

GitHub 新手使用手册



小贤笔记

关注

3 2019.01.23 11:10:05 字数 1,738 阅读 10,570

文章推荐:

[Git 命令总结](#)[如何在同一台电脑上使用 github 和 gitlab](#)

GitHub是基于git实现的代码托管。git是目前最好用的版本控制系统了，非常受欢迎，比之svn更好。

GitHub可以免费使用，并且快速稳定。即使是付费帐户，每个月不超过10美刀的费用也非常便宜。

利用GitHub，你可以将项目存档，与其他人分享交流，并让其他开发者帮助你一起完成这个项目。优点在于，他支持多人共同完成一个项目，因此你们可以在同一页面对话交流。

创建自己的项目，并备份，代码不需要保存在本地或者服务器，GitHub做得非常理想。

学习Git也有很多好处。他被视为一个预先维护过程，你可以按自己的需要恢复、提交出现问题或者您需要恢复任何形式的代码，可以避免很多麻烦。Git最好的特性之一是能够跟踪错误，这让使用Github变得更加简单。Bugs可以公开，你可以通过Github评论，提交错误。

在GitHub页面，你可以直接开始，而不需要设置主机或者DNS。

对于我来说，我可以在公司和家编辑同样的一套代码了，用U盘或者网盘都好麻烦的说

理论说太多容易忘，来实践实践吧。

写下你的评论...

评论0

赞41

...



小贤笔记

关注

总资产129 (约12.10元)

[react 配置组件路径引用 @](#)

阅读 858

[JavaScript Array 对象方法总结](#)

阅读 204

推荐阅读

[用Pytorch做人脸识别](#)

阅读 4,291

[async和await](#)

阅读 2,627

[用大白话解释什么是Socket](#)

阅读 457

[汇量科技前端（实习）面试总结](#)

阅读 2,281

[Pycharm2020.1激活教程，最新！](#)

阅读 5,278

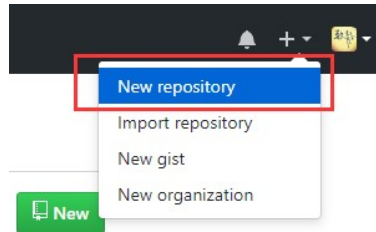
1. 登录 github

2. 创建 repository(仓库)

为啥要叫 repository(仓库)? 我起初也纳闷, 叫代码库不更简单明了么? 但仔细一琢磨, 仓库一般都是放粮食的吧, 这是把代码当作饱腹之物, 多有爱, 瞬间觉得这冰冷的代码充满了查克拉。

扯远了, 来看怎么创建仓库

看到右上角 "+" 没? 点击选择 "New repository"



下面是创建仓库信息, 只有名字是必填项

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner

Repository name *

shenxianhui

 /

Great repository names are short and memorable. Need inspiration? How about **redesigned-waddle**.

Description (optional)

☒ Public

Anyone can see this repository. You choose who can commit.

☐ Private

You choose who can see and commit to this repository.

