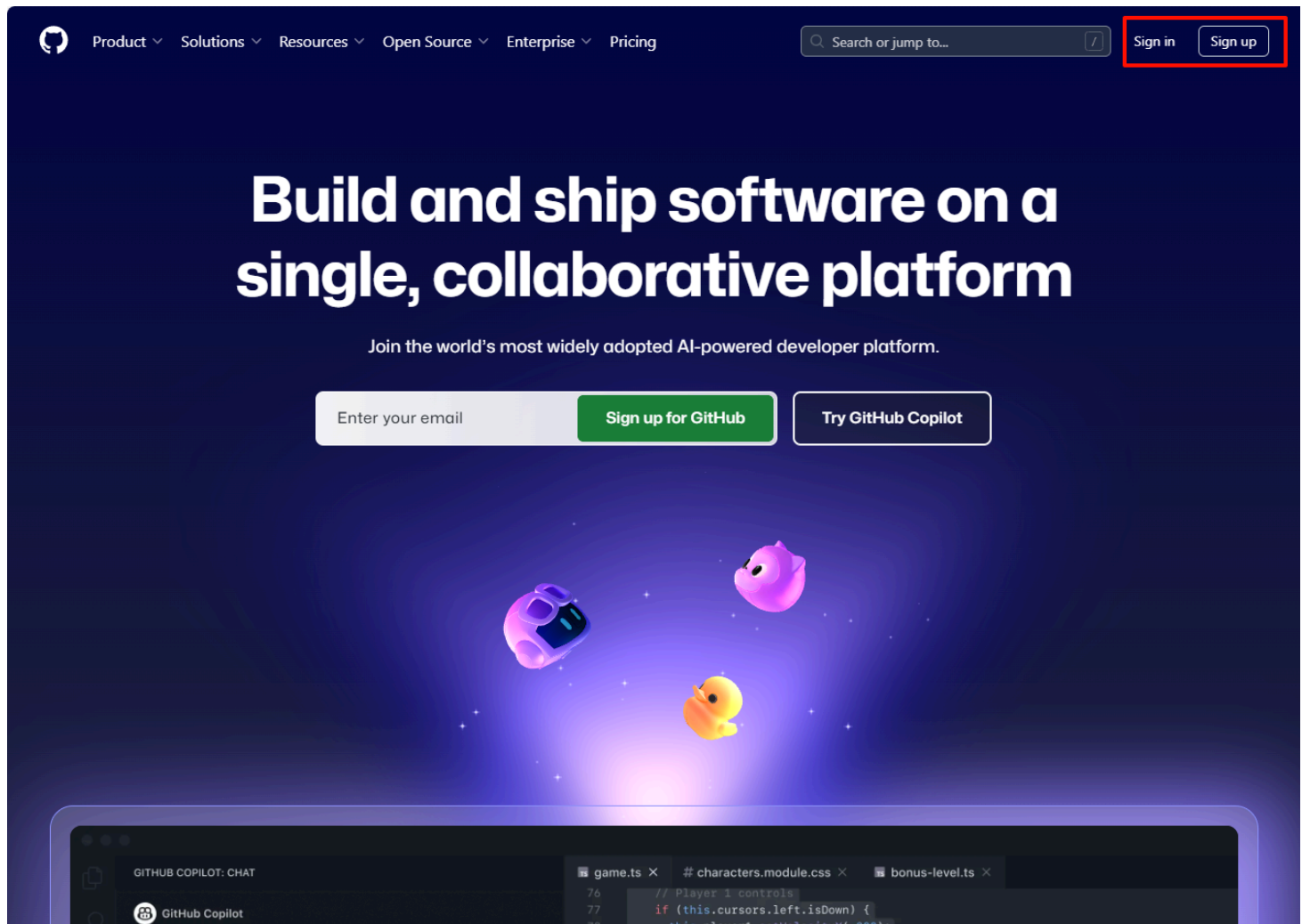


github的使用

github网址：[GitHub · Build and ship software on a single, collaborative platform · GitHub](#)

点击Sign up注册账号（填写邮箱、密码、用户名等信息），会有一个小游戏验证是否为真人，然后是邮箱验证，验证通过后就注册成功啦！



点击这两个地方（任选一个即可），创建一个新的仓库。

- Add a README file：初始化一个 README.md 文件在你的仓库里。README 文件用于介绍你的项目，包括项目说明、用法、安装步骤等内容。在 GitHub 上，这个文件会自动显示在仓库主页的最下方，是很多开源项目的“门面”。
- Add .gitignore：添加 .gitignore 文件，用于定义哪些文件或文件夹不应该被 提交到仓库。在开发过程中，常有一些本地配置文件、缓存、编译生成的临时文件不需要同步到远程仓库，防止泄漏私密信息或污染代码库。
- Choose a license：为你的代码仓库选择一个开源许可证。License 文件说明其他人是否可以使用、怎么使用、修改、分发你的项目。没有 License，别人很难知道能不能合法使用。

Dashboard

Q

Type / to search

Create your first project

Ready to start building? Create a repository for a new idea or bring over an existing repository to keep contributing to it.

