GitHub及Git搭建和使用

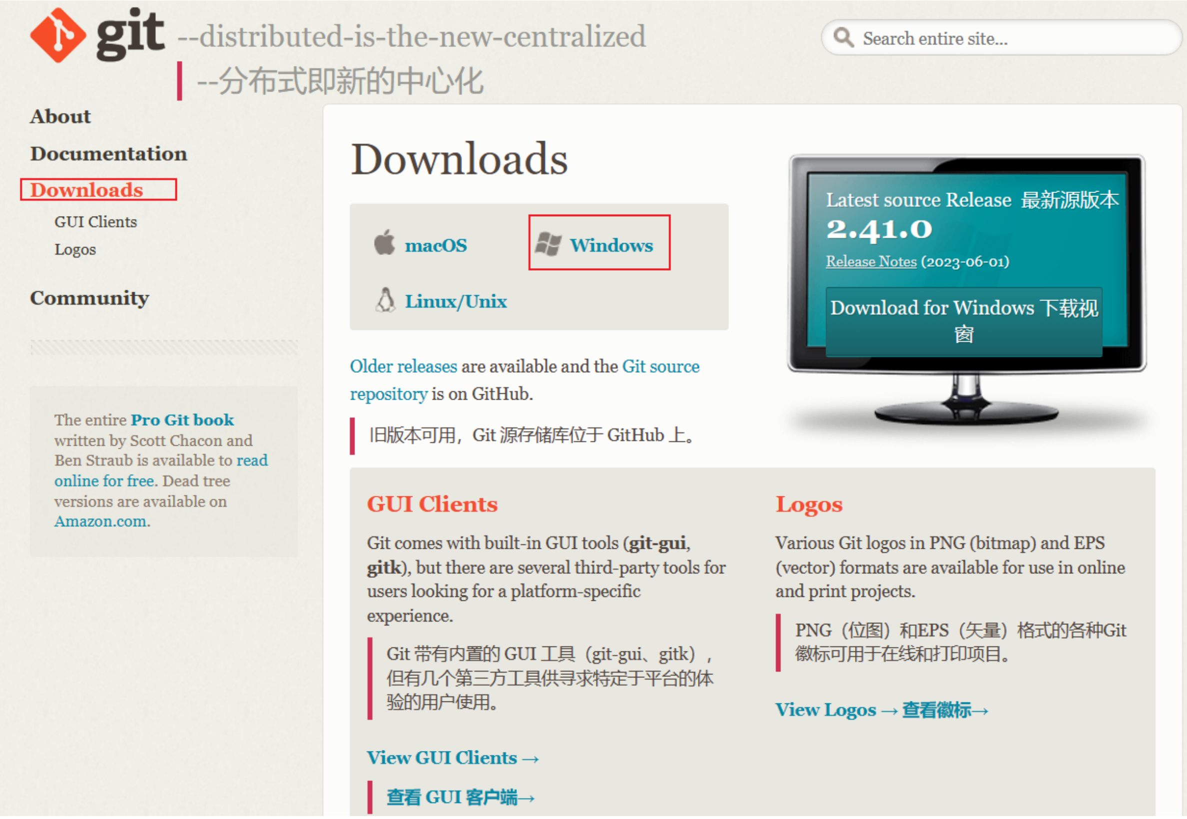
作者：郭毅鑫

1.注册GitHub账户

地址：https://github.com/

2.下载Git

地址：https://git-scm.com/downloads



The entire Pro Git book written by Scott Chacon and Ben Straub is available to read online for free. Dead tree versions are available on Amazon.com.

Older releases are available and the Git source repository is on GitHub.

旧版本可用，Git 源存储库位于 GitHub 上。

GUI Clients

Git comes with built-in GUI tools (**git-gui**, **gitk**), but there are several third-party tools for users looking for a platform-specific experience.

Git 带有内置的 GUI 工具（git-gui、gitk），但有几个第三方工具供寻求特定于平台的体验的用户使用。

View GUI Clients →

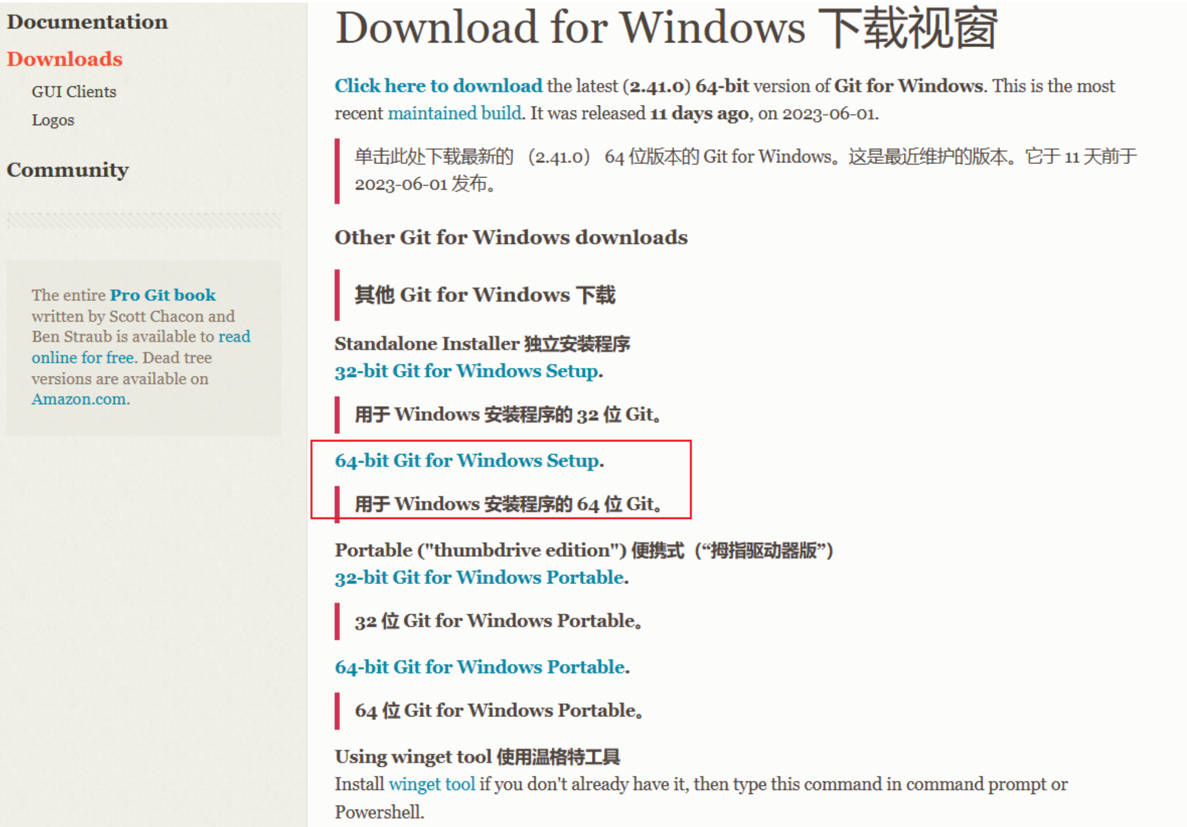
查看 GUI 客户端→

Logos

Various Git logos in PNG (bitmap) and EPS (vector) formats are available for use in online and print projects.