☐ Initialize this repository with a README

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None

Add a license: None

Create repository

创建成功后, 可以看到自己的仓库地址。它还介绍了github仓库的常用指令, 这个指令需要在本地安装 git 客户端

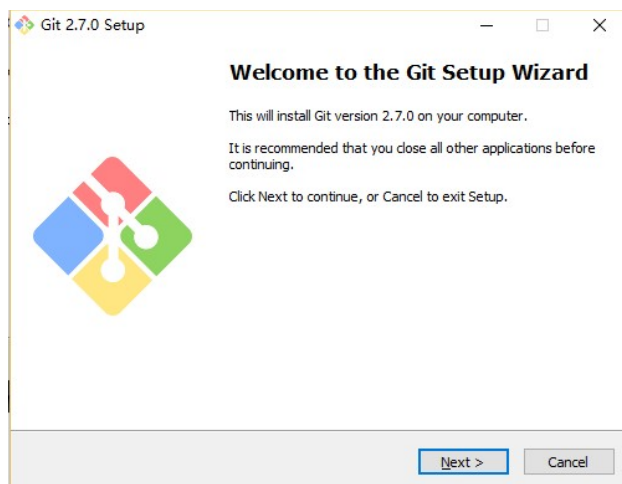
```
1 | git init // 把这个目录变成Git可以管理的仓库
2 | git add README.md // 文件添加到仓库
3 | git add . // 不但可以跟单一文件, 还可以跟通配符, 更可以跟目录。一个点就把当前目录下所有未追踪的文件全
4 | git commit -m "first commit" // 把文件提交到仓库
5 | git remote add origin git@github.com:wangjiay9/practice.git // 关联远程仓库
6 | git push -u origin master // 把本地库的所有内容推送到远程库上
```

二. 安装 git 客户端

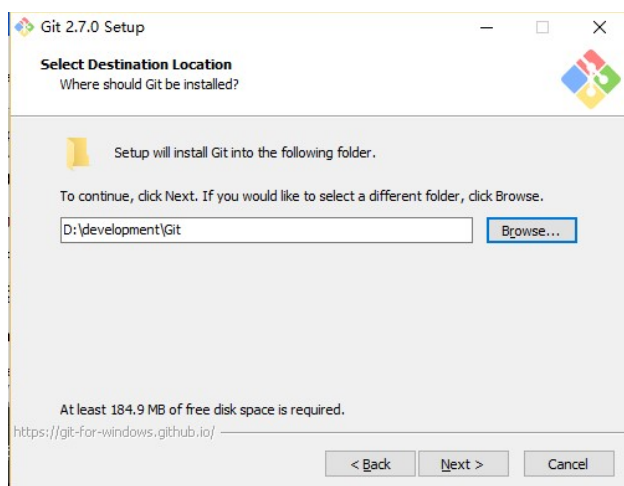
1. 下载 git



2. 安装客户端

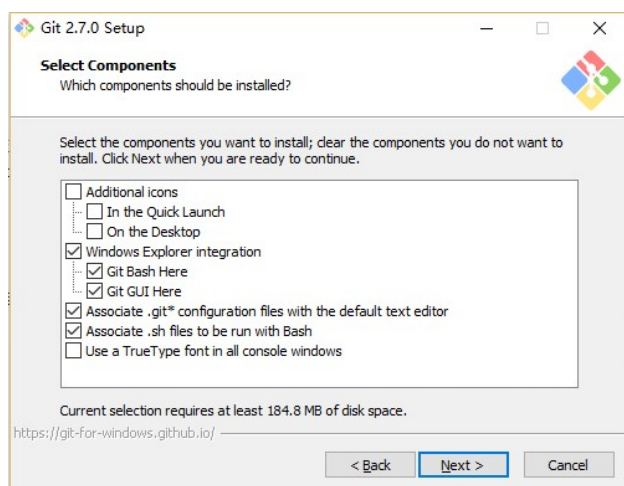


选择安装路径，千万别选带中文的路径，有时候会引起不必要的误会



选择安装组件，按默认的来就好了

- 1) 图标组件(Addition icons) : 选择是否创建快速启动栏图标 或者 是否创建桌面快捷方式;
- 2) 桌面浏览(Windows Explorer integration) : 浏览源码的方法, 单独的上下文浏览 只使用bash 或者 只用Git GUI工具; 高级的上下文浏览方法 使用git-cheetah plugin插件;
- 3) 关联配置文件(Associate .git*) : 是否关联git配置文件, 该配置文件主要显示文本编辑器的样式;
- 4) 关联shell脚本文件(Associate .sh) : 是否关联Bash命令行执行的脚本文件;
- 5) 使用TrueType编码 : 在命令行中是否使用TruthType编码, 该编码是微软和苹果公司制定的通用编码;



设置开始菜单中快捷方式的目录名称，默认就好，下一步吧

写下你的评论...

评论0 赞41 ...