Create repository

Import repository

Home

Ask Copilot

Get started with GitHub

Learn to code

Create a web app

Playlist

GitHub for beginners on YouTube

Designed to help you master the basics of GitHub, whether you're new to coding or looking to enhance your version control skills.

Start playlist

GitHub for Beginners

What is Git

Getting started

0/3 complete

Complete your profile

Add your personal bio and avatar — express yourself by building your social coding presence on GitHub.

Update profile

Chat with Copilot

Create your first repository

Start with GitHub Docs

New repository

Import repository

New codespace

New gist

New organization

New project


根据自己的需求按照顺序，创建自己的仓库

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (*).

Owner *

 Joker-11k ▾

Repository name *

aiSummerCamp2025-bigproject

✓ aiSummerCamp2025-bigproject is available.

1

Great repository names are short and memorable. Need inspiration? How about [congenial-broccoli](#) ?

Description (optional)

In this repository, I will start a project to show my final assignment.

2

3

☒



Public

Anyone on the internet can see this repository. You choose who can commit.

☐



Private

You choose who can see and commit to this repository.

4

Initialize this repository with:

☒

Add a README file

This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore


.gitignore template: None ▾

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: None ▾

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

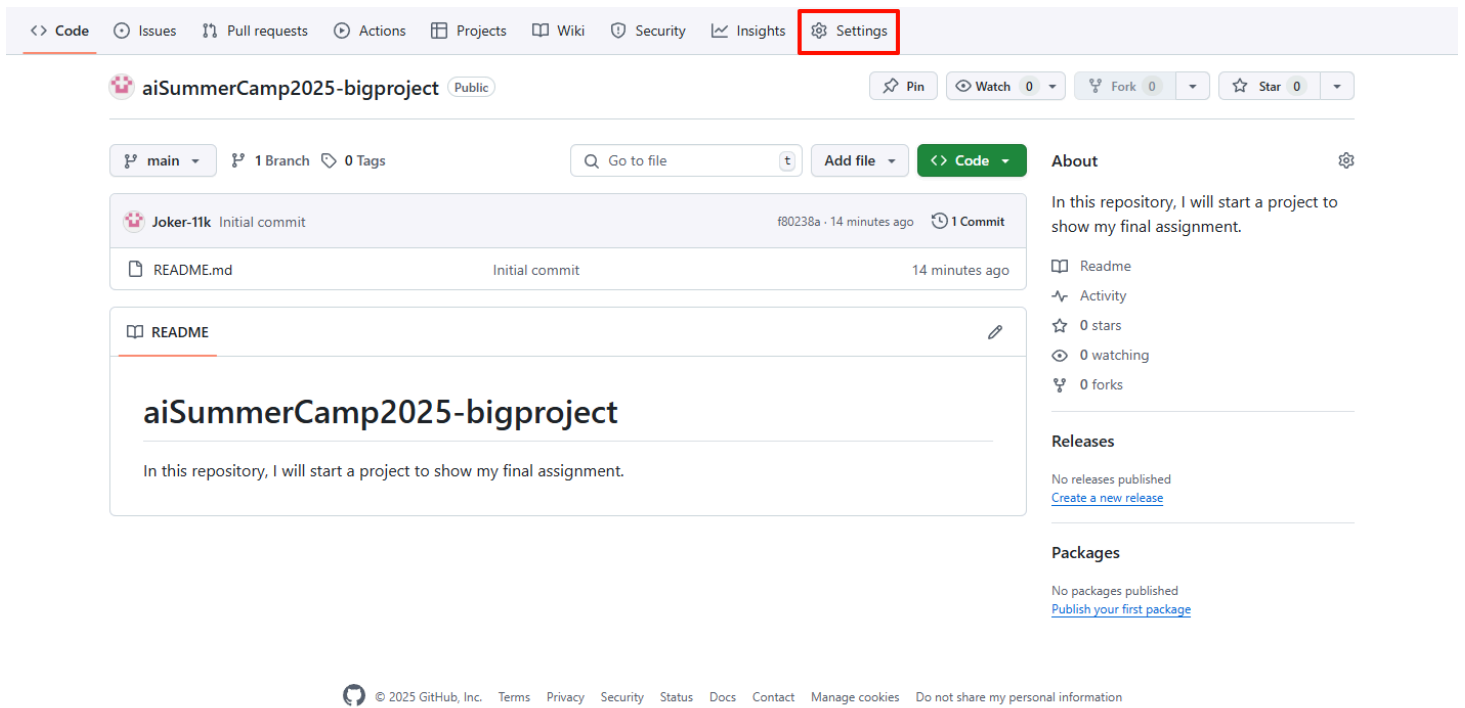
This will set  main as the default branch. Change the default name in your [settings](#).