PNG（位图）和EPS（矢量）格式的各种Git徽标可用于在线和打印项目。

View Logos → 查看徽标→



Documentation

Downloads

GUI Clients

Logos

Community

The entire **Pro Git book** written by Scott Chacon and Ben Straub is available to read online for free. Dead tree versions are available on Amazon.com.

Download for Windows 下载视窗

Click here to download the latest (**2.41.0**) **64-bit** version of **Git for Windows**. This is the most recent maintained build. It was released **11 days ago**, on 2023-06-01.

> 单击此处下载最新的（2.41.0）64 位版本的 Git for Windows。这是最近维护的版本。它于 11 天前于 2023-06-01 发布。

Other Git for Windows downloads

> **其他 Git for Windows 下载**

Standalone Installer 独立安装程序
32-bit Git for Windows Setup.

> **用于 Windows 安装程序的 32 位 Git。**

64-bit Git for Windows Setup.

> **用于 Windows 安装程序的 64 位 Git。**

Portable ("thumbdrive edition") 便携式（"拇指驱动器版"）
32-bit Git for Windows Portable.

> **32 位 Git for Windows Portable。**

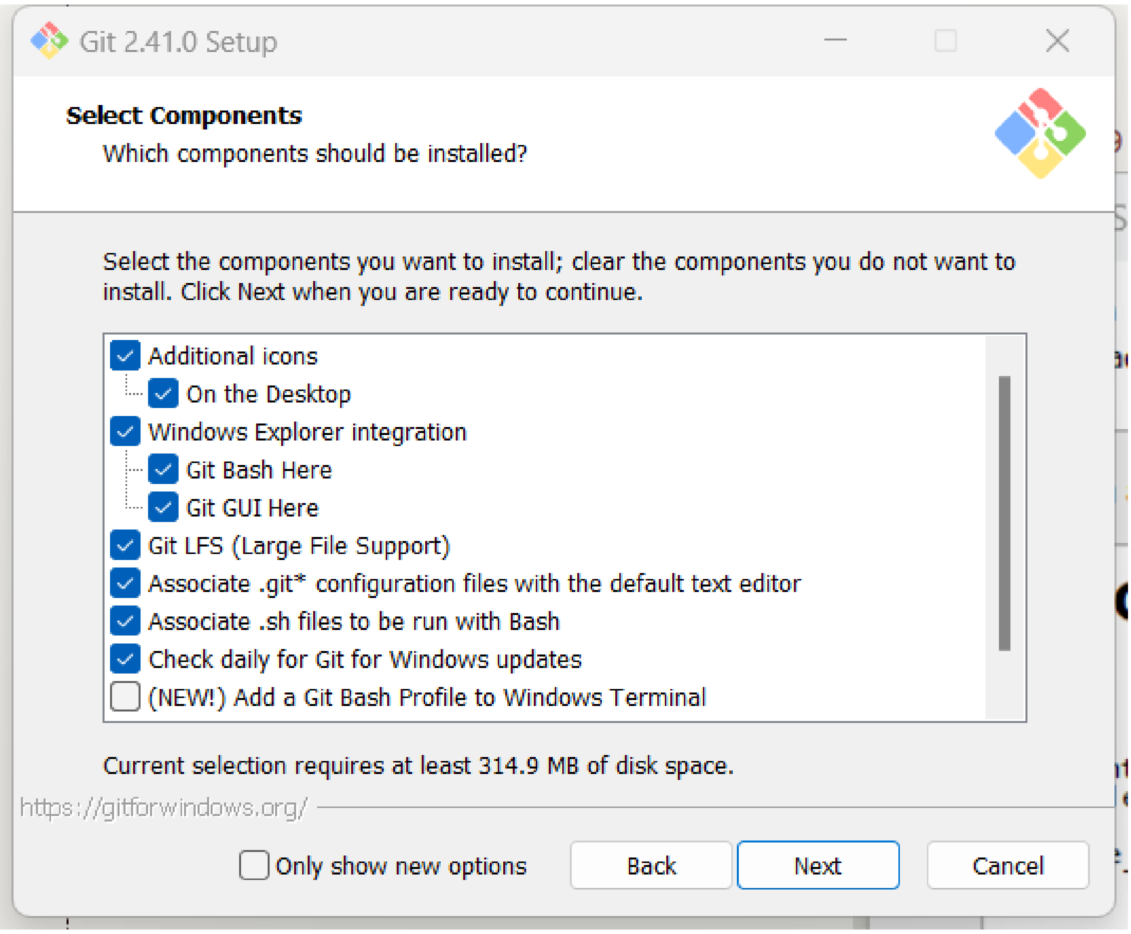
64-bit Git for Windows Portable.

> **64 位 Git for Windows Portable。**

Using winget tool 使用温格特工具
Install winget tool if you don't already have it, then type this command in command prompt or Powershell.

3.安装Git

1. **基本上都是默认配置，以下是需要关注得地方**
2. 安装详细地址：[Git安装教程（windows） - 战争热诚 - 博客园 (cnblogs.com)]
3. 安装配置文件，自己需要的都选上



Select Components

Which components should be installed?

Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.

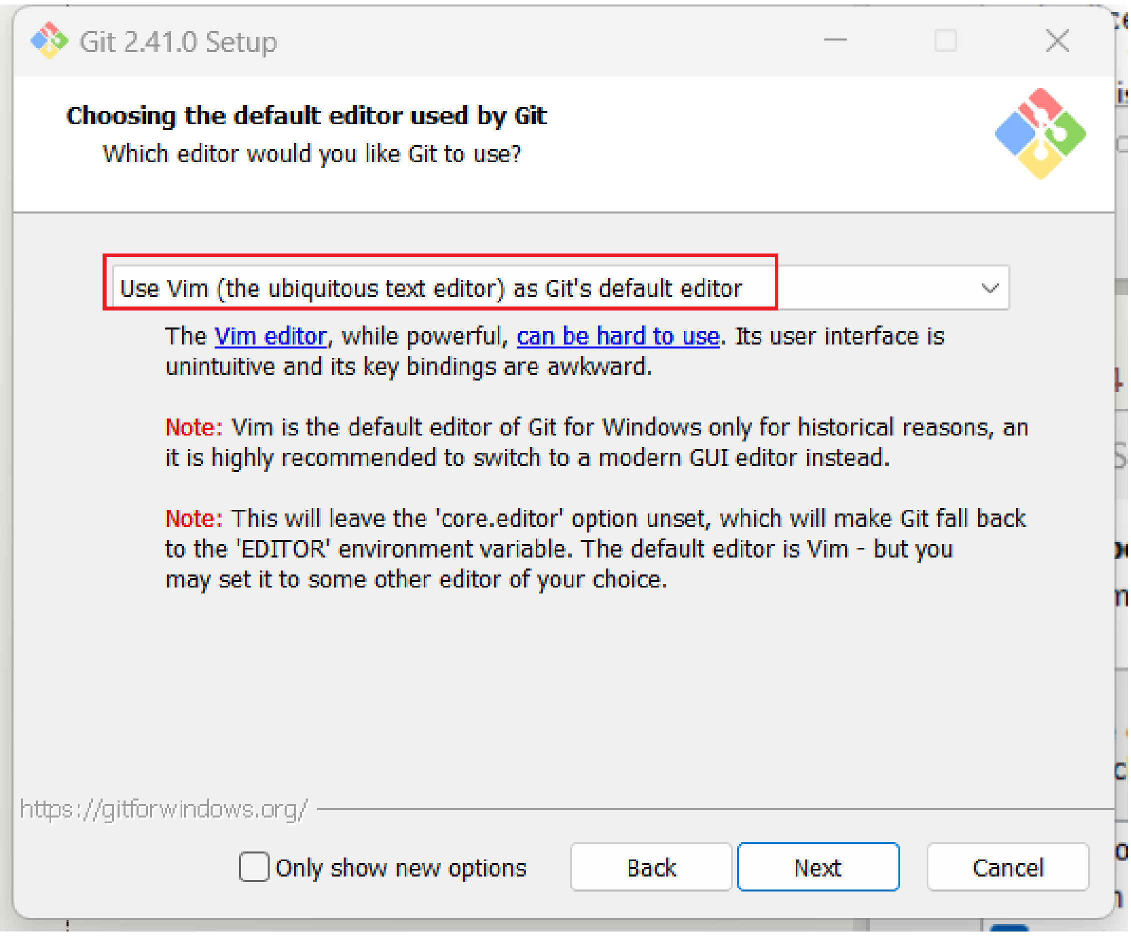
- ☑ Additional icons
  - ☑ On the Desktop
- ☑ Windows Explorer integration
  - ☑ Git Bash Here
  - ☑ Git GUI Here
- ☑ Git LFS (Large File Support)
- ☑ Associate .git* configuration files with the default text editor
- ☑ Associate .sh files to be run with Bash
- ☑ Check daily for Git for Windows updates
- ☐ (NEW!) Add a Git Bash Profile to Windows Terminal

