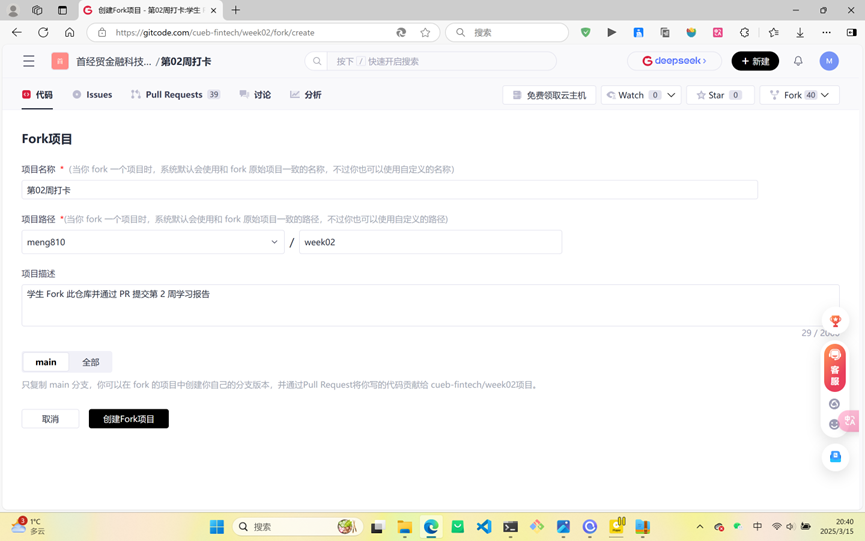
步入函数截图：当代码中有函数调用时，使用 s 命令步入函数内部，截图显示进入函数后的代码行，跟踪函数的执行流程。