 You are creating a public repository in your personal account.

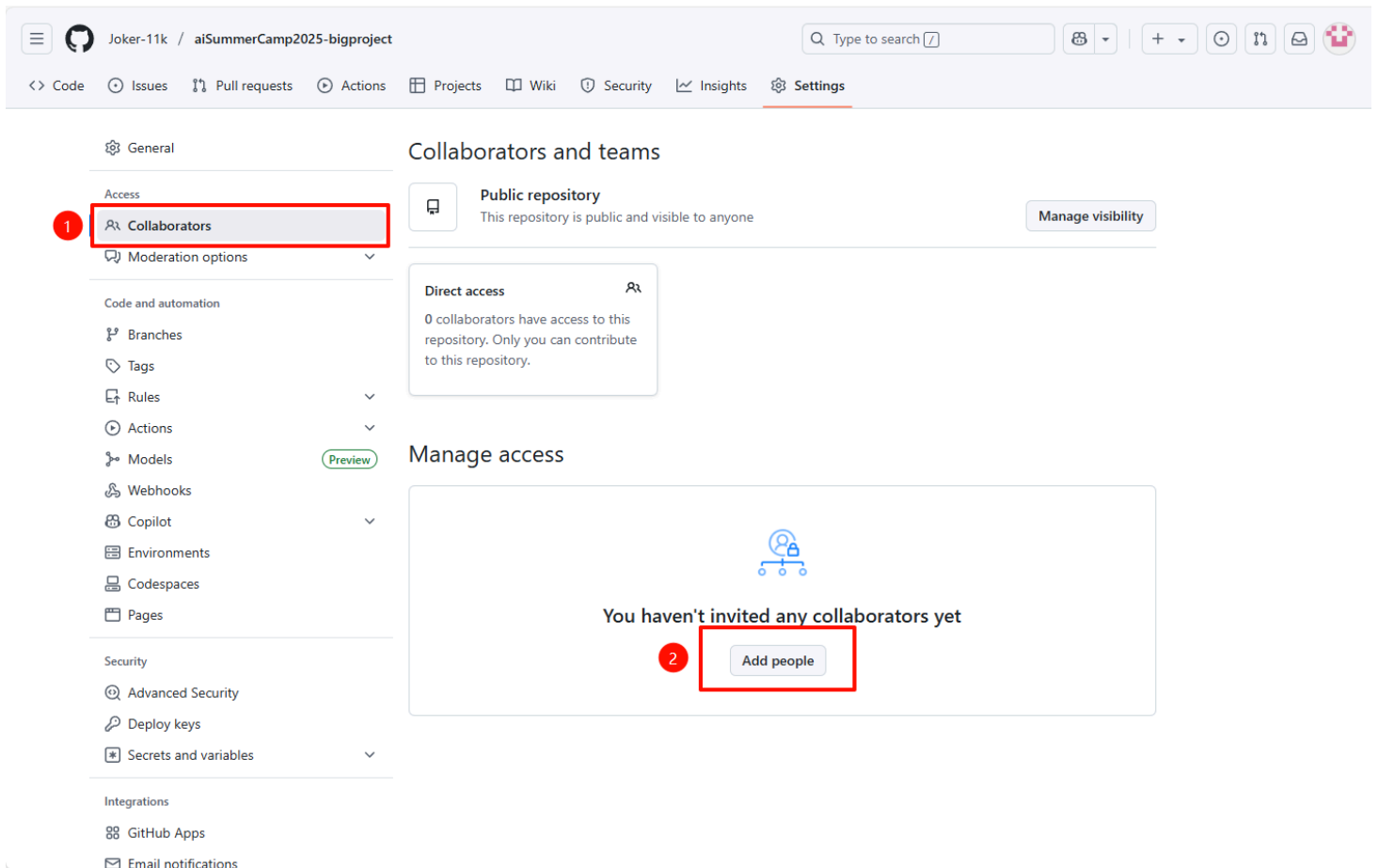
5

Create repository

创建完成后，就会出现以下界面



邀请小组其他成员一起管理



git的使用

git网址: [Git](https://git-scm.com/)



Git is a [free and open source](#) distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

Git is [easy to learn](#) and has a [tiny footprint with lightning fast performance](#). It outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like [cheap local branching](#), convenient [staging areas](#), and [multiple workflows](#).



About

The advantages of Git compared to other source control systems.



Documentation

Command reference pages, Pro Git book content, videos and other material.