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

Set up in Desktop or HTTPS SSH

We recommend every repository include a README, LICENSE, and .gitignore.

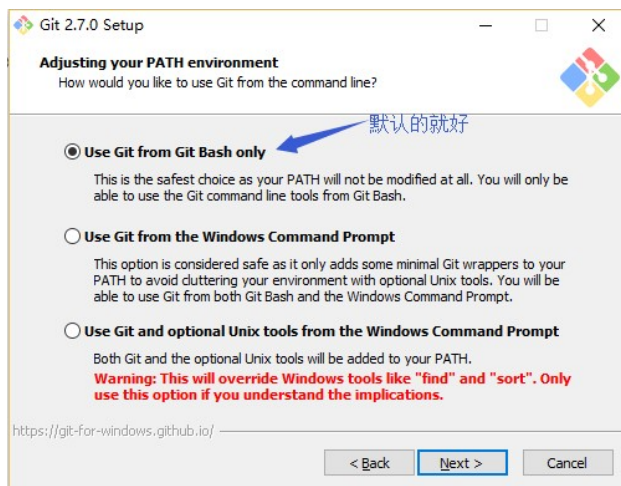
...or create a new repository on the command line

```
echo "# beautifulDay" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin git@github.com:wangjiax9/beautifulDay.git
git push -u origin master
```

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

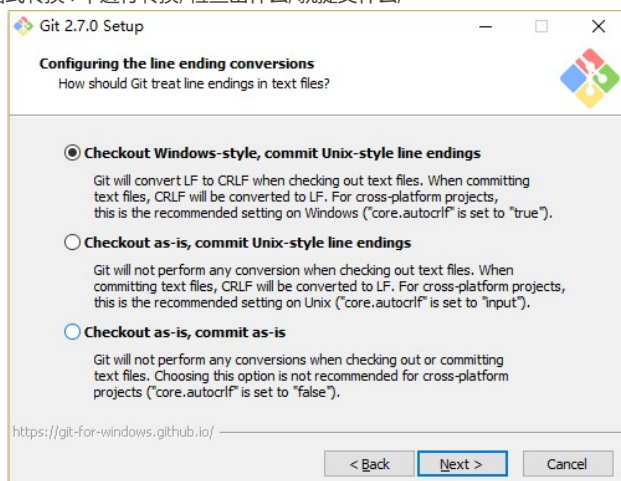
设置环境变量：选择使用什么样的命令行工具，一般情况下我们默认使用Git Bash即可，默认选择：

- 1) Git自带：使用Git自带的Git Bash命令行工具；
- 2) 系统自带CMD：使用Windows系统的命令行工具；
- 3) 二者都有：上面二者同时配置，但是注意，这样会将windows中的find.exe 和 sort.exe工具覆盖，如果不懂这些尽量不要选择；



选择换行格式，依然是默认就好。

- 1) 检查出windows格式转换为unix格式：将windows格式的换行转为unix格式的换行在进行提交；
- 2) 检查出原来格式转为unix格式：不管什么格式的，一律转为unix格式的换行在进行提交；
- 3) 不进行格式转换：不进行转换，检查出什么，就提交什么；

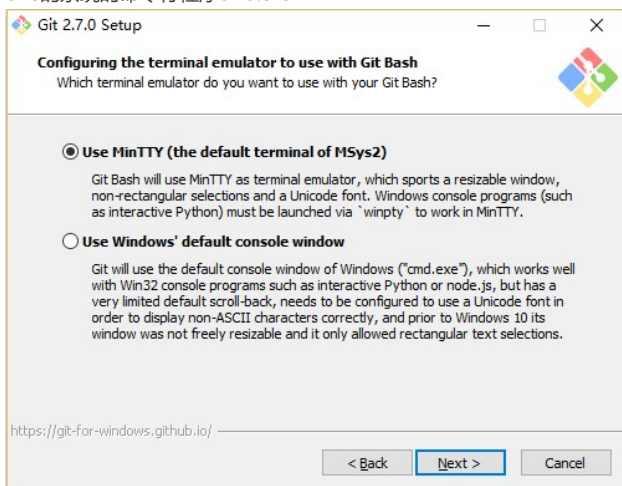


写下你的评论...