Current selection requires at least 314.9 MB of disk space.

https://gitforwindows.org/

☐ Only show new options

4.


Git 2.41.0 Setup

Choosing the default editor used by Git

Which editor would you like Git to use?

Use Vim (the ubiquitous text editor) as Git's default editor

The Vim editor, while powerful, can be hard to use. Its user interface is unintuitive and its key bindings are awkward.

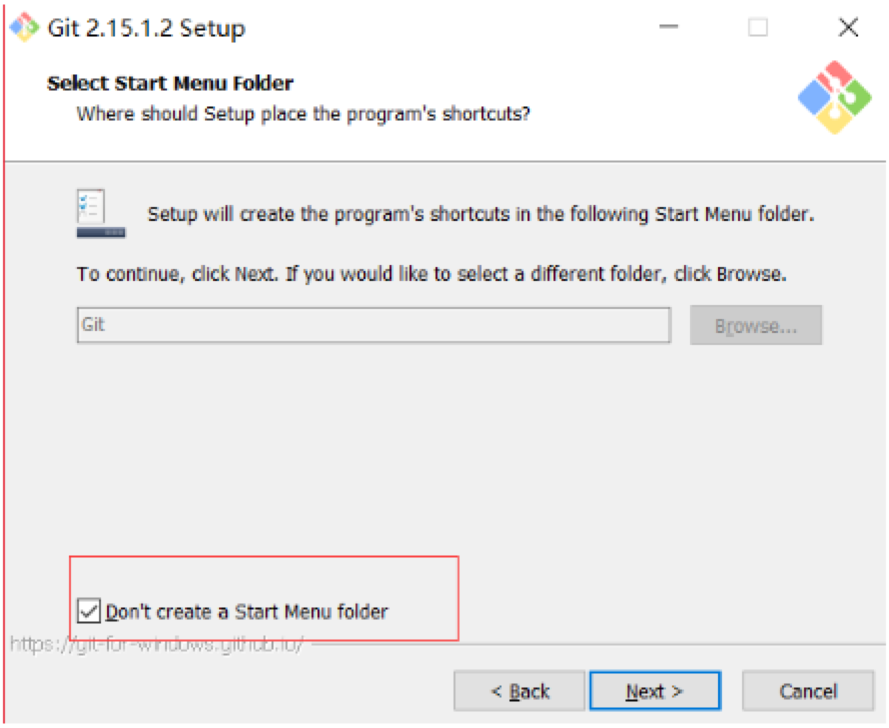
Note: Vim is the default editor of Git for Windows only for historical reasons, an it is highly recommended to switch to a modern GUI editor instead.

Note: This will leave the 'core.editor' option unset, which will make Git fall back to the 'EDITOR' environment variable. The default editor is Vim - but you may set it to some other editor of your choice.

https://gitforwindows.org/

□ Only show new options

5. 不创建启动文件夹


Git 2.15.1.2 Setup

Select Start Menu Folder

Where should Setup place the program's shortcuts?

Setup will create the program's shortcuts in the following Start Menu folder.

To continue, click Next. If you would like to select a different folder, click Browse.