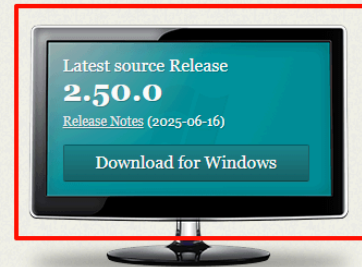
Downloads

GUI clients and binary releases for all major platforms.



Community

Get involved! Bug reporting, mailing list, chat, development and more.



Pro Git by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).



[Windows GUIs](#)



[Tarballs](#)

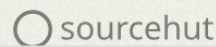


[Mac Build](#)



[Source Code](#)

Products providing Git hosting



Download for Windows

[Click here to download](#) the latest (2.50.0) x64 version of **Git for Windows**. This is the most recent maintained build. It was released **10 days ago**, on 2025-06-16.

Other Git for Windows downloads

Standalone Installer

[Git for Windows/x64 Setup.](#)

[Git for Windows/ARM64 Setup.](#)

Portable ("thumbdrive edition")

[Git for Windows/x64 Portable.](#)

[Git for Windows/ARM64 Portable.](#)

Using winget tool

Install [winget tool](#) if you don't already have it, then type this command in command prompt or Powershell.

```
winget install --id Git.Git -e --source winget
```

The current source code release is version **2.50.0**. If you want the newer version, you can build it from [the source code](#).

Now What?

Now that you have downloaded Git, it's time to start using it.



Read the Book

Dive into the Pro Git book and learn at your own pace.



Download a GUI

Several free and commercial GUI tools are available for the Windows platform.



Get Involved

A knowledgeable Git community is available to answer your questions.

按照它的安装流程傻瓜式安装，选项保持默认即可。

git需要创建自己的账户，使用下面的命令（注意空格，要不然会创建不成功）创建用户名和邮箱地址，邮箱地址最好和gitlab和github注册的邮箱地址相同。

```
git config --global user.name "Your name"
```

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

创建完成后，输入命令git config --list，就可以看见创建的信息，如下图所示。

```
YI@DESKTOP-I5G39AI MINGW64 ~/Desktop
$ git config --global user.name "Yukai Yi"

YI@DESKTOP-I5G39AI MINGW64 ~/Desktop
$ git config --global user.email "15827323587@163.com"

YI@DESKTOP-I5G39AI MINGW64 ~/Desktop
$ git config --list
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=schannel
core.autocrlf=true
core.fscache=true
core.symlinks=false
pull.rebase=false
credential.helper=manager
credential.https://dev.azure.com.usehttppath=true
init.defaultbranch=master
user.name=Yukai Yi
user.email=15827323587@163.com
```

输入命令ssh-keygen -t rsa -C"Your email"（最好和之前注册的邮箱相同），直接键盘上点enter就可以（需要点三次）。

```
YI@DESKTOP-I5G39AI MINGW64 ~/Desktop
$ ssh-keygen -t rsa -C"15827323587@163.com"
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/YI/.ssh/id_rsa):
Enter passphrase for "/c/Users/YI/.ssh/id_rsa" (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/YI/.ssh/id_rsa
Your public key has been saved in /c/Users/YI/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:T5K/kL1VKUz09KEQ7LtpIGVYrmk8DRlwj9baS0728P0 15827323587@163.com
The key's randomart image is:
+---[RSA 3072]-----+
| ..+= .00 .. |
| .0+. . 0.0...|
| 0.0+ . ... .|
| . 00 = + . |
| . *S . + 0 |
| ===0.. 0 |
| .0+0=.+ |
| . B. |
| + E |
+----[SHA256]-----+

YI@DESKTOP-I5G39AI MINGW64 ~/Desktop
$
```

可以在.ssh中看见两个文件，我们打开id_rsa.pub这个文件（可以使用vs code打开），里面是以ssh-rsa开头密钥。

此电脑 > Windows (C:) > 用户 > YI > .ssh >

名称	修改日期	类型	大小
id_rsa	2025/6/26 14:26	文件	3 KB
id_rsa.pub	2025/6/26 14:26	Microsoft Publis...	1 KB

然后我们打开github网站，按照顺序打开Settings--SSH and GPG keys--New SSH Key

Settings

1

Q Type [] to search

+

○

⌵

⌵

⌵

⌵

Okay, you have successfully deleted that key.

Yukai Yi (7k7kk7k7)

Your personal account

Go to your personal profile

Public profile

Account

Appearance

Accessibility

Notifications

Access

Billing and licensing

Emails

Password and authentication

Sessions

2 SSH and GPG keys

Organizations

Enterprises

Moderation

Code, planning, and automation

Repositories

Codespaces

Models

Packages

Conilot

SSH keys

3

New SSH key

There are no SSH keys associated with your account.

Check out our guide to [connecting to GitHub using SSH keys](#) or troubleshoot [common SSH problems](#).

GPG keys

New GPG key

There are no GPG keys associated with your account.