**提交作业**

1. 先fork每周的main分支

点fork





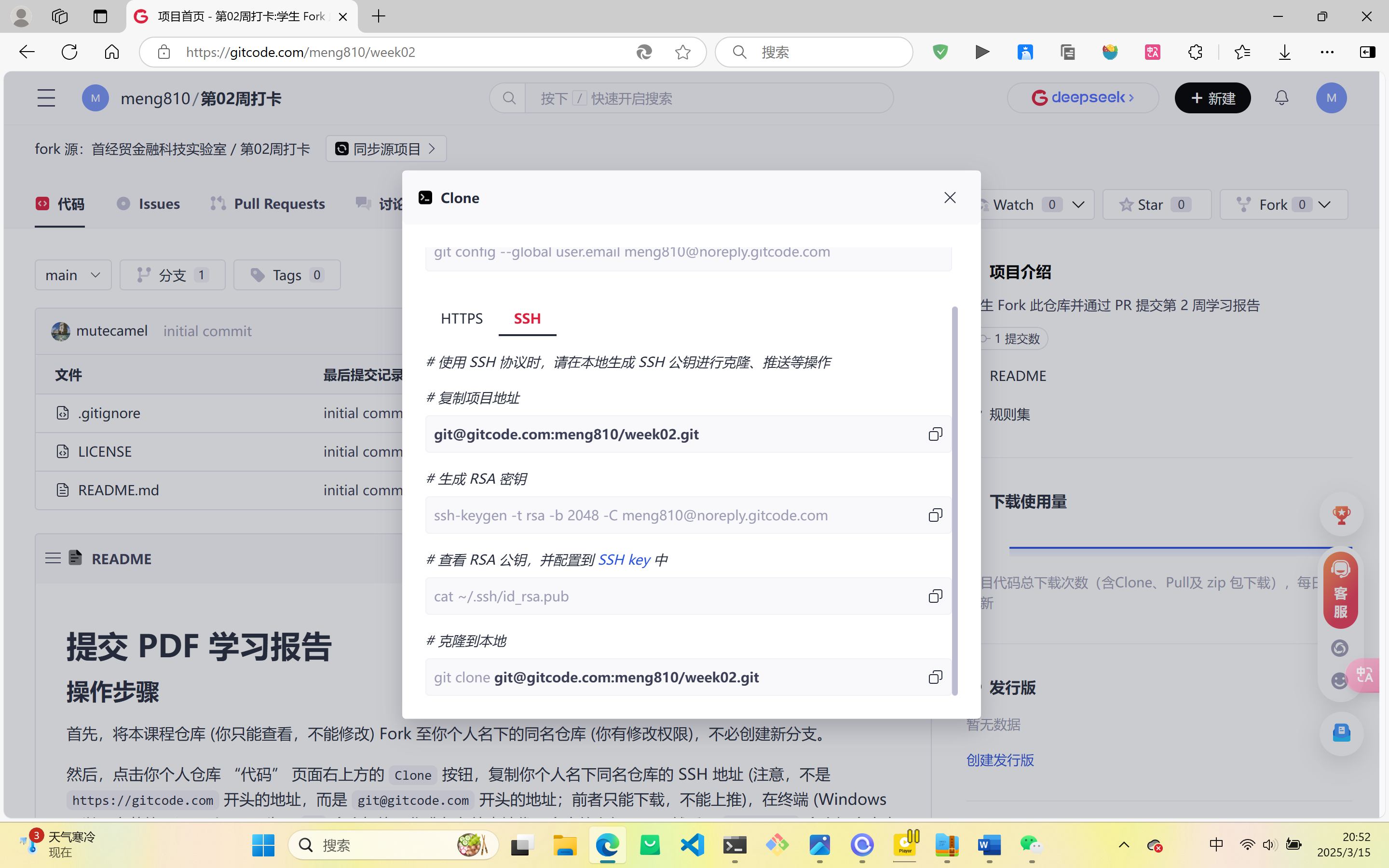
1. 终端操作

cd

pwd

cd repo

pwd



git clone [git@gitcode.com:meng810/week02.gi](mailto:git@gitcode.com:meng810/week02.gi)

