

github在mac上的安装和使用

一、首先打开终端(Terminal),输入” git” 命令,弹出github的安装窗口选择安装,单击” 继续”：



命令行工具许可协议

APPLE INC.
MAC SDK及XCODE协议

使用 开发者软件（定义如下）之前，请仔细阅读本MAC SDK及XCODE协议（“许可证”）。阁下使用 开发者软件，即表示同意接受本许可证的条款约束。如阁下是以电子方式取用开发者软件，请点击“同意”键，表明阁下同意接受本许可证的条款约束。如阁下不同意本许可证的条款，请勿使用开发者软件并点击“不同意”键。

重要提示：在本软件可用于复制材料的范围内，阁下获许可只用以复制没有著作权的材料、阁下拥有其著作权的材料、或阁下获授权或法律允许复制的材料。如果阁下不能肯定阁下复制任何材料的权利，应联系阁下的法律顾问。

1. 一般规定。

A. 本许可证所附的Apple软件、工具、实用程序、样品或示例代码、文件、界面、内容、字体、数据及其它材料，不论是储存于磁盘、打印或电子文件、只读存储器、或任何其它载体或属任何其它形式（统称“开发者软件”），都是由 Apple Inc.（“Apple”）许可阁下使用而非售予阁下，而且阁下只可根据本许可证的条款加以使用，Apple和/或 Apple 的许可人保留对开发者软件本身的所有权，以及保留一切并未明确授予阁下的权利。本许可证的条款将监管由 Apple 提供以取代及/或补充原先开发者软件的任何软件升级版本，除非该升级版本另外附有独立许可证，此情况下则以该许可证的条款管辖。

B. 通过 开发者软件显示或取用的任何内容，其所有权和知识产权属于相关内容的版权所有人。此等内容可能受著作权或其它知识产权法律和条约保护，及可能须遵守提供该等内容的第三方的使用条款。本许可证并不授予阁下使用该等内容的任何权利，亦不保证该等内容将继续提供给阁下。

2. 允许的许可使用方式和限制。

A. 许可证。在不违反本许可证条款和条件的情况下，阁下获授予有限的非独家许可证，可在多台Apple品牌计算机上使用开发者软件，用以开发和测试应用程序及其它软件。阁下仅可制作合理地必要的若干份开发者软件内部使用副本，以按照本许可证允许用途使用开发者软件，并仅向其职责范围要求如此使用开发者软件的阁下的员工分发该等副本；惟阁下须在每份开发者软件或其部分的副本上，复制原件包含的一切著作权或其它专有权通知。

B. 其他使用限制。本许可证列明的授权不允许阁下、且阁下亦同意不在任何非Apple品牌计算机上安装、使用或运行开发者软件，或让别人这样做。除本许可证条款另行明确允许或Apple另行许可外：(i) 任何时候只可由一个用户使用开发者软件，以及(ii) 阁下不得在网络上提供开发者软件，以致开发者软件在同一时间可供多台计算机运行或使用。阁下不得出租、租赁、出借、出售、再许可或以其它方式再分发开发者软件，或以任何未经授权的方式利用开发者软件所提供或通过开发者软件得到的任何服务。

C. 不得倒序制造：限制事项。阁下不得且阁下同意不会或让其他人复制（本许可证明示允许除外）、反汇编、倒序制造、拆装、企图导出其源代码、解码、修改开发者软件或制造其衍生作品、或开发者软件提供的或通过开发者软件提供的任何服务或其任何部分（除非或在适用法律禁止前述限制的范围内，或在监管开放源代码或样品守则的使用的许可证条款予以允许的范围内）。

D. 样品守则。开发者软件的某些部分包含由Apple提供的样品或示例代码（“示例代码”）。阁下只可按照附随该等示例代码或

不同意

同意

二、安装完成后再次输入” git” 命令，跳出如下命令提示等内容表示github已经安装成功了；

generated by haroopad

```

huhuhudeMini:~ huhuhu$ git
usage: git [--version] [--help] [-C <path>] [-c name=value]
         [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
         [-p | --paginate | --no-pager] [--no-replace-objects] [--bare]
         [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
         <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
    reset      Reset current HEAD to the specified state
    rm         Remove files from the working tree and from the index


examine the history and state (see also: git help revisions)
    bisect     Find by binary search the change that introduced a bug
    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
    branch     List, create, or delete branches
    checkout   Switch branches or restore working tree files
    commit     Record changes to the repository
    diff       Show changes between commits, commit and working tree, etc
    merge      Join two or more development histories together
    rebase     Forward-port local commits to the updated upstream head
    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.

