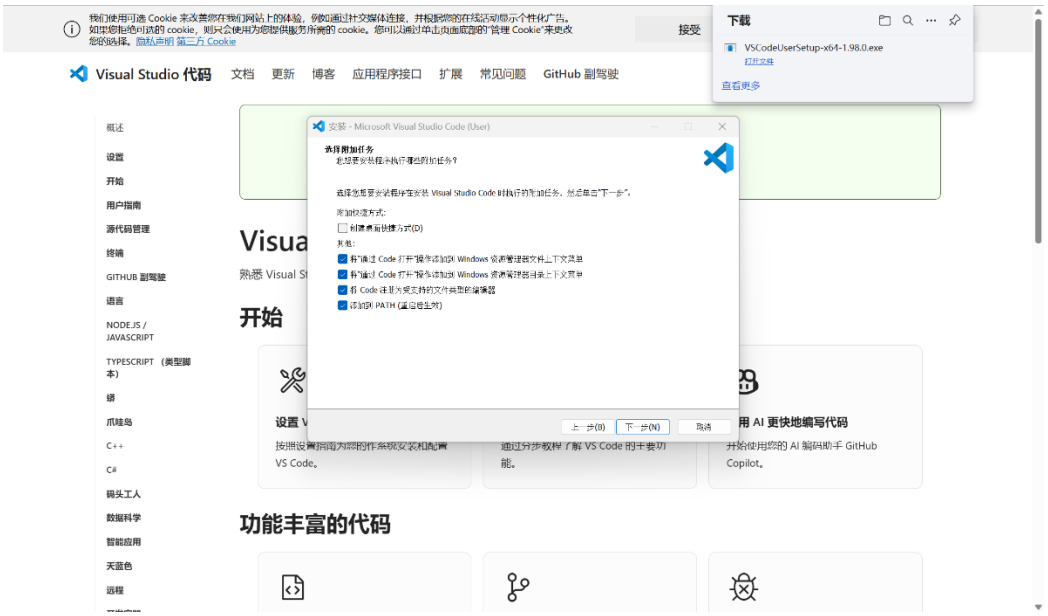
第一周学习报告

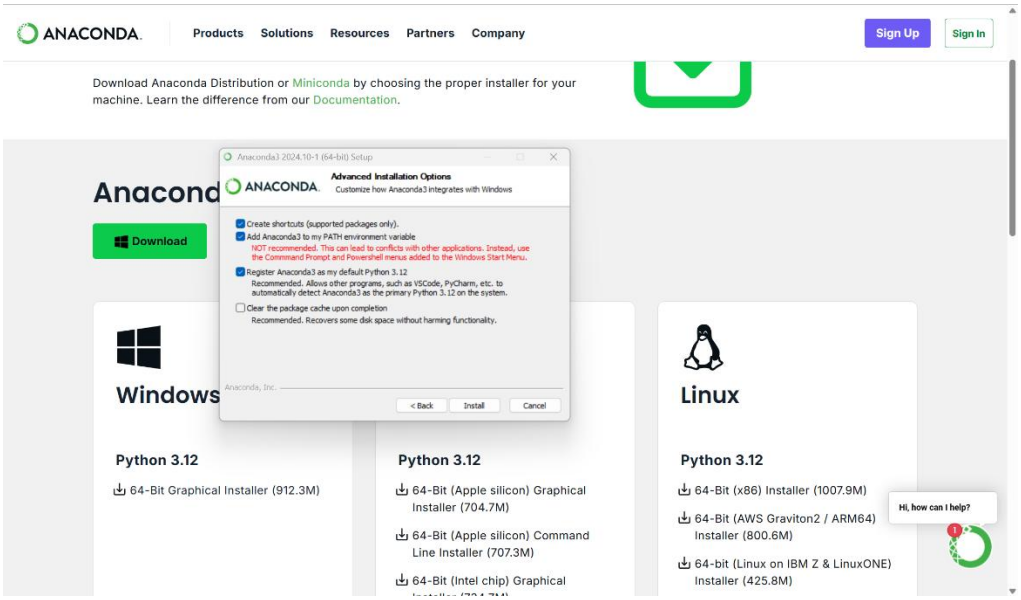
1. 软件下载

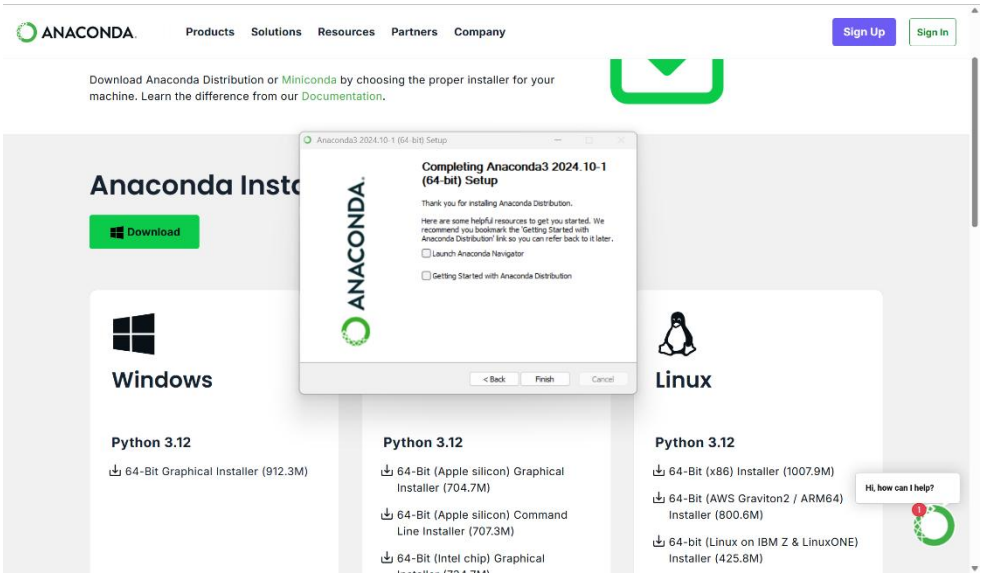


git --一切都是本地的
键入 / 搜索整个站点...
大约
文档
下载
GUI 客户端
标志
社区
下载 Windows 版
Git 2.48.1 Setup
Choosing the default editor used by Git
Which editor would you like Git to use?
Use Visual Studio Code as Git's default editor
Visual Studio Code is an Open Source, lightweight and powerful editor running as a desktop application. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages (such as C++, C#, Java, Python, PHP, Go) and runtimes (such as .NET and Unity).
(WARNING!) This will be installed only for this user.
Use this option to let Git use Visual Studio Code as its default editor.
Only show new options
Back
Next
Cancel
现在怎么办?
现在您已经下载了 Git，是时候开始使用它了。
阅读本书
下载 GUI
参与其中

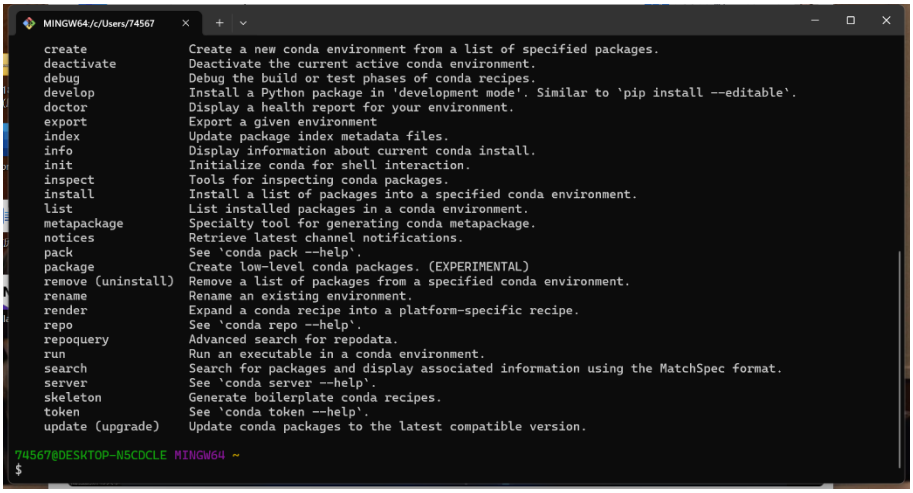
git --一切都是本地的
键入 / 搜索整个站点...
大约
文档
下载
GUI 客户端
标志
社区
下载 Windows 版
Git 2.48.1 Setup
Adjusting the name of the initial branch in new repositories
What would you like Git to name the initial branch after "git init"?
Let Git decide
Let Git use its default branch name (currently: "master") for the initial branch in newly created repositories. The Git project intends to change this default to a more inclusive name in the near future.
Override the default branch name for new repositories