Learn how to [generate a GPG key and add it to your account](#).

Vigilant mode

☐ Flag unsigned commits as unverified

This will include any commit attributed to your account but not signed with your GPG or S/MIME key.
Note that this will include your existing unsigned commits.
[Learn about vigilant mode](#)

将上面复制的密钥粘贴，Title自定义。

Add new SSH Key

The screenshot shows the 'Add new SSH Key' form. It has three main sections: 'Title', 'Key type', and 'Key'. The 'Title' section has a text input field containing 'MySSHKey', highlighted by a red box and a red circle with the number '1'. The 'Key type' section has a dropdown menu set to 'Authentication Key', highlighted by a red box and a red circle with the number '2'. The 'Key' section has a large text area containing a blurred SSH key, highlighted by a red box. Below the 'Key' section is a green button labeled 'Add SSH key', highlighted by a red box and a red circle with the number '3'.

测试SSH是否好用

ssh -T [git@github.com](https://github.com)

```
YI@DESKTOP-I5G39AI MINGW64 ~/Desktop
$ ssh -T git@github.com
The authenticity of host 'github.com (20.205.243.166)' can't be established.
ED25519 key fingerprint is SHA256:+DiY3wvvV6TuJJhbpZisF/zLDA0zPMSvHdkr4UvCOqU.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'github.com' (ED25519) to the list of known hosts.
Hi 7k7kk7k7! You've successfully authenticated, but GitHub does not provide shell
access.
```

直接输入命令git

- 显示git命令的总览帮助文档
- 把常用的git子命令、辅助说明、使用格式都列出来，并提示如何获取更详细的帮助

```

YI@DESKTOP-I5G39AI MINGW64 ~/Desktop
$ git
usage: 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>]

These are common Git commands used in various situations:

start a working area (see also: git help tutorial)
  clone      Clone a repository into a new directory
  init       Create an empty Git repository or reinitialize an existing one

work on the current change (see also: git help everyday)
  add        Add file contents to the index
  mv         Move or rename a file, a directory, or a symlink
  restore    Restore working tree files
  rm         Remove files from the working tree and from the index

examine the history and state (see also: git help revisions)
  bisect     Use binary search to find the commit that introduced a bug
  diff       Show changes between commits, commit and working tree, etc
  grep       Print lines matching a pattern
  log        Show commit logs
  show       Show various types of objects
  status     Show the working tree status

grow, mark and tweak your common history
  backfill   Download missing objects in a partial clone
  branch     List, create, or delete branches
  commit     Record changes to the repository
  merge      Join two or more development histories together
  rebase     Reapply commits on top of another base tip
  reset      Reset current HEAD to the specified state
  switch     Switch branches
  tag        Create, list, delete or verify a tag object signed with GPG

collaborate (see also: git help workflows)
  fetch      Download objects and refs from another repository
  pull       Fetch from and integrate with another repository or a local branch
  push       Update remote refs along with associated objects

'git help -a' and 'git help -g' list available subcommands and some
concept guides. See 'git help <command>' or 'git help <concept>'
to read about a specific subcommand or concept.
See 'git help git' for an overview of the system.

```

- clone: 克隆一个远程仓库到本地，自动下载代码、分支、历史记录等。
例: `git clone https://....git`
- init: 本地初始化一个空仓库，新项目首次用git管理就用这个。例: `git init`
- add: 把文件的内容添加到暂存区。例: `git add name.text` 或 `git add .`
- mv: 移动或重命名文件（或目录等）。
例: `git mv oldname.txt newname.txt` 或 `git mv ./dir1/file.txt ./dir2/`
- restore: 撤销工作区或暂存区的改动，将文件恢复到某个之前的状态。
例: `git restore name.txt` 或 `git restore --staged name.txt`
- rm: 将文件从工作区和暂存区删除。例: `git rm name.txt` 或 `git rm --cached name.txt`

- commit: 将已经 git add 到暂存区的文件，保存到你的本地仓库中，形成一个新的版本记录。
例: `git commit -m "本次提交的说明"`
- fetch: 从远程仓库获取最新的分支、标签信息等你的本地仓库，但不会自动合并和修改。
例: `git fetch`
- pull: 从远程仓库下载最新的历史记录并自动尝试合并到你当前所在的本地分支（会同时更新你的本地仓库和工作区）。例: `git fetch`
- push: 将本地分支上的提交上传到远程仓库。例 `git push -u origin master`

找到要上传的文件夹，鼠标右键选择Open Git Bash here，输入git init

它会在当前目录下创建一个名为 `.git` 的隐藏文件夹，有了这个文件夹就可以使用各种命令去管理文件。

```
YI@DESKTOP-I5G39AI MINGW64 /d/其他文件/暑期AI训练营/aiSummerCamp2025/bigProject
(master)
$ git init
Initialized empty Git repository in D:/其他文件/暑期AI训练营/aiSummerCamp2025/bigProject/.git/
```

名称	修改日期	类型
readme.md	2025/6/26 16:36	Markdown 源文件
报告封面.docx	2025/6/26 16:36	Microsoft Word ...
大作业选题.docx	2025/6/26 16:36	Microsoft Word ...
附件-课程报告目录.docx	2025/6/26 16:36	Microsoft Word ...