```

三、然后需要设置如何上传文件或文件夹:

1. 首先创建一个专属存放上传github文件的新文件夹, 在终端输入命令:” mkdir git (新建文件夹名称)”, 然后输入” cd git”, 打开刚刚新建的文件夹, 再次输入” pwd /Users/huhuhu/git (文件夹所在路径)” 显示该文件夹当前的目录;

```

huhuhudeMac-mini:~ huhuhu$ mkdir git
huhuhudeMac-mini:~ huhuhu$ cd git
huhuhudeMac-mini:git huhuhu$ pwd /Users/huhuhu/git
/Users/huhuhu/git
huhuhudeMac-mini:git huhuhu$ █

```

- 由于mac的系统权限限制, 发现之前创建的新文件夹无法被查看, 输入” cd” 退出当前文件夹, 再输入” ls” 查看目前的主文件夹目录, 发现新建的” git” 文件没有建立在” Desktop” 文件夹中(在mac系统中, 由于权限原因所以只能查看到此文件夹中的内容), 然后输入” sudo rm -r -f git(文件夹名称)” 进行删除文件夹操作, 输入密码后再次输入” ls” 命令, 看到刚才新建的” git” 文件夹已经被删除,

```

huhuhudeMac-mini:git huhuhu$ ls
huhuhudeMac-mini:git huhuhu$ cd
huhuhudeMac-mini:~ huhuhu$ ls
Applications                               Music
Desktop                                   Pictures
Documents                                 Public
Downloads                               VirtualBox VMs
Library                                  git
Movies                                linux虚拟机安装 nodejs.mdown
huhuhudeMac-mini:~ huhuhu$ sudo rm -r -f git
Password:
huhuhudeMac-mini:~ huhuhu$ ls
Applications                               Music
Desktop                                   Pictures
Documents                                 Public
Downloads                               VirtualBox VMs
Library                                  linux虚拟机安装 nodejs.mdown
Movies

```

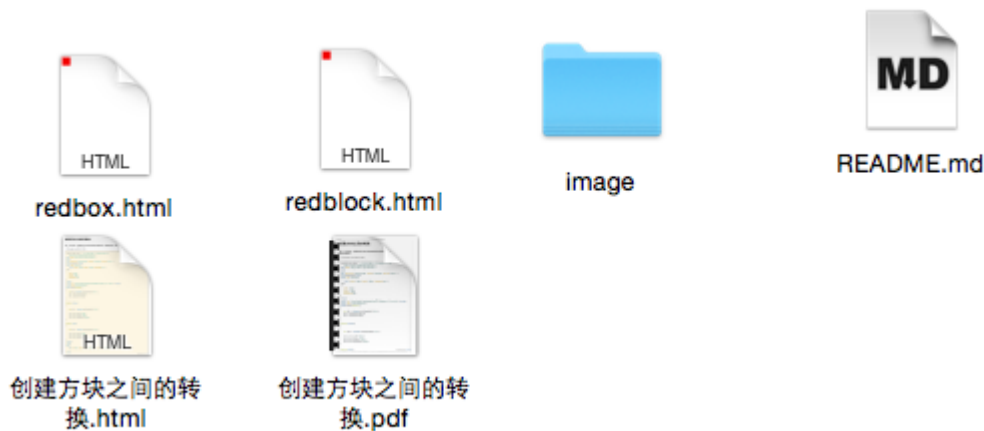
- 接下来使用” cd +文件夹路径” 进入想要新建文件夹得位置，在重复上述新建文件夹的步骤，新建出” git” 文件夹，
 - 在新建文件夹的过程中因为路径输入不正确(不完整)，所以出现了” -bash: cd: sublime文件: No such file or directory” (路径不存在或是相对路径而不是绝对路径)的错误提示，我采用的方法是逐步输入路径，首先输入” ls” ,然后” cd +Desкто” 进入” Desktop文件夹”,在使用同样的方法进入下一层文件夹路径，直到进入自己新建的” git” 文件夹内，
 - 在建立新的文件夹时，由于新建文件夹名时再次出现” mkdir: github: File exists” (检查文件或目录是否存在)错误,提示我该文件夹名重复，于是重新更改文件夹名称即可解决问题；
- 和之前使用相同的方法进入新建的文件夹” git”

```

huhuhudeMac-mini:~ huhuhu$ cd sublime文件
-bash: cd: sublime文件: No such file or directory
huhuhudeMac-mini:~ huhuhu$ cd /桌面/sublime文件
-bash: cd: /桌面/sublime文件: No such file or directory
huhuhudeMac-mini:~ huhuhu$ cd sublime文件
-bash: cd: sublime文件: No such file or directory
huhuhudeMac-mini:~ huhuhu$ ls
Applications                               Music
Desktop                                   Pictures
Documents                                 Public
Downloads                               VirtualBox VMs
Library                                  linux虚拟机安装 nodejs.mdown
Movies
huhuhudeMac-mini:~ huhuhu$ cd Desktop
huhuhudeMac-mini:Desktop huhuhu$ cd sublime文件
huhuhudeMac-mini:sublime文件 huhuhu$ mkdir github
mkdir: github: File exists
huhuhudeMac-mini:sublime文件 huhuhu$ mkdir git
huhuhudeMac-mini:sublime文件 huhuhu$ cd git
huhuhudeMac-mini:git huhuhu$ pwd /Users/huhuhu/Desktop/sublime文件/git
/Users/huhuhu/Desktop/sublime文件/git