ll

code week02/

cd week02/

ll

git status

git add 第二周笔记.pdf

git log

git status

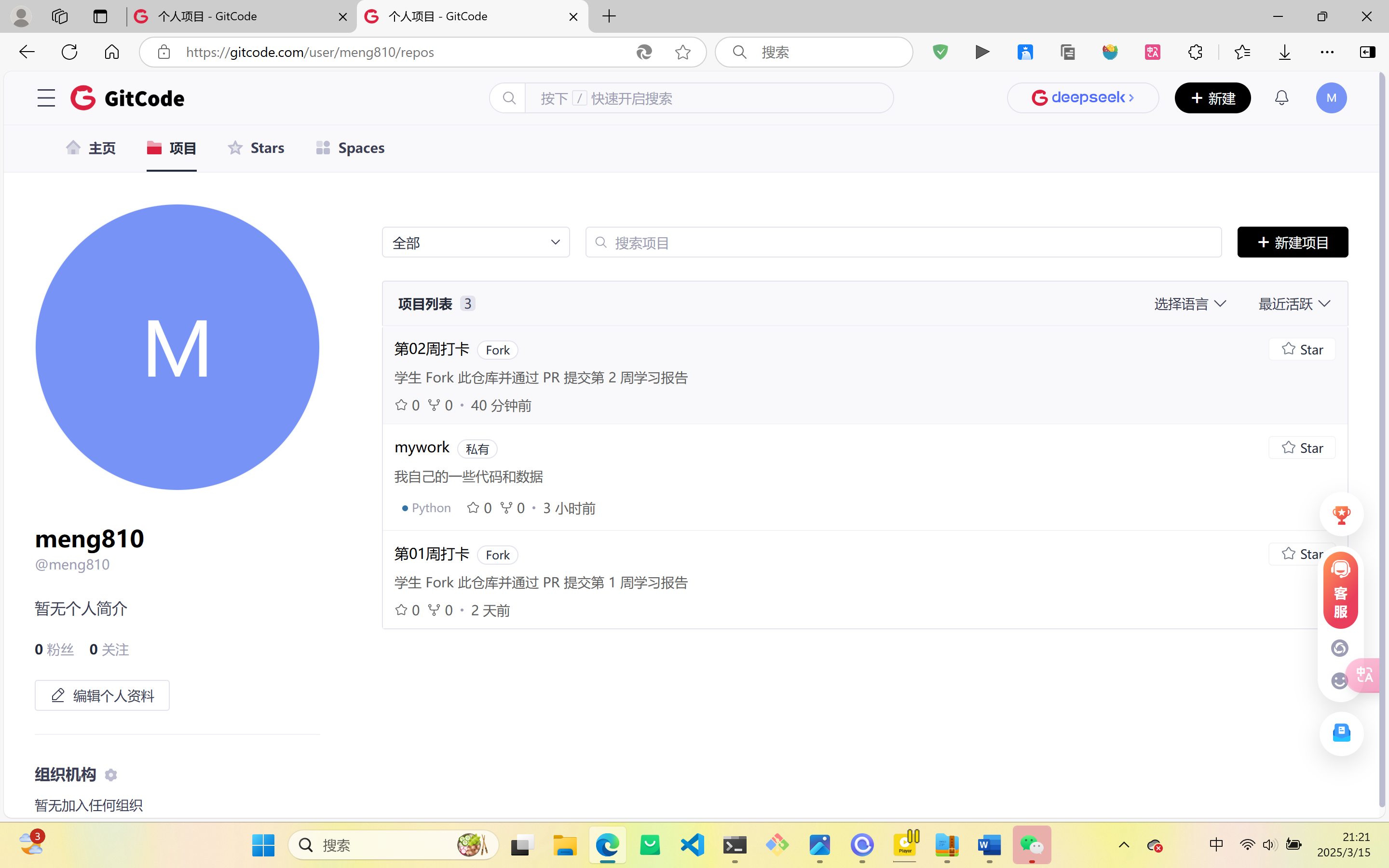
git commit -m "提交了第二周笔记"