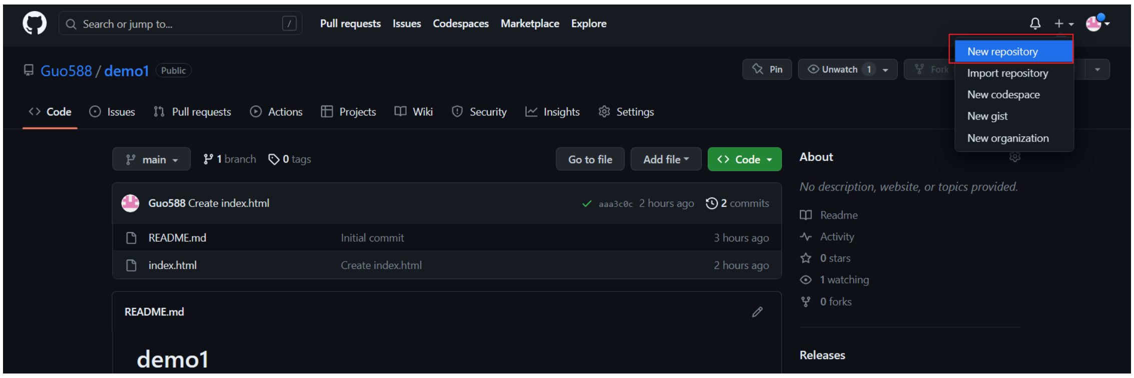
Git

☑ Don't create a Start Menu folder

https://git-for-windows.github.io/

4.创建GitHub仓库

1. 新建存储库



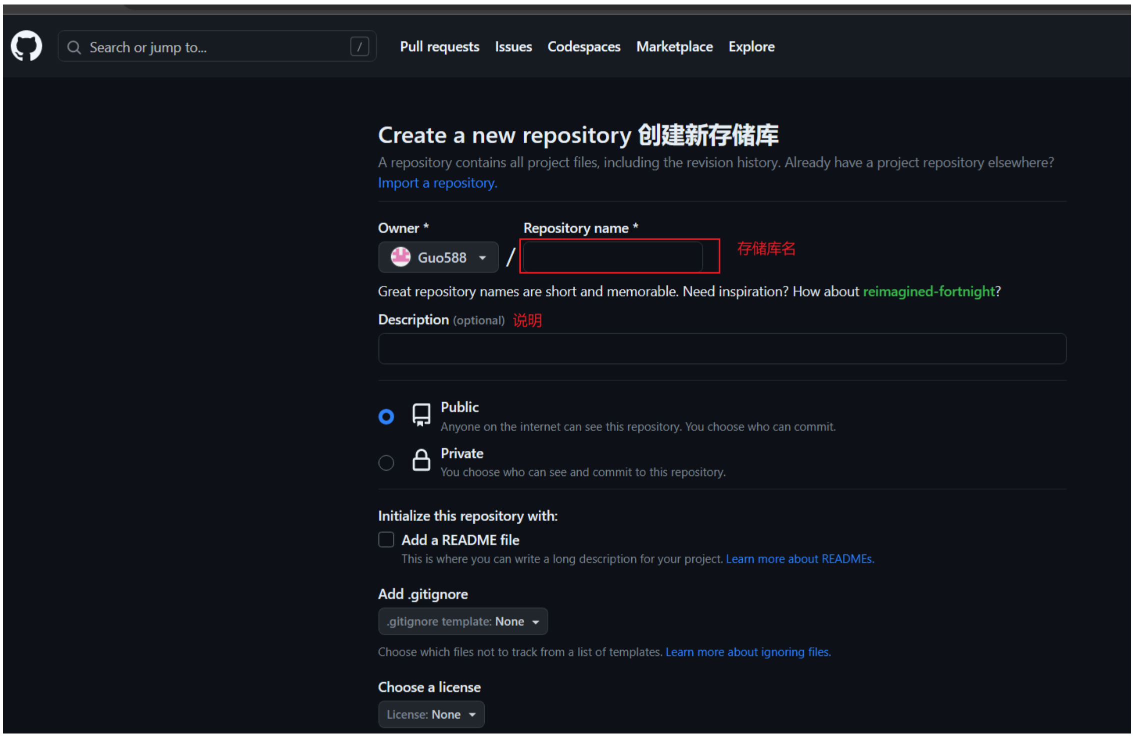
No description, website, or topics provided.

Guo588 Create index.html	aaa3c0c 2 hours ago	2 commits
README.md	Initial commit	3 hours ago
index.html	Create index.html	2 hours ago

README.md

demo1

2. 填写信息(需要部署简易网站时可以填写：用户名+.github.io;例如：Guo588.github.io；具体配置请看第6小节)，后点击提交创建



Create a new repository 创建新存储库

A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.

Owner * Guo588 / Repository name * 存储库名

Great repository names are short and memorable. Need inspiration? How about reimagined-fortnight?

Description (optional) 说明

● 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.

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

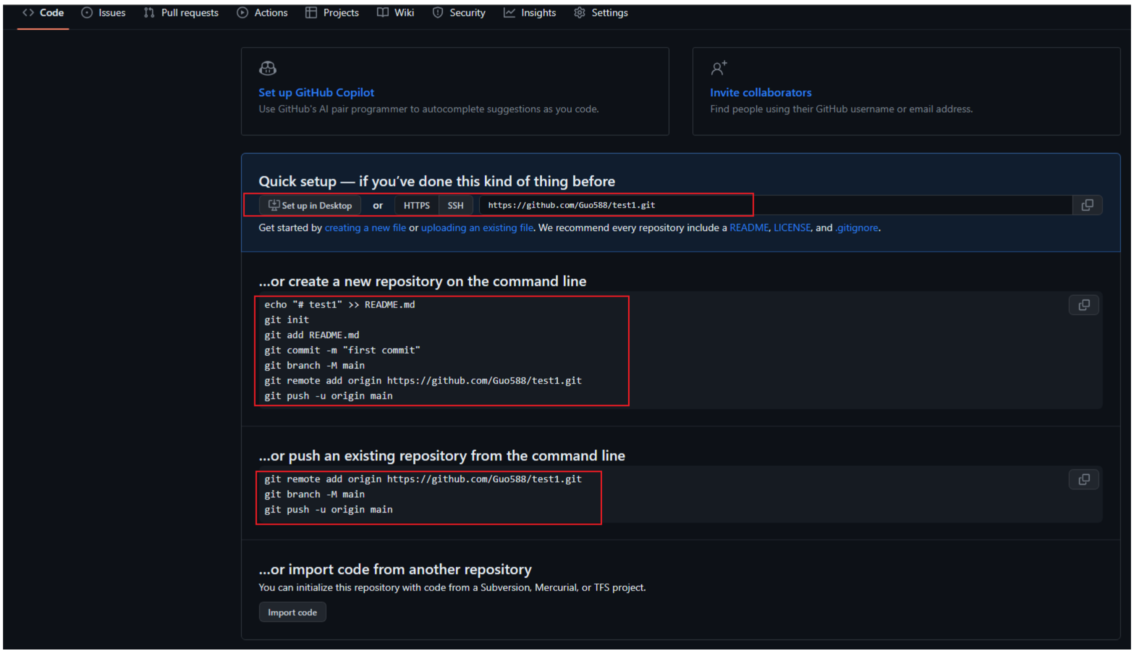
.gitignore template: None

Choose which files not to track from a list of templates. Learn more about ignoring files.

Choose a license

License: None

3. 查看配置



Set up GitHub Copilot
Use GitHub's AI pair programmer to autocomplete suggestions as you code.

Invite collaborators
Find people using their GitHub username or email address.

Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH https://github.com/Guo588/test1.git

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

...or create a new repository on the command line

```
echo "# test1" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/Guo588/test1.git
git push -u origin main
```

...or push an existing repository from the command line

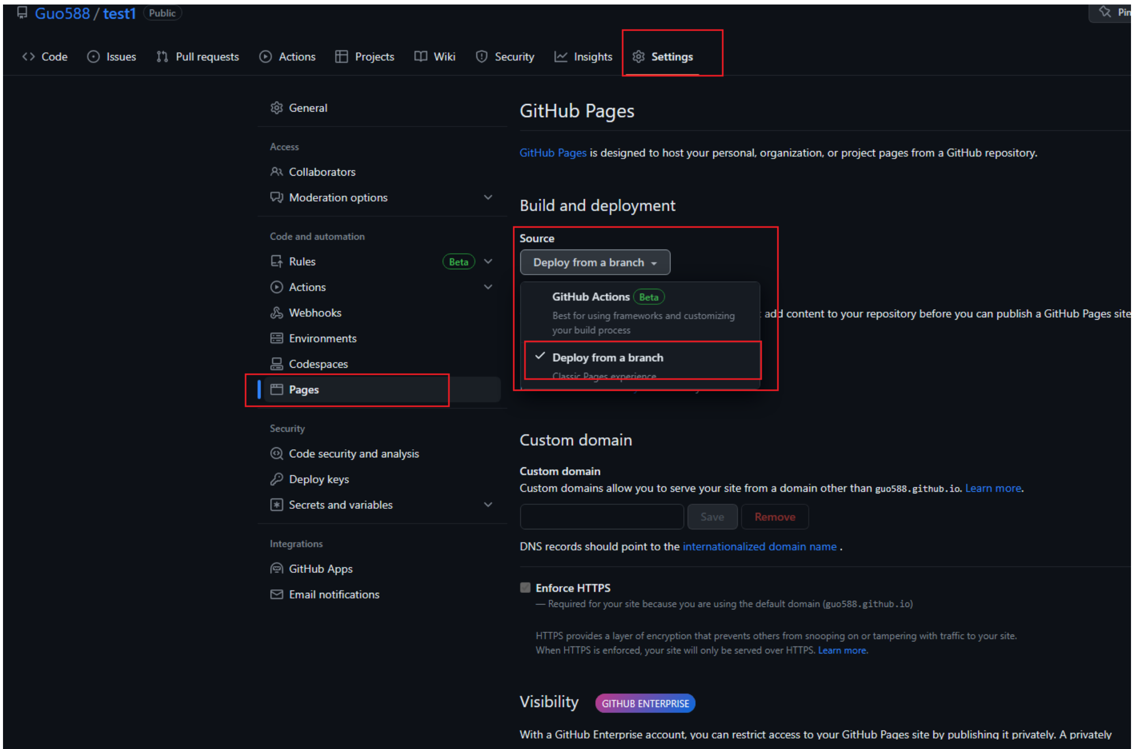
```
git remote add origin https://github.com/Guo588/test1.git
git branch -M main
git push -u origin main
```

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

Import code

4. 设置分支，然后点击save保存



GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Build and deployment

Source

Deploy from a branch

GitHub Actions Beta
Best for using frameworks and customizing your build process

✓ Deploy from a branch
Classic Pages experience

add content to your repository before you can publish a GitHub Pages site

Custom domain

Custom domain
Custom domains allow you to serve your site from a domain other than guo588.github.io. Learn more.