名称	修改日期	类型
.git	2025/6/26 16:40	文件夹
readme.md	2025/6/26 16:36	Markdown 源文件
报告封面.docx	2025/6/26 16:36	Microsoft Word ...
大作业选题.docx	2025/6/26 16:36	Microsoft Word ...
附件-课程报告目录.docx	2025/6/26 16:36	Microsoft Word ...

输入git add .命令

- 将当前目录及其子目录下所有有变动的文件（包括新建、修改、删除的文件）全部添加到暂存区，准备提交
- 也可以写更具体的文件名，比如git add `readme.md`，只加这一个文件。

输入git commit -m "first commit"命令

- 把已经添加到暂存区的更改，生成一个快照，永久保存到你的本地git仓库的历史记录中
- “ ”中的内容是本次提交的信息注释，可以写成自己需要的内容

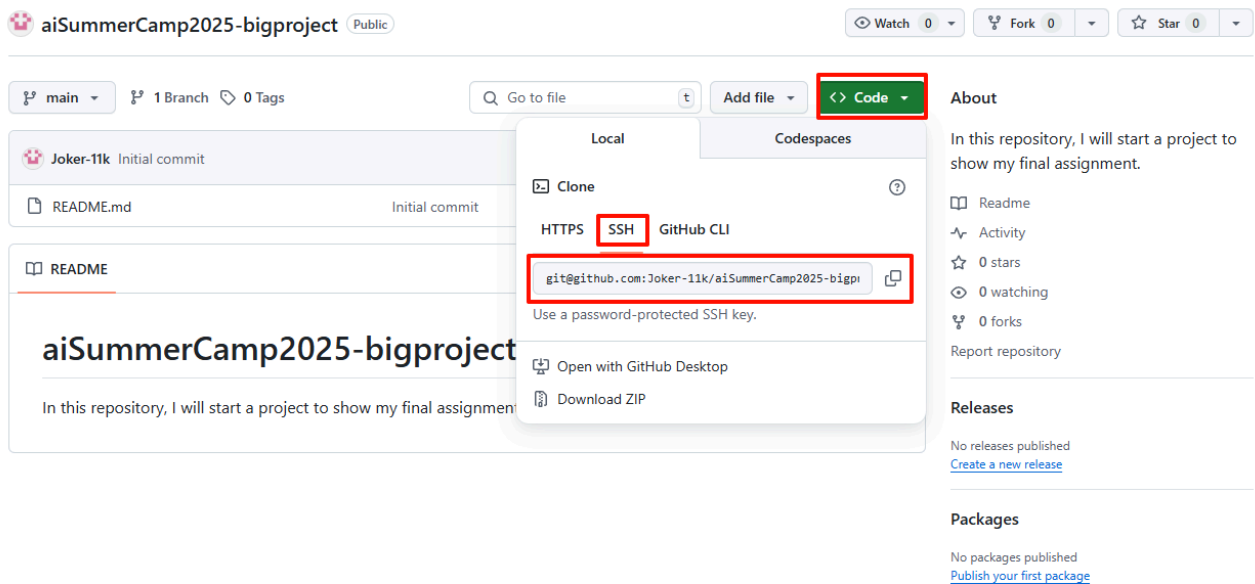
```

YI@DESKTOP-I5G39AI MINGW64 /d/其他文件/暑期AI训练营/aiSummerCamp2025/bigProject
(master)
$ git add .
warning: in the working copy of 'readme.md', LF will be replaced by CRLF the next
time Git touches it

YI@DESKTOP-I5G39AI MINGW64 /d/其他文件/暑期AI训练营/aiSummerCamp2025/bigProject
(master)
$ git commit -m "first commit"
[master (root-commit) b6ad1aa] first commit
 4 files changed, 3 insertions(+)
 create mode 100644 readme.md
 create mode 100644 "\\345\\244\\247\\344\\275\\234\\344\\270\\232\\351\\200\\211\\351\\242\\23
0.docx"
 create mode 100644 "\\346\\212\\245\\345\\221\\212\\345\\260\\201\\351\\235\\242.docx"
 create mode 100644 "\\351\\231\\204\\344\\273\\266-\\350\\257\\276\\347\\250\\213\\346\\212\\2
45\\345\\221\\212\\347\\233\\256\\345\\275\\225.docx"