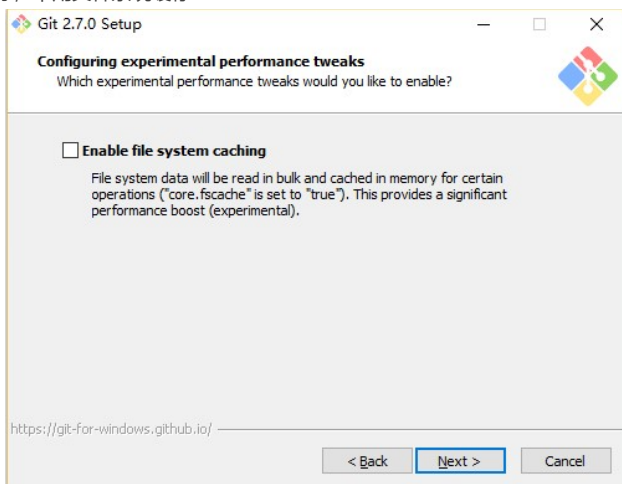
评论0 赞41 ...

选择终端模拟器，依然默认就好

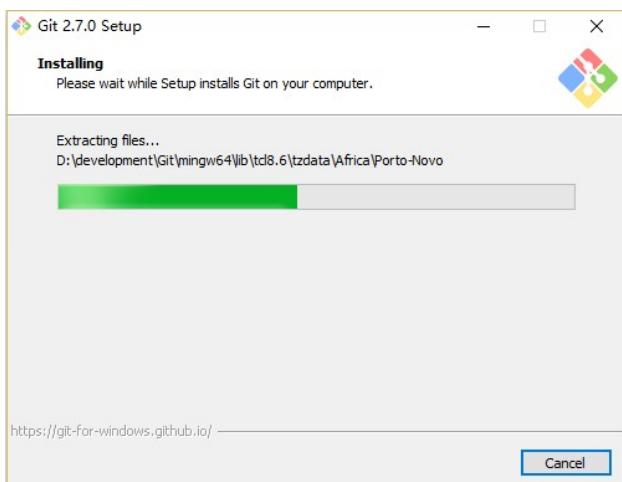
- 1) 使用MinTTY，就是在Windows开了一个简单模拟Linux命令环境的窗口Git Bash
- 2) 使用windows的系统的命令行程序cmd.exe



选择默认就好，不用文件系统缓存



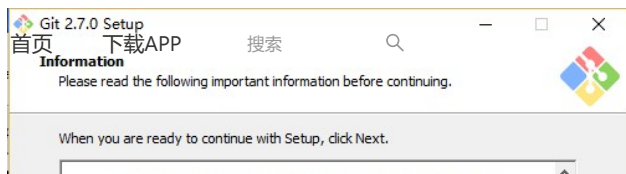
安装ing...



安装成功

写下你的评论...

评论0 赞41 ...



3. 绑定用户

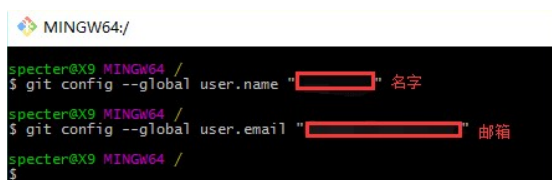
打开git-bash.exe, 在桌面快捷方式/开始菜单/安装目录中

因为Git是分布式版本控制系统, 所以需要填写用户名和邮箱作为一个标识, 用户和邮箱为你github注册的账号和邮箱

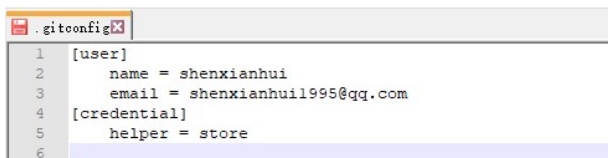
```
git config --global user.name "Your Name"
```

```
git config --global user.email "your_email@example.com"
```