NEW! Many teams already renamed their default branches; common choices are "main", "trunk" and "development". Specify the name "git init" should use for the initial branch:
main
This setting does not affect existing repositories.
Only show new options
Back
Next
Cancel
现在怎么办?
现在您已经下载了 Git，是时候开始使用它了。
阅读本书
下载 GUI
参与其中

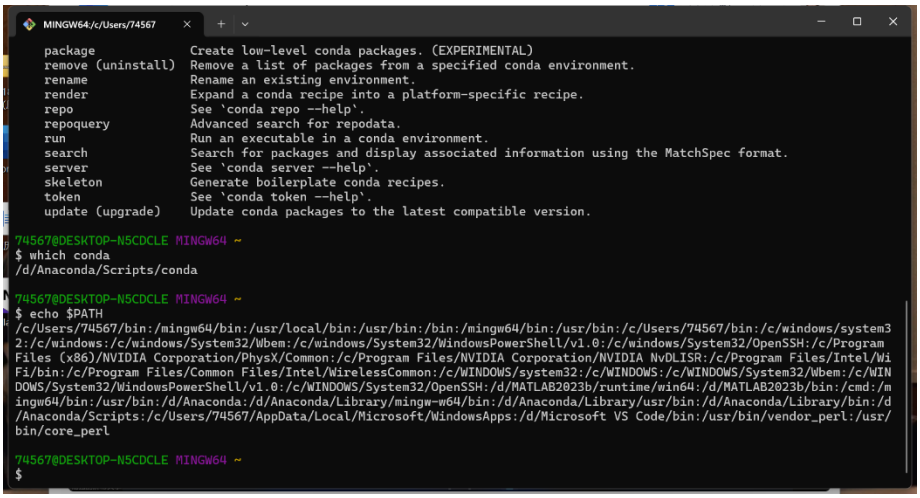
git --一切都是本地的
键入 / 搜索整个站点...
大约
文档
下载
GUI 客户端
标志
社区
下载 Windows 版
Git 2.48.1 Setup
Adjusting your PATH environment
How would you like to use Git from the command line?
Use Git from Git Bash only
This is the most cautious choice as your PATH will not be modified at all. You will only be able to use the Git command line tools from Git Bash.
Git from the command line and also from 3rd-party software
(Recommended) This option adds only some minimal Git wrappers to your PATH to avoid cluttering your environment with optional Unix tools. You will be able to use Git from Git Bash, the Command Prompt and the Windows PowerShell as well as any third-party software looking for Git in PATH.
Use Git and optional Unix tools from the Command Prompt
Both Git and the optional Unix tools will be added to your PATH.
Warning: This will override Windows tools like "find" and "sort". Only use this option if you understand the implications.
Only show new options
Back
Next
Cancel
现在怎么办?
现在您已经下载了 Git，是时候开始使用它了。
阅读本书
下载 GUI
参与其中