Save Remove

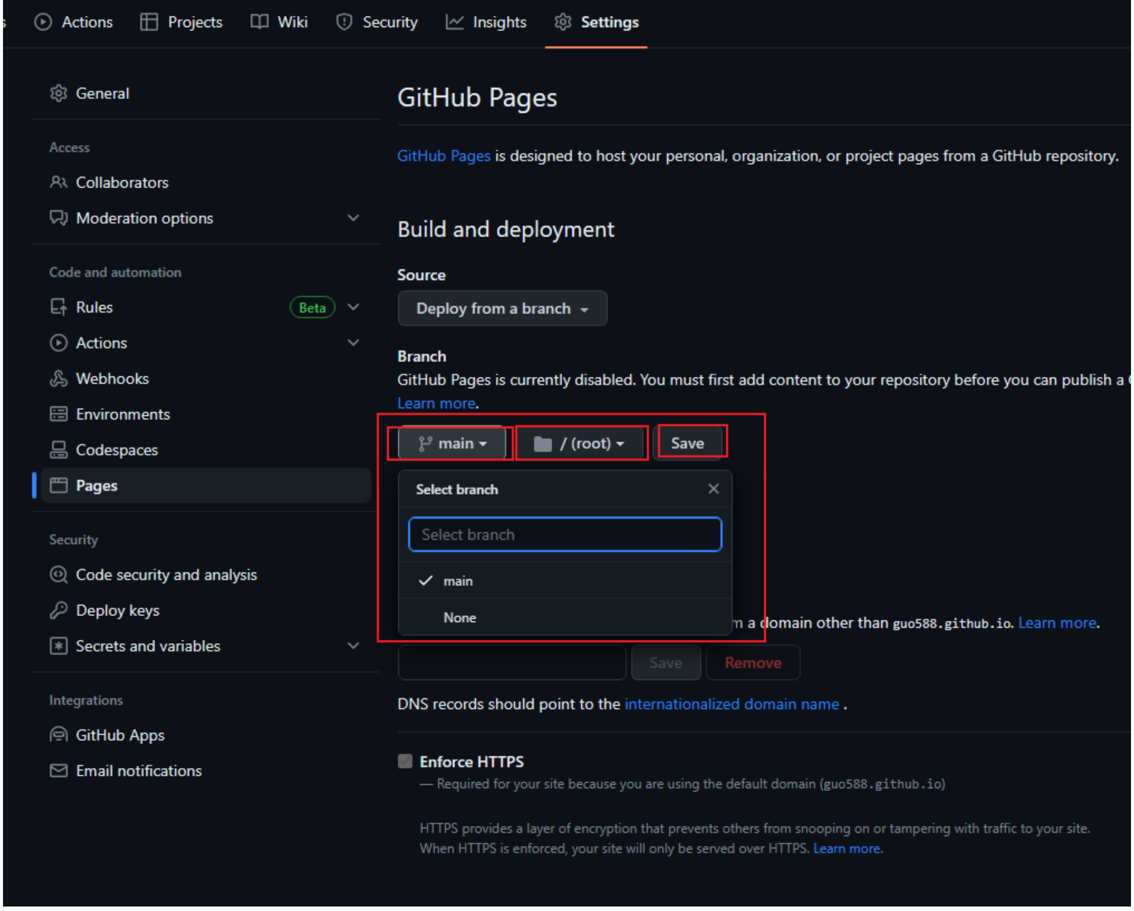
DNS records should point to the internationalized domain name .

□ Enforce HTTPS
— Required for your site because you are using the default domain (guo588.github.io)

HTTPS provides a layer of encryption that prevents others from snooping on or tampering with traffic to your site.
When HTTPS is enforced, your site will only be served over HTTPS. Learn more.

Visibility GITHUB ENTERPRISE

With a GitHub Enterprise account, you can restrict access to your GitHub Pages site by publishing it privately. A privately



GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Build and deployment

Source

Deploy from a branch

Branch

GitHub Pages is currently disabled. You must first add content to your repository before you can publish a
Learn more.

main / (root) Save

Select branch

Select branch

✓ main

None

a domain other than guo588.github.io. Learn more.

Save Remove

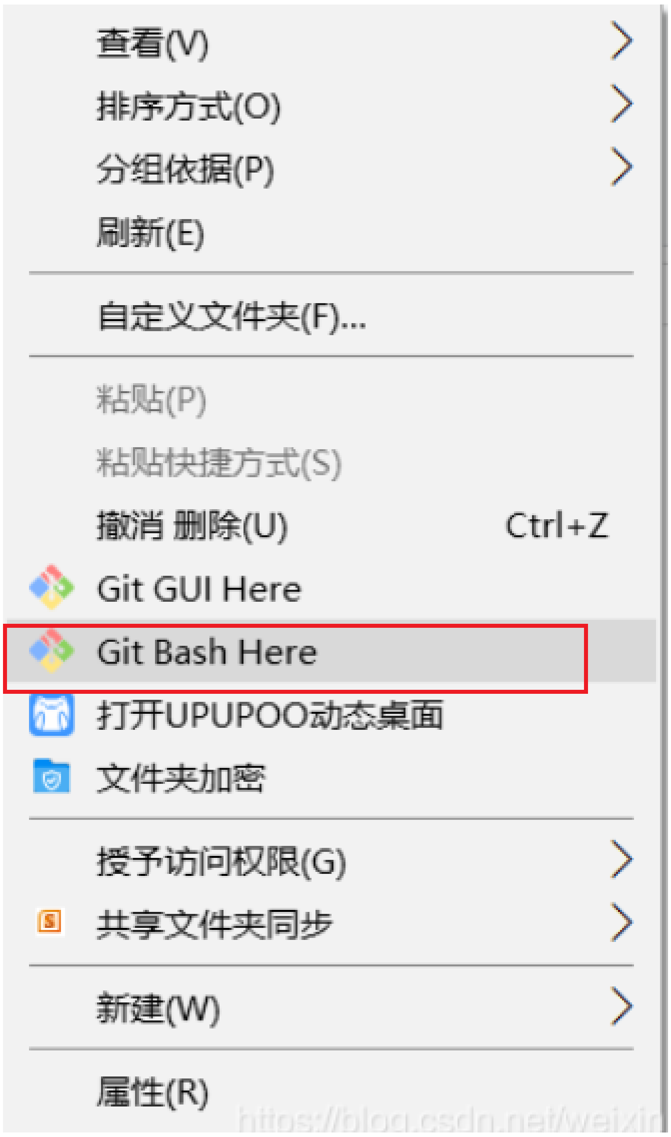
DNS records should point to the internationalized domain name .