注: [your_email@example.com](#) 的部分请改成你在创建账户时用的邮箱地址



这个命令会在 C:\Users\Administrator (Administrator为电脑用户名, 每个人不同) 目录下生成 .gitconfig 文件



PS: 下面代码是为了记住用户名和密码, 不用每次都输入, 需要的直接添加保存即可



当然, 还有其他方法保存, 可以直接在git bash 中执行命令:

- 记住密码十五分钟

```
git config --global credential.helper cache
```

- 自定义记住的时间 (单位: 秒)

```
git config credential.helper 'cache --timeout=3600'
```

- 永久记住

```
git config --global credential.helper store
```

最后再输入一次账号密码就可以保存了

众所周知 ssh key 是加密传输。

加密传输的算法有好多，git 使用 rsa，rsa 要解决的一个核心问题是，如何使用一对特定的数字，使其中一个数字可以用来加密，而另外一个数字可以用来解密。这两个数字就是你在使用 git 和 github 的时候所遇到的 public key 也就是公钥以及 private key 私钥。

其中，公钥就是那个用来加密的数字，这也就是为什么你在本机生成了公钥之后，要上传到 github 的原因。从 github 发回来的，用那公钥加密过的数据，可以用你本地的私钥来还原。

如果你的 key 丢失了，不管是公钥还是私钥，丢失一个都不能用了，解决方法也很简单，重新再生成一次，然后在 github.com 里再设置一次就行

1. 生成 SSH Key

首先检查是否已生成密钥 `cd ~/.ssh`

`ls` 如果有3个文件，则密钥已经生成，id_rsa.pub 就是公钥

```
specter@X9 MINGW64 /
$ cd ~/.ssh

specter@X9 MINGW64 ~/.ssh
$ ls
id_rsa  id_rsa.pub  known_hosts
```

也可以打开我的电脑 C:\Users\Administrator\.ssh (Administrator为电脑用户名，每个人不同) 里面找到

此电脑 > SYSTEM (C:) > 用户 > specter > .ssh		
名称		修改
id_rsa		2016
id_rsa.pub		2016
known_hosts		2016

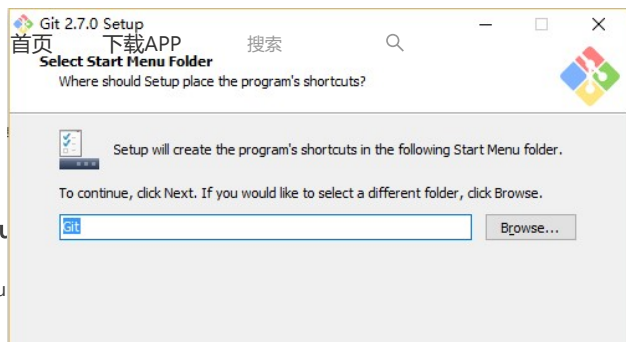
如果没有生成，那么通过 `$ ssh-keygen -t rsa -C "your_email@example.com"` 来生成