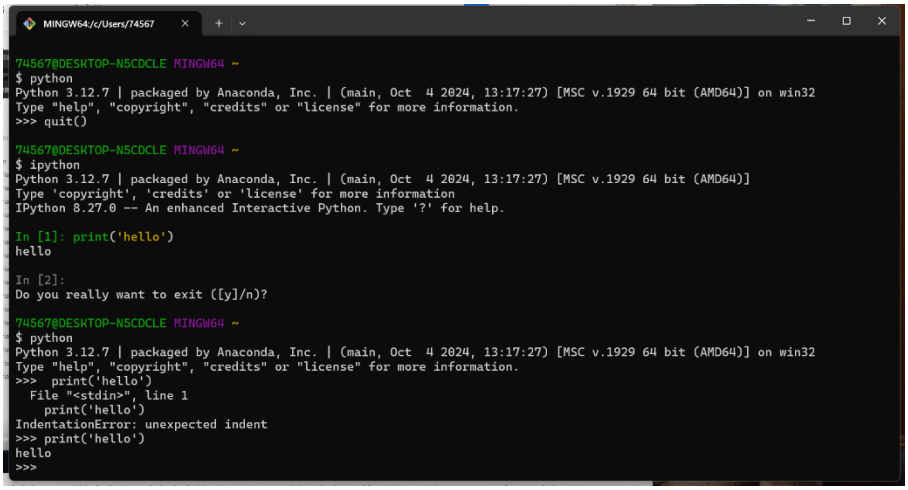

ANACONDA.
Products
Solutions
Resources
Partners
Company
Sign Up
Sign In
Download Anaconda Distribution or Miniconda by choosing the proper installer for your machine. Learn the difference from our Documentation.
Anaconda3 2024.10-1 (64-bit) Setup
ANACONDA.
Advanced Installation Options
Customize how Anaconda3 integrates with Windows
Create shortcuts (supported packages only).
Add Anaconda3 to my PATH environment variable
NOT recommended. This can lead to conflicts with other applications. Instead, use the Command Prompt and Powershell menus added to the Windows Start Menu.
Register Anaconda3 as my default Python 3.12
Recommended. Allows other programs, such as VSCode, PyCharm, etc. to automatically detect Anaconda3 as the primary Python 3.12 on the system.
Clear the package cache upon completion
Recommended. Recovers some disk space without harming functionality.
Anaconda, Inc.
< Back
Install
Cancel
Anaconda
Download
Windows
Python 3.12
64-Bit Graphical Installer (912.3M)
Python 3.12
64-Bit (Apple silicon) Graphical Installer (704.7M)
64-Bit (Apple silicon) Command Line Installer (707.3M)
64-Bit (Intel chip) Graphical
Linux
Python 3.12
64-Bit (x86) Installer (1007.9M)
64-Bit (AWS Graviton2 / ARM64) Installer (800.6M)
64-bit (Linux on IBM Z & LinuxONE) Installer (425.8M)
Hi, how can I help?