□ Enforce HTTPS
— Required for your site because you are using the default domain (guo588.github.io)

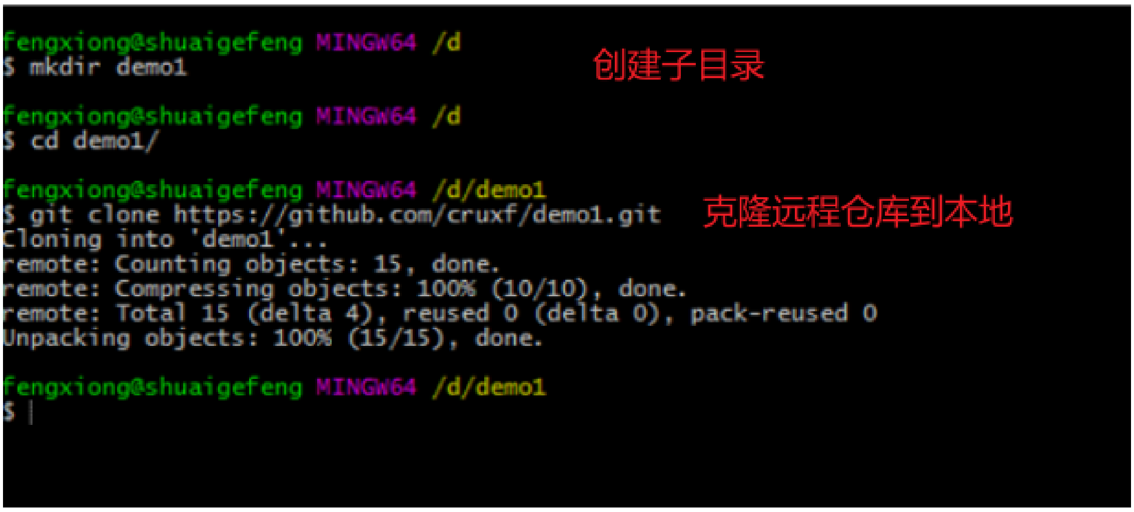
HTTPS provides a layer of encryption that prevents others from snooping on or tampering with traffic to your site.
When HTTPS is enforced, your site will only be served over HTTPS. Learn more.

5.使用Git，让本地上传到GitHub存储库

1. 打开此电脑，选择一个盘，比如 D 盘，右键空白处点击 git bash here（前提是git已经安装好）。

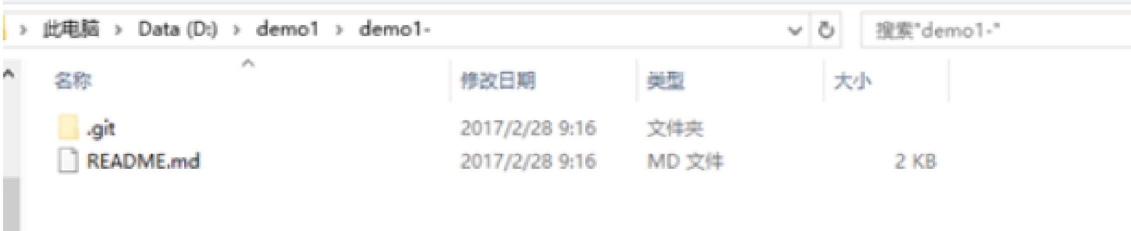


2. 输入如下命令，用来在 D 盘创建 demo1 文件放你的github上的demo1 repository，克隆test repository到 demo1 文件中。



```
fengxiong@shuaigefeng MINGW64 /d
$ mkdir demo1

fengxiong@shuaigefeng MINGW64 /d
$ cd demo1/

fengxiong@shuaigefeng MINGW64 /d/demo1
$ git clone https://github.com/cruxf/demo1.git
Cloning into 'demo1'...
remote: Counting objects: 15, done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 15 (delta 4), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (15/15), done.

fengxiong@shuaigefeng MINGW64 /d/demo1
$ |
```

3. 生成文件夹

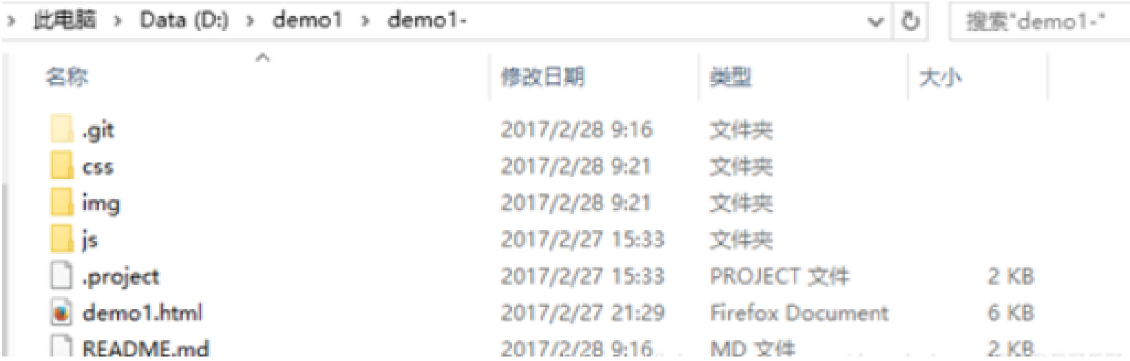


› 此电脑 › Data (D:) › demo1 › demo1-

搜索"demo1-"

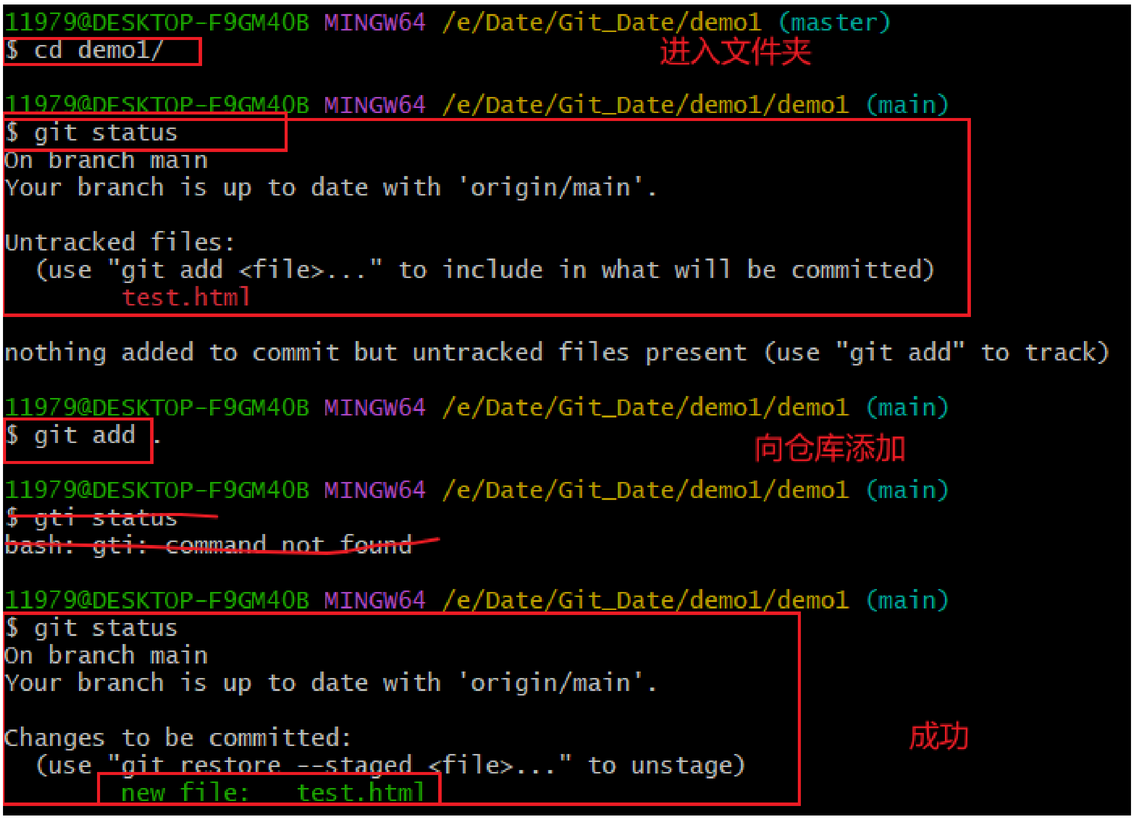
名称	修改日期	类型	大小
.git	2017/2/28 9:16	文件夹	
README.md	2017/2/28 9:16	MD 文件	2 KB