注: `your_email@example.com` 的部分请改成你在创建账户时用的邮箱地址

- 1) 是路径确认，直接按回车存默认路径即可
- 2) 直接回车键，这里我们不使用密码进行登录，用密码太麻烦；
- 3) 直接回车键

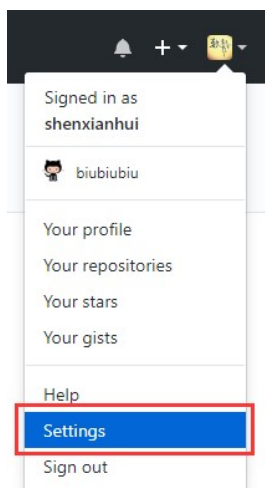
```
specter@X9 MINGW64 ~
$ ssh-keygen -t rsa -C "specter@163.com"
Generating public/private rsa key pair.
Enter file in which to save the key (/c:/Users/specter/.ssh/id_rsa):
Enter passphrase (empty for no passphrase): 输入密码
Enter same passphrase again: 再次输入密码
Your identification has been saved in /c:/Users/specter/.ssh/id_rsa.
Your public key has been saved in /c:/Users/specter/.ssh/id_rsa.pub.
The key's fingerprint is:
SHA256:G600Vs2veJ8YmgBgaKinb3SrxTvD15ijqvo4Y4ZRrC8 6215048wj1@163.com
The key's randomart image is:
+----[RSA 2048]-----+
|+
|+0
|O..
|. +.
|= . S .
|+ .... +
|=. 00.+++
|E+. =00o .
|BB+00=++*++
+----[SHA256]-----+
```

生成成功后，去对应目录 C:\Users\Administrator\.ssh (Administrator为电脑用户名，每个人不同) 用记事本打开 id_rsa.pub，得到 ssh key公钥

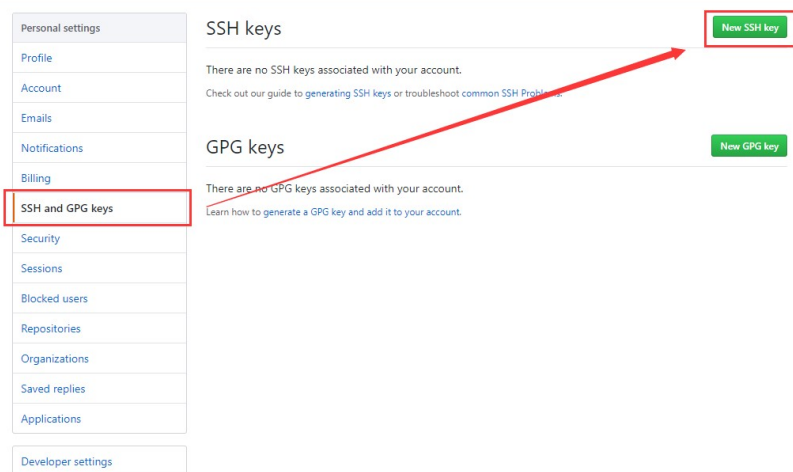


2. 为 github

切换到 github

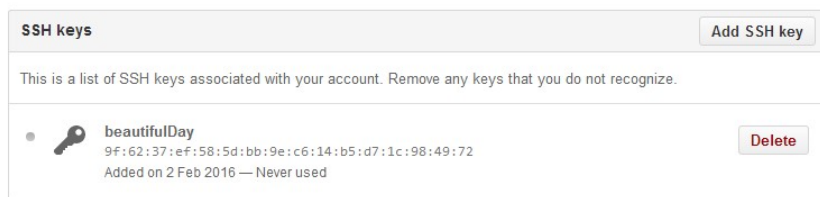


然后打开 SSH and GPG keys 菜单， 点击 New SSH key 新增密钥



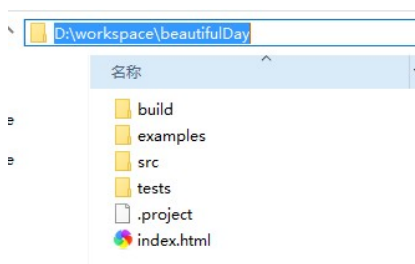
标题建议和仓库名一致, 便于区分. 接着将 id_rsa.pub 文件中 key 粘贴到此, 最后 Add SSH key 生成密钥

如此，github账号的SSH keys配置完成



四. 上传本地项目到 github

1. 创建一个本地项目



2. 建立本地仓库

再来复习一下创建新仓库的指令：

```
1 | git init // 把这个目录变成Git可以管理的仓库
2 | git add README.md // 文件添加到仓库
3 | git add . // 不但可以跟单一文件，还可以跟通配符，更可以跟目录。一个点就把当前目录下所有未追踪的文件全
4 | git commit -m "first commit" // 把文件提交到仓库
5 | git remote add origin git@github.com:wangjiay9/practice.git // 关联远程仓库
6 | git push -u origin master // 把本地库的所有内容推送到远程库上
```