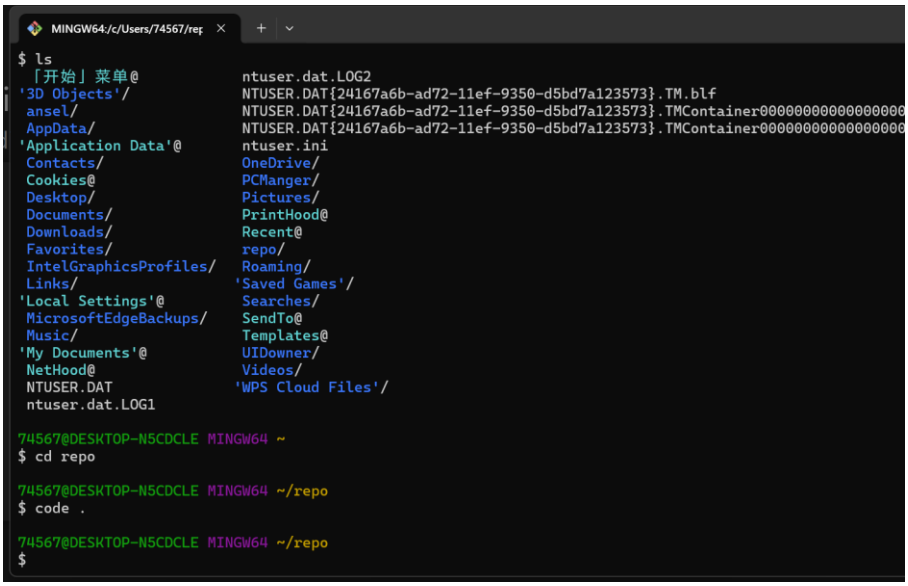


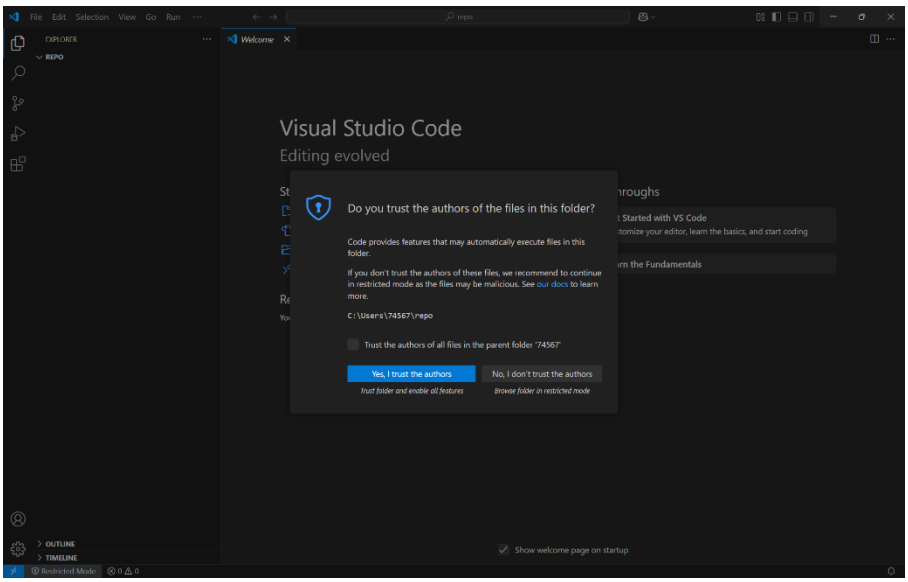
2. 生成自己设备的 SSH 密钥 (公钥-私钥对)，将公钥添加到自己的 GitCode 安全设置里