```

找到远程仓库的SSH，输入命令git remote add origin 远程仓库的URL



```

YI@DESKTOP-I5G39AI MINGW64 /d/其他文件/暑期AI训练营/aiSummerCamp2025/bigProject
(master)
$ git remote add origin git@github.com:Joker-11k/aiSummerCamp2025-bigproject.git

```

输入git remote -v，可以查看配置远程仓库的URL

```

YI@DESKTOP-I5G39AI MINGW64 /d/其他文件/暑期AI训练营/aiSummerCamp2025/bigProject
(master)
$ git remote -v
origin git@github.com:Joker-11k/aiSummerCamp2025-bigproject.git (fetch)
origin git@github.com:Joker-11k/aiSummerCamp2025-bigproject.git (push)

```

最后是推送指令，推送到指定仓库的master分支下

git push -u origin master

```
YI@DESKTOP-I5G39AI MINGW64 /d/其他文件/暑期AI训练营/aiSummerCamp2025/bigProject
(master)
$ git push -u origin master
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 20 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 58.36 KiB | 198.00 KiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:   https://github.com/Joker-11k/aiSummerCamp2025-bigproject/pull/new/m
aster
remote:
To github.com:Joker-11k/aiSummerCamp2025-bigproject.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

👤 master had recent pushes 47 seconds ago

Compare & pull request

🔗 main 2 Branches 0 Tags

🔍 Go to file t Add file <> Code

Switch branches/tags

🔍 Find or create a branch...

Branches Tags

✓ main default

master

View all branches

f80238a · 1 hour ago 1 Commit

Initial commit 1 hour ago

5-bigproject

In this repository, I will start a project to show my final assignment.

📖 Readme

🔔 Activity

☆ 0 stars

👁 0 watching

🍴 0 forks

Report repository

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

👤 aiSummerCamp2025-bigproject Public

👁 Watch 0 🍴 Fork 0 ☆ Star 0

👤 master 2 Branches 0 Tags

🔍 Go to file t Add file <> Code

This branch is 1 commit ahead of, 1 commit behind main .

👤 Contribute

7k7kk7k7 first commit b6ad1aa · 25 minutes ago 1 Commit

📄 readme.md	first commit	25 minutes ago
📄 大作业选题.docx	first commit	25 minutes ago
📄 报告封面.docx	first commit	25 minutes ago
📄 附件-课程报告目录.docx	first commit	25 minutes ago

📖 README

Big Final project

Put the final project assignment files like data and requirements here.

About

In this repository, I will start a project to show my final assignment.

📖 Readme 🔔 Activity ☆ 0 stars 👁 0 watching 🍴 0 forks Report repository

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

如何查看提交的信息

- 最常用的是git log

```
YI@DESKTOP-I5G39AI MINGW64 /d/其他文件/暑期AI训练营/aiSummerCamp2025/bigProject
(master)
$ git log
commit b6ad1aa6c0106ad07aa87729e56d1e277b42d75f (HEAD -> master, origin/master)
Author: Yukai Yi <15827323587@163.com>
Date: Thu Jun 26 14:55:56 2025 +0800

    first commit
```

- 在仓库主页，点击Commit也可查看

aiSummerCamp2025-bigprojectPublic

Watch0

Fork0

Star0

master had recent pushes 13 minutes ago

Compare & pull request

master2 Branches0 Tags

Go to file

Add file

Code

This branch is 1 commit ahead of, 1 commit behind main.

7k7kk7k7 first commitb6ad1aa · 30 minutes ago1 Commit

readme.md	first commit	30 minutes ago
大作业选题.docx	first commit	30 minutes ago
报告封面.docx	first commit	30 minutes ago
附件-课程报告目录.docx	first commit	30 minutes ago

README

Big Final project

Put the final project assignment files like data and requirements here.

About

In this repository, I will start a project to show my final assignment.

Readme

Activity

0 stars

0 watching

0 forks

Report repository

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

Commits

master

All usersAll time

Commits on Jun 26, 2025

first commit

7k7kk7k7 committed 33 minutes ago

b6ad1aa

Branch

分支就是你在项目中开辟出的“独立开发路线”，可以想象成“并行宇宙”，每个分支有各自的提交历史。

git fetch --all获取最新远程分支列表

查看分支的命令

- git branch: 查看本地分支

- git branch -r: 查看远程分支
- git branch -a: 查看所有分支 (本地+远程)

```
YI@DESKTOP-I5G39AI MINGW64 /d/其他文件/暑期AI训练营/aiSummerCamp2025/bigProject
(master)
$ git branch
* master

YI@DESKTOP-I5G39AI MINGW64 /d/其他文件/暑期AI训练营/aiSummerCamp2025/bigProject
(master)
$ git branch -r
  origin/HEAD -> origin/main
  origin/main
  origin/master