```

2.将需要上传的文件夹放入” git” 文件夹中，并将需要显示的.md文件改名为” README.md”



3.根据github上传的指令提示,

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

Set up in Desktop or **HTTPS** **SSH** `https://github.com/hujinbei/Root.git`

We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

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

```
echo "# Root" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/hujinbei/Root.git
git push -u origin master
```

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

```
git remote add origin https://github.com/hujinbei/Root.git
git push -u origin master
```

...or import code from another repository

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

4.输入初始的命令” git init” 初始化现存的git仓库并且添加一个隐藏的”.git” 目录，然后尝试上传命令，输入” git add redbox(上传文件夹的名称)” ，然后输入” git commit -m “对上传文件的描述”

- 出现了” Please tell me who you are.” （未设置登入信息）错误提示，根据错误提示中的解决建议，在终端输入自己github账号的名称和邮箱

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



```

huhuhudeMac-mini:git huhuhu$ git init
Initialized empty Git repository in /Users/huhuhu/Desktop/sublime文件/git/.git/
huhuhudeMac-mini:git huhuhu$ ls
redbox
huhuhudeMac-mini:git huhuhu$ git add redbox
huhuhudeMac-mini:git huhuhu$ git commit -m "将红色方框转换为蓝色的方框效果"

*** 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 'huhuhu@huhuhudeMini.(none)')
huhuhudeMac-mini:git huhuhu$ git config --global user.name "hujinbei"
huhuhudeMac-mini:git huhuhu$ git config --global user.email "hujinbei@protonmail
e.com"
huhuhudeMac-mini:git huhuhu$ git config --global user.email "hujinbei@protonmail
.com"

```

- 然后尝试根据以上方法重新上传，根据终端的命令信息提示，表示该文件夹中的内容已经上传到了版本库中，但还未上传至github，

```

huhuhudeMac-mini:git huhuhu$ pwd
/Users/huhuhu/Desktop/sublime文件/git
huhuhudeMac-mini:git huhuhu$ ls
redbox
huhuhudeMac-mini:git huhuhu$ git init
Reinitialized existing Git repository in /Users/huhuhu/Desktop/sublime文件/git/.
git/
huhuhudeMac-mini:git huhuhu$ ls
redbox
huhuhudeMac-mini:git huhuhu$ git add redbox/
huhuhudeMac-mini:git huhuhu$ git commit -m "将红色方框转换为蓝色的方框效果"
[master (root-commit) 83ca981] 将红色方框转换为蓝色的方框效果
14 files changed, 398 insertions(+)
create mode 100644 redbox/.DS_Store
create mode 100644 redbox/README.md
create mode 100644 redbox/image/.DS_Store
create mode 100644 redbox/image/1.png
create mode 100644 redbox/image/2.png
create mode 100644 redbox/image/3.png
create mode 100644 redbox/image/4.png
create mode 100644 redbox/image/5.png
create mode 100644 redbox/image/7.png
create mode 100644 redbox/image/8.png
create mode 100644 redbox/redblock.html