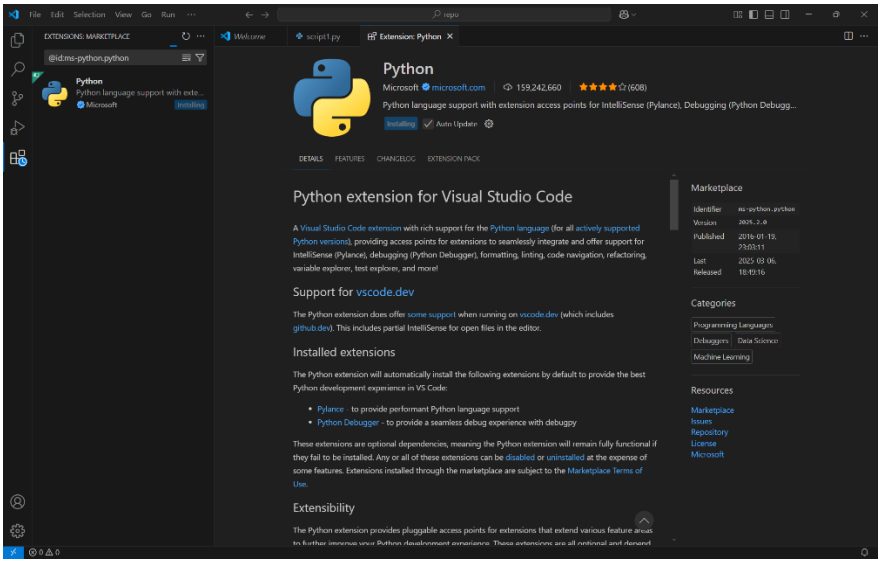


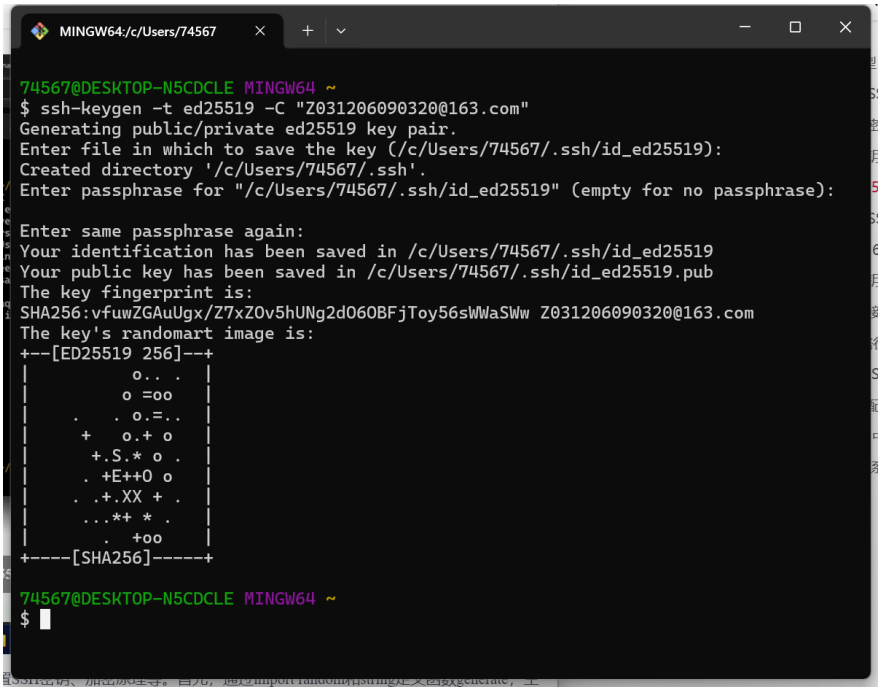


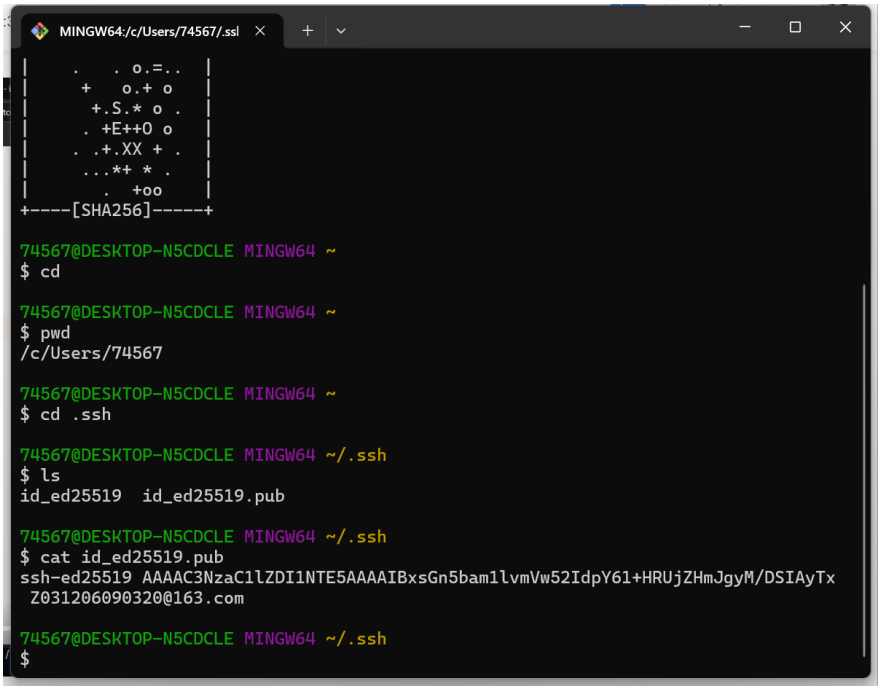

MINGW64:/c/Users/74567
74567@DESKTOP-N5CDCLE MINGW64 ~
$ python
Python 3.12.7 | packaged by Anaconda, Inc. | (main, Oct 4 2024, 13:17:27) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> quit()
74567@DESKTOP-N5CDCLE MINGW64 ~
$ ipython
Python 3.12.7 | packaged by Anaconda, Inc. | (main, Oct 4 2024, 13:17:27) [MSC v.1929 64 bit (AMD64)]
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)?
74567@DESKTOP-N5CDCLE MINGW64 ~
$ python
Python 3.12.7 | packaged by Anaconda, Inc. | (main, Oct 4 2024, 13:17:27) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print('hello')
File "<stdin>", line 1
print('hello')
IndentationError: unexpected indent
>>> print('hello')
hello
>>>

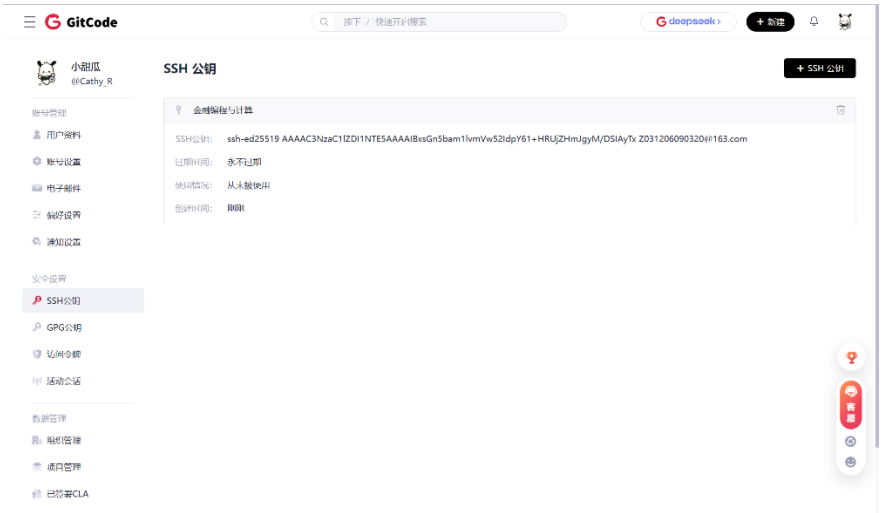

MINGW64:/c/Users/74567/rep
$ ls
「开始」菜单@ ntuser.dat.LOG2
'3D Objects'/ NTUSER.DAT{24167a6b-ad72-11ef-9350-d5bd7a123573}.TM.blf
ansel/ NTUSER.DAT{24167a6b-ad72-11ef-9350-d5bd7a123573}.TMContainer00000000000000000
AppData/ NTUSER.DAT{24167a6b-ad72-11ef-9350-d5bd7a123573}.TMContainer00000000000000000
'Application Data'@ ntuser.ini
Contacts/ OneDrive/
Cookies@ PCManger/
Desktop/ Pictures/
Documents/ PrintHood@
Downloads/ Recent@
Favorites/ repo/
IntelGraphicsProfiles/ Roaming/
Links/ 'Saved Games'/
'Local Settings'@ Searches/
MicrosoftEdgeBackups/ SendTo@
Music/ Templates@
'My Documents'@ UIDowner/
NetHood@ Videos/
NTUSER.DAT 'WPS Cloud Files'/
ntuser.dat.LOG1
74567@DESKTOP-N5CDCLE MINGW64 ~
$ cd repo
74567@DESKTOP-N5CDCLE MINGW64 ~/repo
$ code .
74567@DESKTOP-N5CDCLE MINGW64 ~/repo
$