4. 将自己的网页文件复制粘贴至 D 盘的 demo1- 文件中



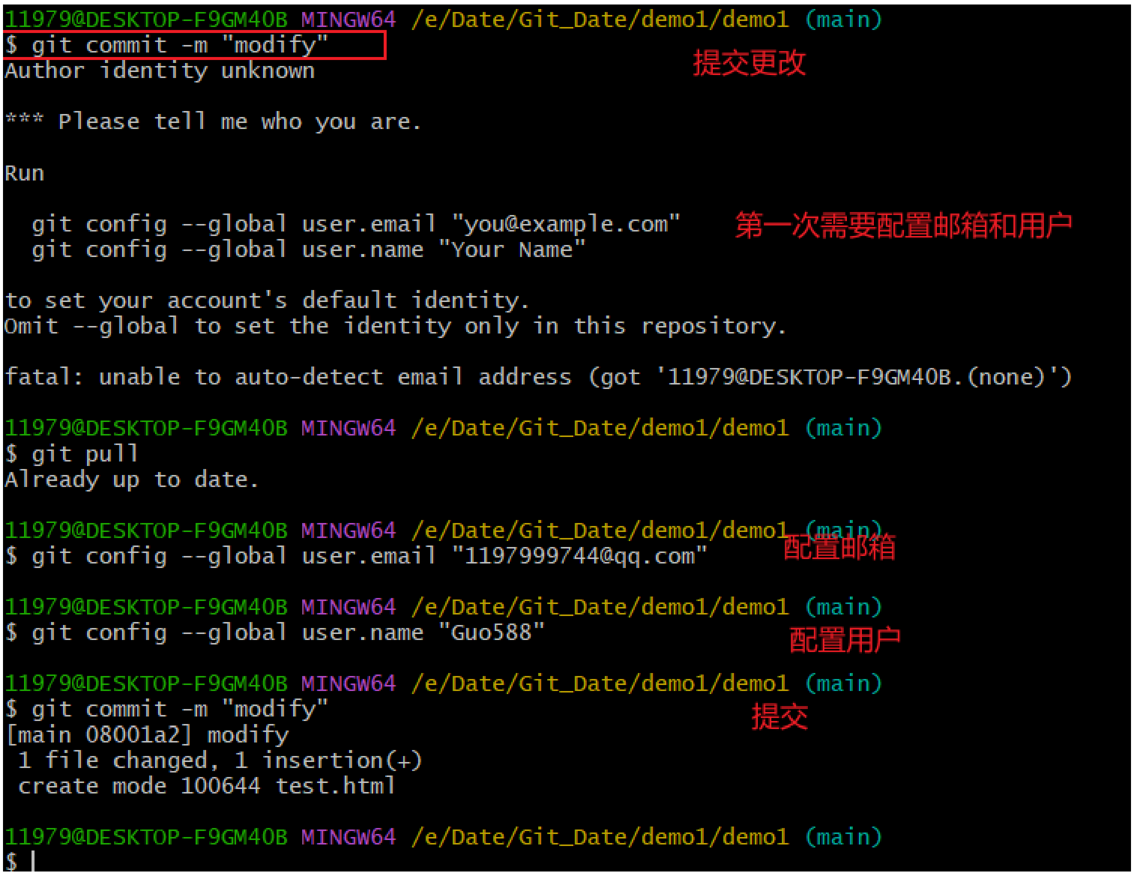
名称	修改日期	类型	大小
.git	2017/2/28 9:16	文件夹	
css	2017/2/28 9:21	文件夹	
img	2017/2/28 9:21	文件夹	
js	2017/2/27 15:33	文件夹	
.project	2017/2/27 15:33	PROJECT 文件	2 KB
demo1.html	2017/2/27 21:29	Firefox Document	6 KB
README.md	2017/2/28 9:16	MD 文件	2 KB

5. 执行如下命令



```
11979@DESKTOP-F9GM40B MINGW64 /e/Date/Git_Date/demo1 (master)
$ cd demo1/

11979@DESKTOP-F9GM40B MINGW64 /e/Date/Git_Date/demo1/demo1 (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        test.html

nothing added to commit but untracked files present (use "git add" to track)

11979@DESKTOP-F9GM40B MINGW64 /e/Date/Git_Date/demo1/demo1 (main)
$ git add .

11979@DESKTOP-F9GM40B MINGW64 /e/Date/Git_Date/demo1/demo1 (main)
~~$ gti status~~
~~bash: gti: command not found~~

11979@DESKTOP-F9GM40B MINGW64 /e/Date/Git_Date/demo1/demo1 (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   test.html
```

6. 提交更改



```
11979@DESKTOP-F9GM40B MINGW64 /e/Date/Git_Date/demo1/demo1 (main)
$ git commit -m "modify"
Author identity unknown

*** Please tell me who you are.

Run

  git config --global user.email "you@example.com"
  git config --global user.name "Your Name"

to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got '11979@DESKTOP-F9GM40B.(none)')

11979@DESKTOP-F9GM40B MINGW64 /e/Date/Git_Date/demo1/demo1 (main)
$ git pull
Already up to date.

11979@DESKTOP-F9GM40B MINGW64 /e/Date/Git_Date/demo1/demo1 (main)
$ git config --global user.email "1197999744@qq.com"

11979@DESKTOP-F9GM40B MINGW64 /e/Date/Git_Date/demo1/demo1 (main)
$ git config --global user.name "Guo588"

11979@DESKTOP-F9GM40B MINGW64 /e/Date/Git_Date/demo1/demo1 (main)
$ git commit -m "modify"
[main 08001a2] modify
 1 file changed, 1 insertion(+)
 create mode 100644 test.html

11979@DESKTOP-F9GM40B MINGW64 /e/Date/Git_Date/demo1/demo1 (main)
$ |
```