```

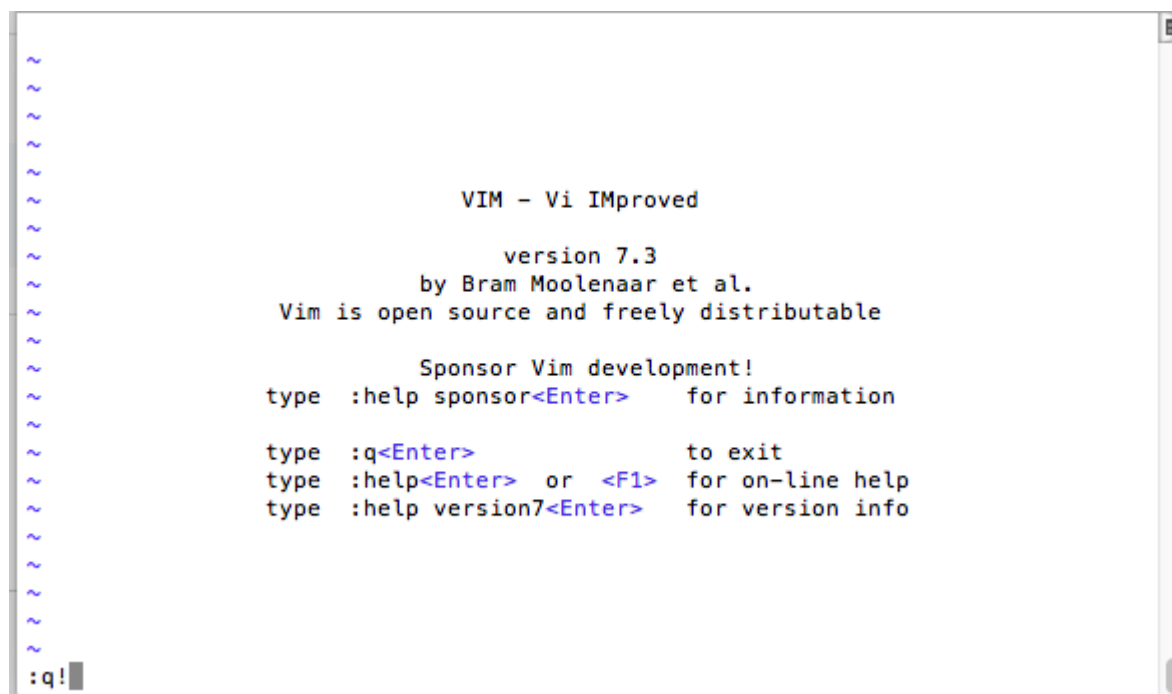
- 根据命令提示，由于首次登入github上传文件时，需要输入一下命令：“git remote add origin + 你需要上传的github仓库位置”
- 接下来输入“git push -u origin master”命令，登入github，输入用户名及密码
 - 错误提示“error: failed to push some refs to ‘<https://github.com/hujinbei/Study.git>’”，push上传错误后根据教程，输入“get pull origin master”命令后，

```

Create mode 100644 'redbox/\345\210\233\345\273\272\346\220\271\345\235\227\344
\271\213\351\227\264\347\232\204\350\275\254\346\215\242.html"
create mode 100644 "redbox/\345\210\233\345\273\272\346\226\271\345\235\227\344
\271\213\351\227\264\347\232\204\350\275\254\346\215\242.pdf"
huhuhudeMac-mini:git huhuhu$ git remote add origin https://github.com/hujinbei/S
tudy.git
huhuhudeMac-mini:git huhuhu$ git push -u origin master
Username for 'https://github.com': hujinbei
Password for 'https://hujinbei@github.com':
To https://github.com/hujinbei/Study.git
 ! [rejected]        master -> master (fetch first)
error: failed to push some refs to 'https://github.com/hujinbei/Study.git'
hint: Updates were rejected because the remote contains work that you do
hint: not have locally. This is usually caused by another repository pushing
hint: to the same ref. You may want to first integrate the remote changes
hint: (e.g., 'git pull ...') before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
huhuhudeMac-mini:git huhuhu$ git pull origin master
warning: no common commits
remote: Counting objects: 146, done.
remote: Compressing objects: 100% (141/141), done.
remote: Total 146 (delta 8), reused 142 (delta 4), pack-reused 0
Receiving objects: 100% (146/146), 11.36 MiB | 1.32 MiB/s, done.