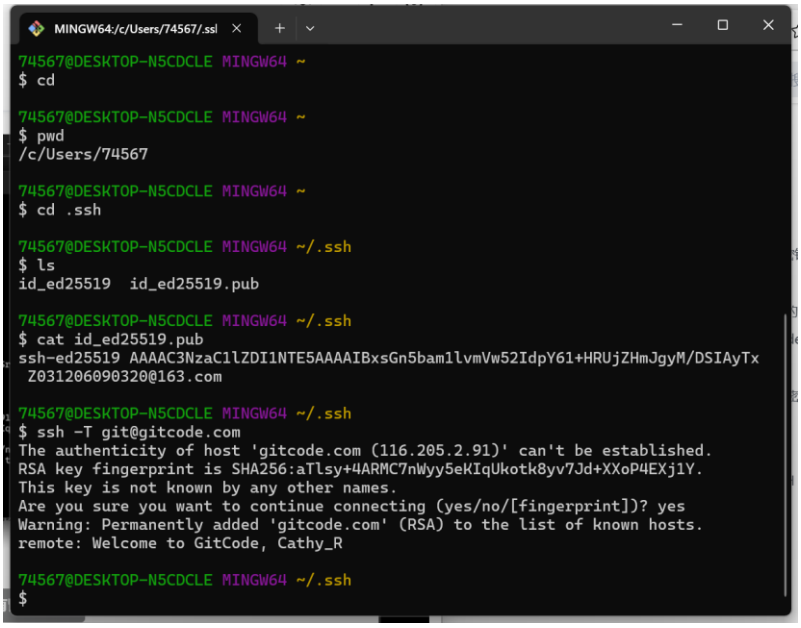

Visual Studio Code
Editing evolved
Do you trust the authors of the files in this folder?
Code provides features that may automatically execute files in this folder.
If you don't trust the authors of these files, we recommend to continue in restricted mode as the files may be malicious. See our docs to learn more.
C:\Users\74567\repo
Trust the authors of all files in the parent folder '74567'
Yes, I trust the authors
No, I don't trust the authors
Trust folder and enable all features
Browse folder in restricted mode
Show welcome page on startup

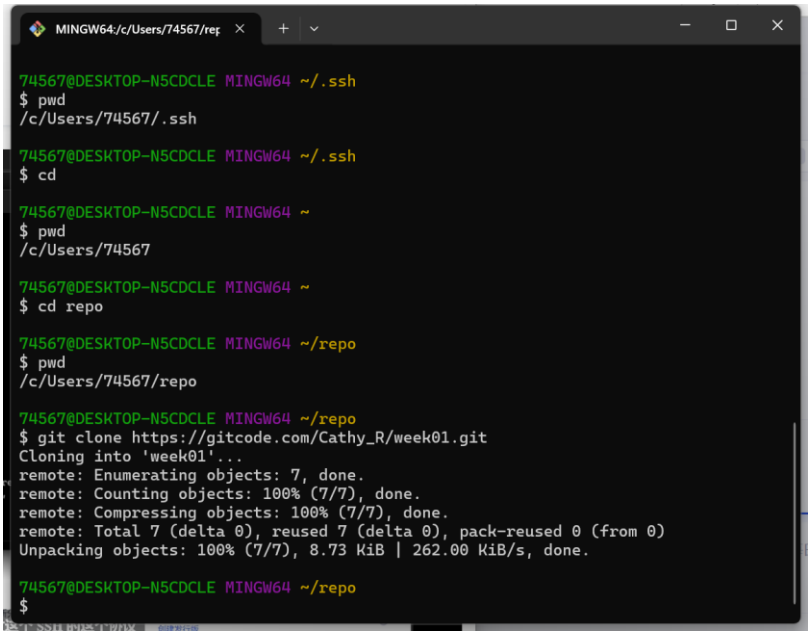

EXTENSIONS: MARKETPLACE
@id:ms-python.python
Python
Python language support with exte...
Microsoft
Installing
Welcome
script1.py
Extension: Python
Python
Microsoft microsoft.com
159,242,660
(608)
Python language support with extension access points for IntelliSense (Pylance), Debugging (Python Debugg...
Installing
Auto Update
DETAILS
FEATURES
CHANGELOG
EXTENSION PACK
Python extension for Visual Studio Code
A Visual Studio Code extension with rich support for the Python language (for all actively supported Python versions), providing access points for extensions to seamlessly integrate and offer support for IntelliSense (Pylance), debugging (Python Debugger), formatting, linting, code navigation, refactoring, variable explorer, test explorer, and more!
Support for vscode.dev
The Python extension does offer some support when running on vscode.dev (which includes github.dev). This includes partial IntelliSense for open files in the editor.
Installed extensions
The Python extension will automatically install the following extensions by default to provide the best Python development experience in VS Code:
Pylance - to provide performant Python language support
Python Debugger - to provide a seamless debug experience with debugpy
These extensions are optional dependencies, meaning the Python extension will remain fully functional if they fail to be installed. Any or all of these extensions can be disabled or uninstalled at the expense of some features. Extensions installed through the marketplace are subject to the Marketplace Terms of Use.
Extensibility
The Python extension provides pluggable access points for extensions that extend various feature areas
Marketplace
Identifier
Version
Published
Last Released
Categories
Programming Languages
Debuggers
Data Science
Machine Learning
Resources
Marketplace
Issues
Repository
License
Microsoft