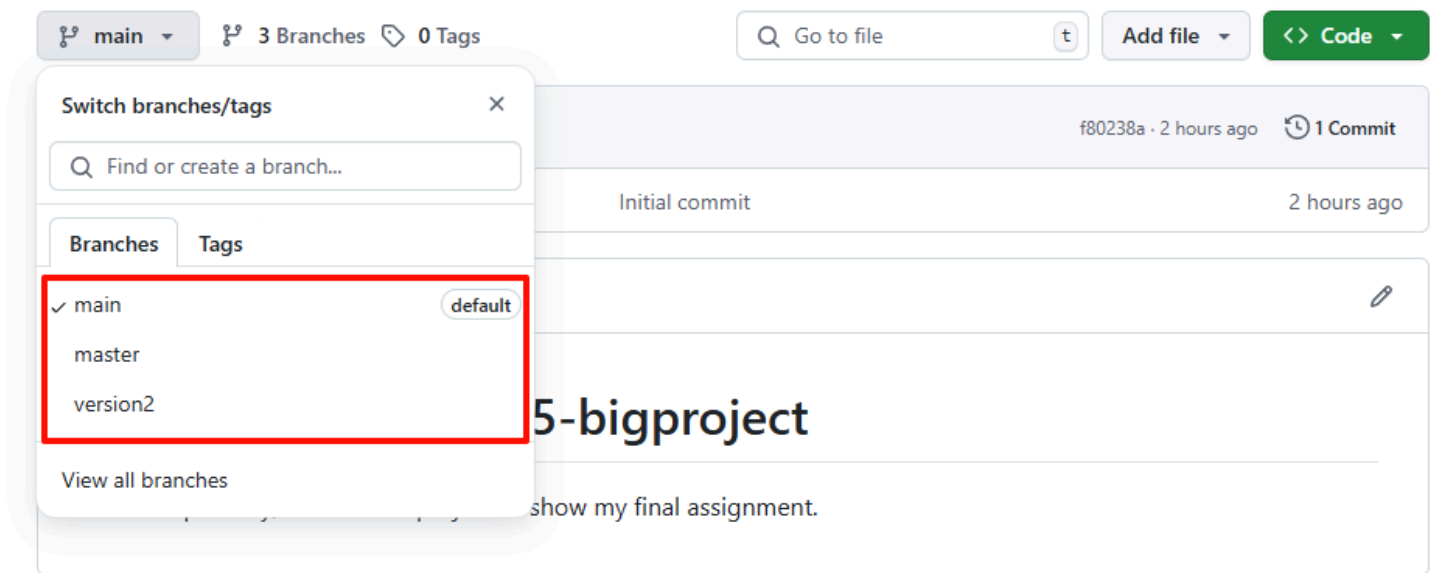
YI@DESKTOP-I5G39AI MINGW64 /d/其他文件/暑期AI训练营/aiSummerCamp2025/bigProject
(master)
$ git branch -a
* master
  remotes/origin/HEAD -> origin/main
  remotes/origin/main
  remotes/origin/master
```

创建新的分支，git branch 新分支名，但是这个新分支只存在于本地仓库，需要使用命令git push origin 新分支名，这样github上才会出现这个分支。

```
YI@DESKTOP-I5G39AI MINGW64 /d/其他文件/暑期AI训练营/aiSummerCamp2025/bigProject
(master)
$ git branch version2

YI@DESKTOP-I5G39AI MINGW64 /d/其他文件/暑期AI训练营/aiSummerCamp2025/bigProject
(master)
$ git branch -a
* master
  version2
  remotes/origin/HEAD -> origin/main
  remotes/origin/main
  remotes/origin/master

YI@DESKTOP-I5G39AI MINGW64 /d/其他文件/暑期AI训练营/aiSummerCamp2025/bigProject
(master)
$ git push origin version2
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'version2' on GitHub by visiting:
remote:   https://github.com/Joker-11k/aiSummerCamp2025-bigproject/pull/new/v
ersion2
remote:
To github.com:Joker-11k/aiSummerCamp2025-bigproject.git
 * [new branch]   version2 -> version2
```



切换分支 git switch 要切换到分支名

```
YI@DESKTOP-I5G39AI MINGW64 /d/其他文件/暑期AI训练营/aiSummerCamp2025/bigProject
(master)
$ git switch version2
Switched to branch 'version2'
```

github上有意思的库

1. hello world

网址：[leachim6/hello-world](#): Hello world in every computer language. Thanks to everyone who contributes to this, make sure to see contributing.md for contribution instructions!

学习很多语言都是从一个hello world！开始

2. scikit-learn

网址：[scikit-learn/scikit-learn](#): scikit-learn: machine learning in Python

3. 小游戏

2048网址：[gabrielecirulli/2048](#): The source code for 2048

贪吃蛇网址：[CodeExplainedRepo/Snake-JavaScript](#): The Snake game, created using JavaScript, and The HTML5 canvas.