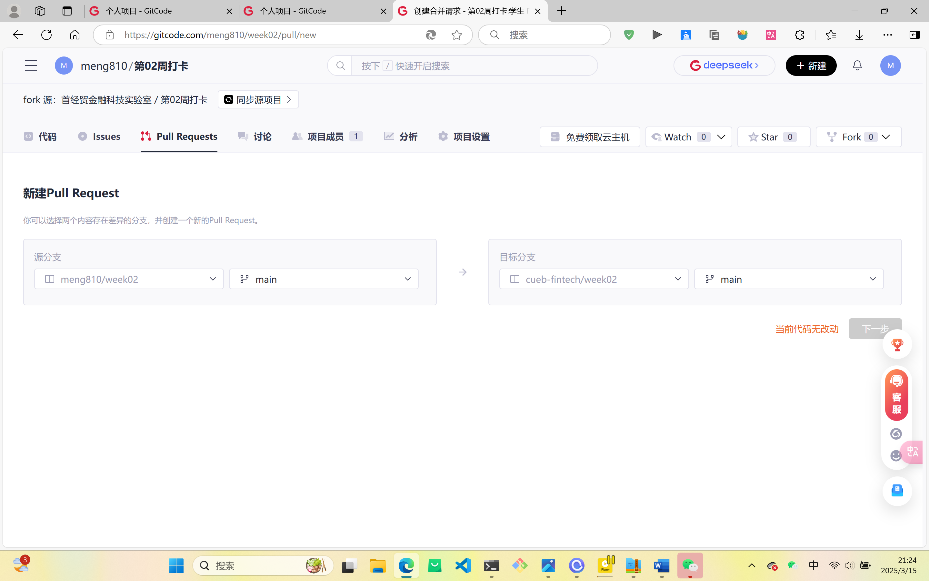
git status

git log

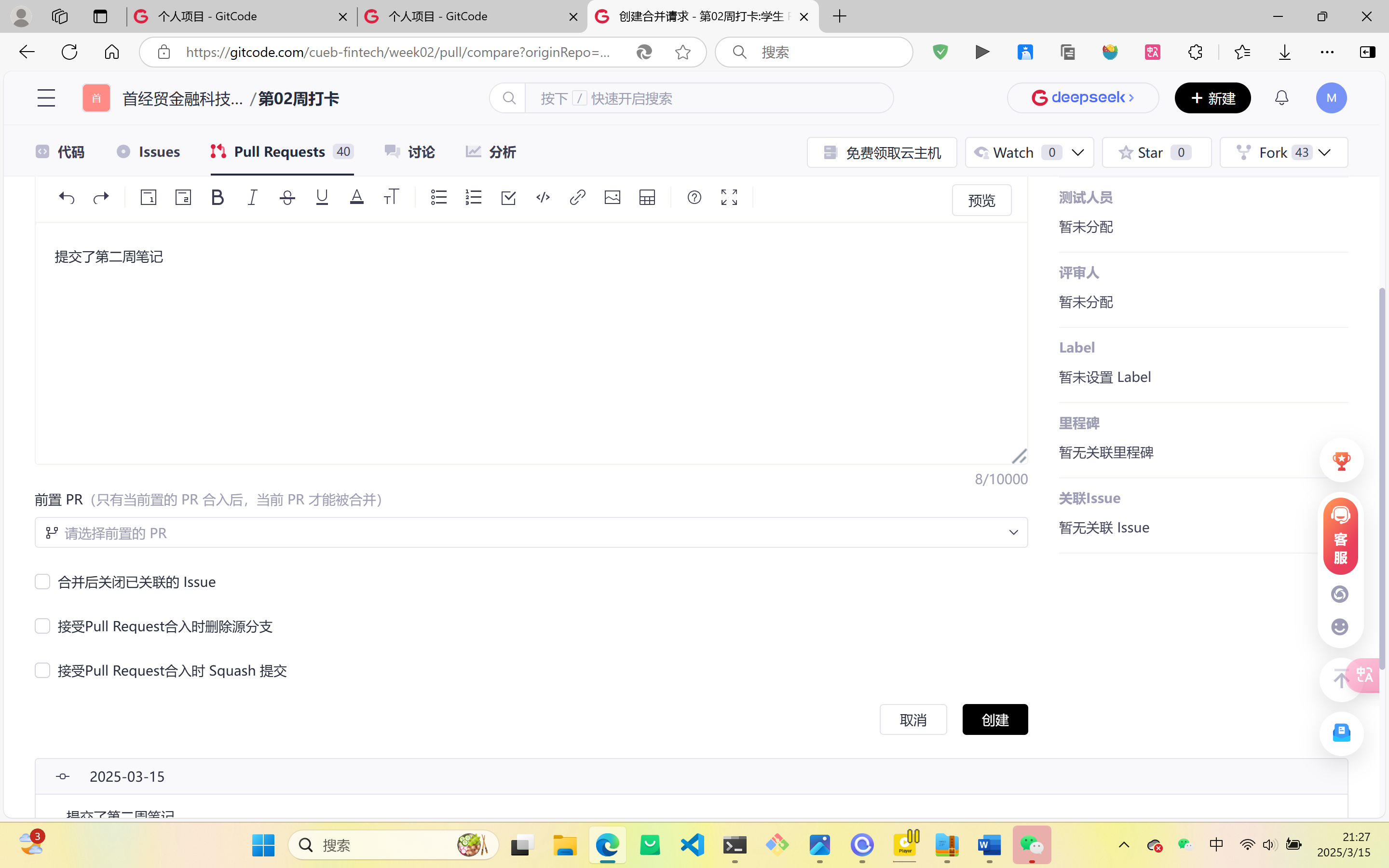
git push origin main

git log





直接点下一步



直接点创建