首先，进入到 beautifulDay 项目目录，还记得创建仓库成功后的那个页面吧，指令都在呢

然后执行指令：`git init`

```
specter@X9 MINGW64 ~
$ cd d:
specter@X9 MINGW64 /d
$ cd workspace/beautifulDay/
specter@X9 MINGW64 /d/workspace/beautifulDay
$ git init
Initialized empty Git repository in D:/workspace/beautifulDay/.git/
```

初始化成功后你会发现项目里多了一个隐藏文件夹.git

这个目录是Git用来跟踪管理版本库的，没事千万不要手动修改这个目录里面的文件，不然改乱了，就把Git仓库给破坏了。

接着，将所有文件添加到仓库

执行指令：`git add .`

```
specter@X9 MINGW64 /d/workspace/beautifulDay (master)
$ git add .
```

然后把文件提交到仓库，双引号内是提交注释

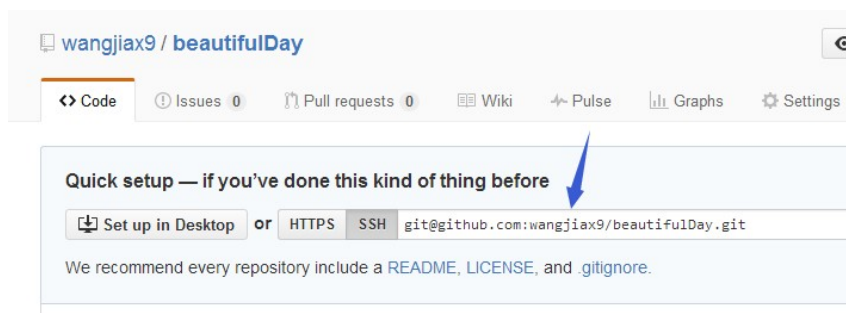
执行指令：`git commit -m "提交文件"`

```
specter@X9 MINGW64 /d/workspace/beautifulDay (master)
$ git commit -m "提交文件"
[master (root-commit) 61c1615] 提交文件
2 files changed, 53 insertions(+)
create mode 100644 .project
create mode 100644 index.html
```

如此本地仓库建立好了

3. 关联 github 仓库

到 github beautifulDay 仓库复制仓库地址



然后执行指令：`git remote add origin git@github.com:wangjiax9/beautifulDay.git`

```
specter@X9 MINGW64 /d/workspace/beautifulDay (master)
$ git remote add origin git@github.com:wangjiax9/beautifulDay.git
```

4. 上传本地代码

执行指令：`git push -u origin master`

敲一个：yes，然后回车

```
specter@X9 MINGW64 /d/workspace/beautifulDay (master)
$ git push -u origin master
The authenticity of host 'github.com (192.30.252.128)' can't be established.
RSA key fingerprint is SHA256:nThbg6kXUpJWGl7EiIG0Cspr0mTxdCARLviKw6E5SY8.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'github.com,192.30.252.128' (RSA) to the list of known hosts.
Counting objects: 4, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 734 bytes | 0 bytes/s, done.
Total 4 (delta 0), reused 0 (delta 0)
To git@github.com:wangjiax9/beautifulDay.git
 * [new branch]      master -> master
Branch master set up to track remote branch master from origin.
```

到此，本地代码已经推送到github仓库了，我们现在去 github 仓库看看

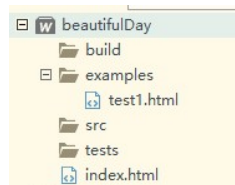
噢！奇怪了，我的目录呢？这个坑突然冒出来是不是印象很深刻呢~

注意咯：git是不能管理空的文件夹的，文件夹里必须有文件才能add

写下你的评论...