提交更改

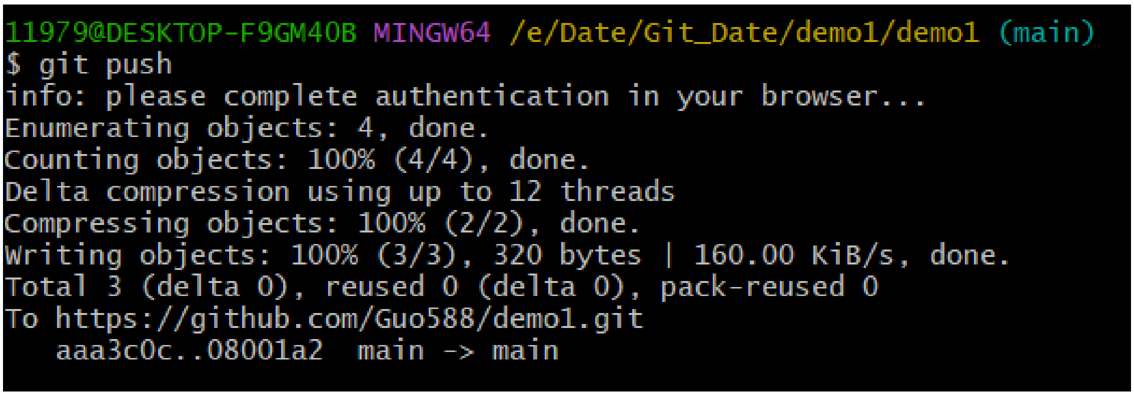
第一次需要配置邮箱和用户

配置邮箱

配置用户

提交

7. 输入git push来发布本地提交



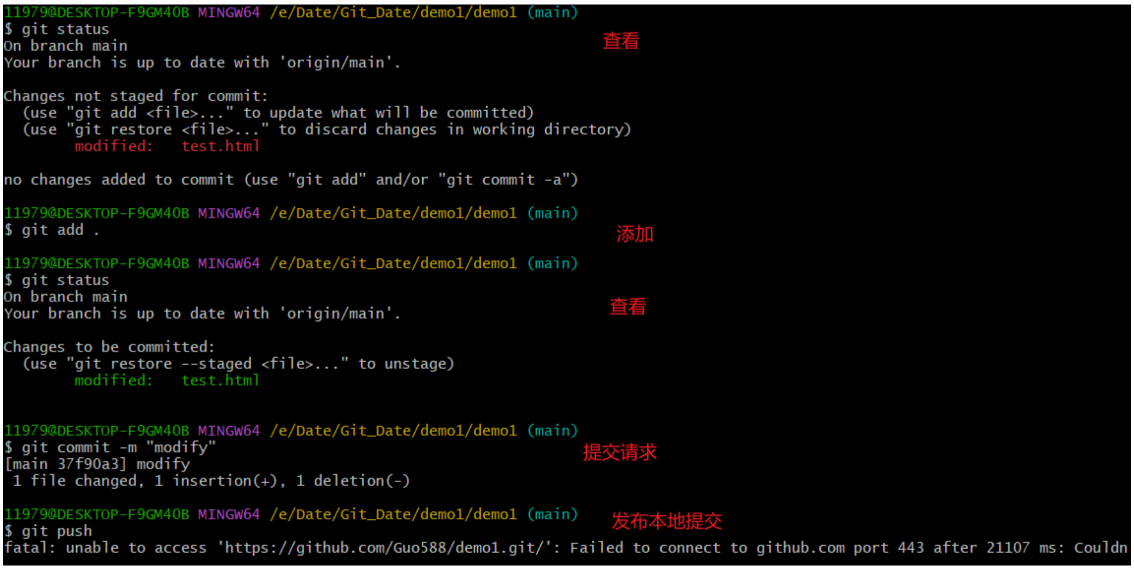
```
11979@DESKTOP-F9GM40B MINGW64 /e/Date/Git_Date/demo1/demo1 (main)
$ git push
info: please complete authentication in your browser...
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 320 bytes | 160.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Guo588/demo1.git
   aaa3c0c..08001a2  main -> main
```

会弹出确定框，确认就好

6.部署好后的使用流程

1. 将项目放进本地克隆仓库中，具体位置查看第5.4小节

2. 使用流程

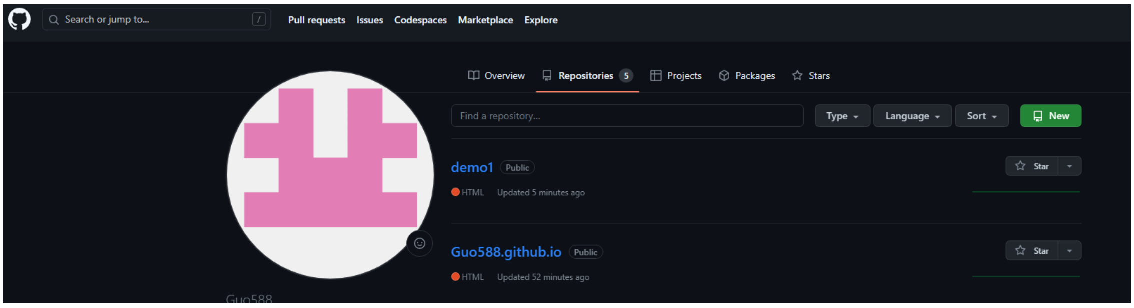


```
11979@DESKTOP-F9GM40B MINGW64 /e/Date/Git_Date/demo1/demo1 (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   test.html

no changes added to commit (use "git add" and/or "git commit -a")

11979@DESKTOP-F9GM40B MINGW64 /e/Date/Git_Date/demo1/demo1 (main)
$ git add .

11979@DESKTOP-F9GM40B MINGW64 /e/Date/Git_Date/demo1/demo1 (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   test.html


11979@DESKTOP-F9GM40B MINGW64 /e/Date/Git_Date/demo1/demo1 (main)
$ git commit -m "modify"
[main 37f90a3] modify
 1 file changed, 1 insertion(+), 1 deletion(-)

11979@DESKTOP-F9GM40B MINGW64 /e/Date/Git_Date/demo1/demo1 (main)
$ git push
fatal: unable to access 'https://github.com/Guo588/demo1.git/': Failed to connect to github.com port 443 after 21107 ms: Couldn
```

上述最后一步是连接异常，多尝试几次提交就可以

7.网站部署理解

- 创建存储库（如果是以用户为名创建的库，是为默认网址；其创建方式查看第4.2小节）
- 非用户名创建的库，需要在访问路径上加库的名
- 示例：



- 以上为第二个为非用户名创建的库
- **默认访问地址为**

  ```
  https://guo588.github.io
  ```

  另一个库则为

  ```
  https://guo588.github.io/demo1
  ```

  当一个库内部有多个页面，而你也需要进入**不同页面时可以添加上指定路径**

  ```
  https://guo588.github.io/index.html
  ```

  或另一个页面

  ```
  https://guo588.github.io/index1.html
  ```

- **非用户的库则需要再加上库的名字**

```
https://guo588.github.io/demo1/index.html
```

8.Git中文件名乱码问题

1. 设置Git全局配置中的文件名编码

可以通过以下命令来设置Git全局配置中的文件名编码为UTF-8：

```
$ git config --global core.quotepath false
$ git config --global gui.encoding utf-8
$ git config --global i18n.commit.encoding utf-8
$ git config --global i18n.logoutputencoding utf-8
```