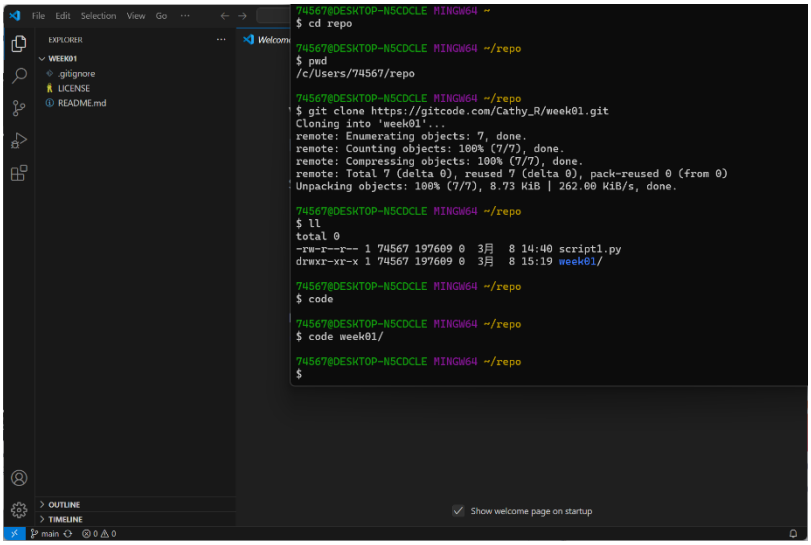

MINGW64:/c/Users/74567
74567@DESKTOP-N5CDCLE MINGW64 ~
$ ssh-keygen -t ed25519 -C "Z031206090320@163.com"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/c/Users/74567/.ssh/id_ed25519):
Created directory '/c/Users/74567/.ssh'.
Enter passphrase for "/c/Users/74567/.ssh/id_ed25519" (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/74567/.ssh/id_ed25519
Your public key has been saved in /c/Users/74567/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:vfuwZGAuUgx/Z7xZOv5hUNg2dO6OBFjToy56sWWaSWw Z031206090320@163.com
The key's randomart image is:
+--[ED25519 256]--+
| o.. . |
| o =oo |
| .   . o.=.. |
| +   o.+ o |
| +.S.* o . |
| . +E++0 o |
| . .+.XX + . |
| ...*+ * . |
| . +oo |
+----[SHA256]-----+
74567@DESKTOP-N5CDCLE MINGW64 ~
$

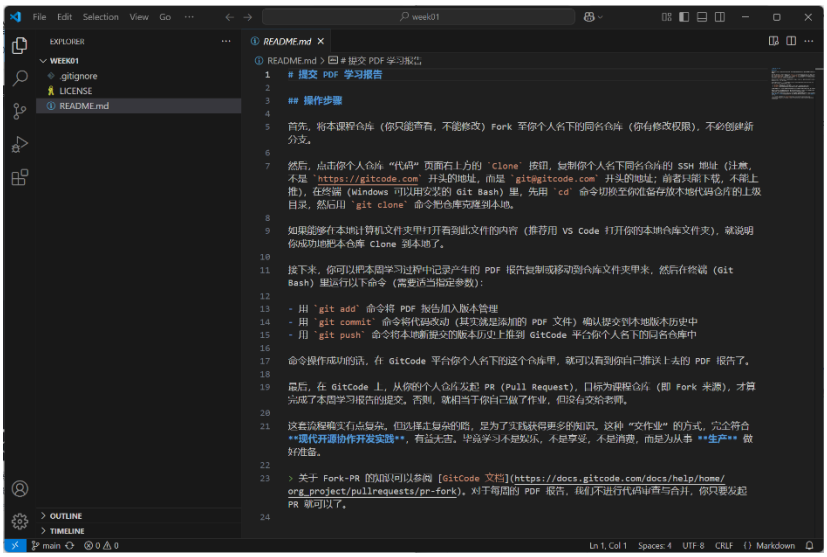

MINGW64:/c/Users/74567/.ssl
+----[SHA256]-----+
74567@DESKTOP-N5CDCLE MINGW64 ~
$ cd
74567@DESKTOP-N5CDCLE MINGW64 ~
$ pwd
/c/Users/74567
74567@DESKTOP-N5CDCLE MINGW64 ~
$ cd .ssh
74567@DESKTOP-N5CDCLE MINGW64 ~/.ssh
$ ls
id_ed25519 id_ed25519.pub
74567@DESKTOP-N5CDCLE MINGW64 ~/.ssh
$ cat id_ed25519.pub
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIBxsGn5bam1lvmVw52IdpY61+HRUjZHmJgyM/DSIAyTx
Z031206090320@163.com
74567@DESKTOP-N5CDCLE MINGW64 ~/.ssh
$


GitCode
SSH 公钥
+ SSH 公钥
SSH公钥: ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIBxsGn5bam1lvmVw52IdpY61+HRUjZHmJgyM/DSIAyTx Z031206090320@163.com
过期时间: 永不过期
使用情况: 从未被使用
创建时间: 刚刚