```

- 会出现” vim” 编辑框(提交更改时显示), 然后需要” Esc” —>输入” :+q+!” —>” Enter (回车) ”, 即可退出此编辑框, 然后会自动更改提交的操作, 如需再次打开vim, 则在终端输入” vim” 就可以打开了;
 - 之后产生的数据, 就意味着将原有github仓库中的文件全部下载放入新建的” git” 文件夹中, 打开文件夹发现新增许多仓库中下载的文件夹:



```

create mode 100044 redbox/\345\210\233\345\273\272\346\226\271\345\235\227\344\271\213\351\227\264\347\232\204\350\275\254\346\215\242.html"
create mode 100644 "redbox/\345\210\233\345\273\272\346\226\271\345\235\227\344\271\213\351\227\264\347\232\204\350\275\254\346\215\242.pdf"
huhuhudeMac-mini:git huhuhu$ git remote add origin https://github.com/hujinbei/Study.git
huhuhudeMac-mini:git huhuhu$ git push -u origin master
Username for 'https://github.com': hujinbei
Password for 'https://hujinbei@github.com':
To https://github.com/hujinbei/Study.git
 ! [rejected]        master -> master (fetch first)
error: failed to push some refs to 'https://github.com/hujinbei/Study.git'
hint: Updates were rejected because the remote contains work that you do
hint: not have locally. This is usually caused by another repository pushing
hint: to the same ref. You may want to first integrate the remote changes
hint: (e.g., 'git pull ...') before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
huhuhudeMac-mini:git huhuhu$ git pull origin master
warning: no common commits
remote: Counting objects: 146, done.
remote: Compressing objects: 100% (141/141), done.
remote: Total 146 (delta 8), reused 142 (delta 4), pack-reused 0
Receiving objects: 100% (146/146), 11.36 MiB | 1.32 MiB/s, done.

```

```

Receiving objects: 100% (146/146), 11.36 MiB | 1.32 MiB/s, done.
Resolving deltas: 100% (8/8), done.

```

```

From https://github.com/hujinbei/Study
 * branch                master      -> FETCH_HEAD
 * [new branch]          master      -> origin/master

```

Merge made by the 'recursive' strategy.

Github/README.md	84 +++++
Github/contributions5.png	Bin 0 -> 60448 bytes
Github/create a new repository7.png	Bin 0 -> 55211 bytes
Github/github.html	26 ++
"Github/github\346\263\250\345\206\2141.png"	Bin 0 -> 22053 bytes
...\212\344\274\240\346\210\220\345\212\23713.png"	Bin 0 -> 64913 bytes
...\212\344\274\240\346\210\220\345\212\23714.png"	Bin 0 -> 70754 bytes
...\212\344\274\240\346\210\220\345\212\23715.png"	Bin 0 -> 40878 bytes
...\212\344\274\240\346\210\220\345\212\23716.png"	Bin 0 -> 57761 bytes
...\212\344\274\240\346\226\207\344\273\26611.png"	Bin 0 -> 89205 bytes
...\212\344\274\240\346\226\207\344\273\26612.png"	Bin 0 -> 26080 bytes
"Github/git\344\277\256\346\224\27117.png"	Bin 0 -> 77753 bytes
"Github/git\345\256\211\350\243\20510.png"	Bin 0 -> 38827 bytes
"Github/git\345\256\211\350\243\2058.png"	Bin 0 -> 125083 bytes
"Github/git\345\256\211\350\243\2059.png"	Bin 0 -> 45823 bytes
"Github/git\351\224\231\350\257\25718.png"	Bin 0 -> 64637 bytes
"Github/git\351\224\231\350\257\25719.png"	Bin 0 -> 53334 bytes
"Github/git\351\224\231\350\257\25720.png"	Bin 0 -> 22694 bytes



redbox



数据库



虚拟机安装



Github



HTML



markdown教程



nodejs



z变换教案制作

- 更新数据操作完成后，再次输入“git push -u origin master”命令，就可以成功上传

```
huhuhudeMac-mini:git huhuhu$ git push -u origin master
Counting objects: 20, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (19/19), done.
Writing objects: 100% (20/20), 655.52 KiB | 0 bytes/s, done.
Total 20 (delta 3), reused 0 (delta 0)
remote: Resolving deltas: 100% (3/3), completed with 1 local object.
To https://github.com/hujinbei/Study.git
    7ded4e5..b05b58e  master -> master
Branch master set up to track remote branch master from origin.
```