MINGW64:/c/Users/74567/.ssl
74567@DESKTOP-N5CDCLE MINGW64 ~
$ cd
74567@DESKTOP-N5CDCLE MINGW64 ~
$ pwd
/c/Users/74567
74567@DESKTOP-N5CDCLE MINGW64 ~
$ cd .ssh
74567@DESKTOP-N5CDCLE MINGW64 ~/.ssh
$ ls
id_ed25519 id_ed25519.pub
74567@DESKTOP-N5CDCLE MINGW64 ~/.ssh
$ cat id_ed25519.pub
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIBxsGn5bam1lvmVw52IdpY61+HRUjZHmJgyM/DSIAyTx
Z031206090320@163.com
74567@DESKTOP-N5CDCLE MINGW64 ~/.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, Cathy_R
74567@DESKTOP-N5CDCLE MINGW64 ~/.ssh
$


MINGW64:/c/Users/74567/reg
74567@DESKTOP-N5CDCLE MINGW64 ~/.ssh
$ pwd
/c/Users/74567/.ssh
74567@DESKTOP-N5CDCLE MINGW64 ~/.ssh
$ cd
74567@DESKTOP-N5CDCLE MINGW64 ~
$ pwd
/c/Users/74567
74567@DESKTOP-N5CDCLE MINGW64 ~
$ cd repo
74567@DESKTOP-N5CDCLE MINGW64 ~/repo
$ pwd
/c/Users/74567/repo
74567@DESKTOP-N5CDCLE MINGW64 ~/repo
$ git clone https://gitcode.com/Cathy_R/week01.git
Cloning into 'week01'...
remote: Enumerating objects: 7, done.
remote: Counting objects: 100% (7/7), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 7 (delta 0), reused 7 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (7/7), 8.73 KiB | 262.00 KiB/s, done.
74567@DESKTOP-N5CDCLE MINGW64 ~/repo
$


File Edit Selection View Go
EXPLORER
WEEK01
.gitignore
LICENSE
README.md
OUTLINE
TIMELINE
main
74567@DESKTOP-N5CDCLE MINGW64 ~
$ cd repo
74567@DESKTOP-N5CDCLE MINGW64 ~/repo
$ pwd
/c/Users/74567/repo
74567@DESKTOP-N5CDCLE MINGW64 ~/repo
$ git clone https://gitcode.com/Cathy_R/week01.git
Cloning into 'week01'...
remote: Enumerating objects: 7, done.
remote: Counting objects: 100% (7/7), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 7 (delta 0), reused 7 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (7/7), 8.73 KiB | 262.00 KiB/s, done.
74567@DESKTOP-N5CDCLE MINGW64 ~/repo
$ ll
total 0
-rw-r--r-- 1 74567 197609 0 3月 8 14:40 script1.py
drwxr-xr-x 1 74567 197609 0 3月 8 15:19 week01/
74567@DESKTOP-N5CDCLE MINGW64 ~/repo
$ code
74567@DESKTOP-N5CDCLE MINGW64 ~/repo
$ code week01/
74567@DESKTOP-N5CDCLE MINGW64 ~/repo
$
Show welcome page on startup


File Edit Selection View Go
week01
EXPLORER
WEEK01
.gitignore
LICENSE
README.md
README.md
README.md > # 提交 PDF 学习报告
提交 PDF 学习报告
操作步骤
首先，将本课程仓库（你只能查看，不能修改）Fork 至你个人名下的同名仓库（你有修改权限），不必创建新分支。
然后，点击你个人仓库“代码”页面右上方的 `Clone` 按钮，复制你个人名下同名仓库的 SSH 地址（注意，不是 `https://gitcode.com` 开头的地址，而是 `git@gitcode.com` 开头的地址；前者只能下载，不能上推），在终端（Windows 可以用安装的 Git Bash）里。先用 `cd` 命令切换至你准备存放本地代码仓库的上级目录，然后用 `git clone` 命令把仓库完整到本地。
如果能够在本地计算机文件夹里打开看到此文件的内容（推荐用 VS Code 打开你的本地仓库文件夹），就说明你成功地把本仓库 Clone 到本地了。
接下来，你可以把本周学习过程中记录产生的 PDF 报告复制或移动到仓库文件夹里来，然后在终端（Git Bash）里运行以下命令（需要适当指定参数）：
- 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
命令操作成功的话，在 GitCode 平台你个人名下的这个仓库里，就可以看到你自己推送上去的 PDF 报告了。
最后，在 GitCode 上，从你的个人仓库发起 PR（Pull Request），目标为课程仓库（即 Fork 来源），才算完成了本周学习报告的提交。否则，就相当于你自己做了作业，但没有交给老师。
这套流程确实有点复杂。但选择走复杂的路，是为了实践获得更多的知识。这种“交作业”的方式，完全符合**现代开源协作开发实践**，有益无害。毕竟学习不是娱乐，不是享受，不是消费，而是为从事 **生产** 做好准备。
> 关于 Fork-PR 的原理可以参阅 [GitCode 文档](https://docs.gitcode.com/docs/help/home/org_project/pullrequests/pr-fork)。对于每周的 PDF 报告，我们不进行代码审查与合并，你只要发起 PR 就可以了。
OUTLINE
TIMELINE
main
Ln 1, Col 1
Spaces: 4
UTF-8
CRLF
Markdown