评论0 赞41 ...

好，我们来试一下，我在examples里新建了一个 test1.html 文件



执行指令添加文件->提交文件->推送文件

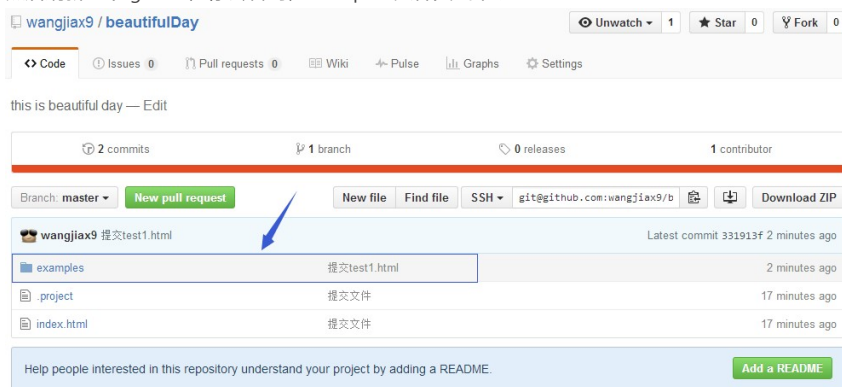
```
git add .
```

```
git commit -m "提交test1.html"
```

```
git push -u origin master
```

```
specter@X9 MINGW64 /d/workspace/beautifulDay (master)
$ git add .
specter@X9 MINGW64 /d/workspace/beautifulDay (master)
$ git commit -m "提交test1.html"
[master 331913f] 提交test1.html
1 file changed, 9 insertions(+)
create mode 100644 examples/test1.html
specter@X9 MINGW64 /d/workspace/beautifulDay (master)
$ git push -u origin master
Warning: Permanently added the RSA host key for IP address '192.30.252.131' to the list of known hosts.
Counting objects: 4, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 449 bytes | 0 bytes/s, done.
Total 4 (delta 0), reused 0 (delta 0)
To git@github.com:wangjiax9/beautifulDay.git
61c1615..331913f master -> master
Branch master set up to track remote branch master from origin.
```

然后刷新一下 github，你会看到，examples 文件夹出来了



总结

现在通过博客来将 github 入门梳理了一遍，印象好深刻，应该不会变成零碎了。如果大家想要知道更多的git操作和知识，推荐去看看[缪雪峰的git教程](#)，讲得很详细



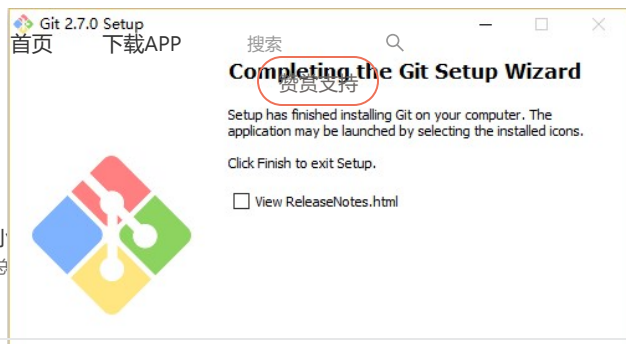
写下你的评论...

评论0

赞41

...





关注

Powered By Intel Xeon Platinum

Limited offer! High-Performance Virtual Machines with Data Transfer Plan from \$2.40/1st Mo

OPEN



写下你的评论...

全部评论 0 只看作者

按时间倒序 按时间正序

被以下专题收入，发现更多相似内容



全栈笔记



让前端飞



Python(...)

推荐阅读

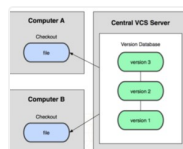
更多精彩内容 >

GitHub,OSChina,码市的详细使用

一、Git的简介 1.1.什么是git? git是一款开源的分布式版本控制工具,在世界上所有的分布式版本控制工具中, ...



llIronMan 阅读 520 评论 0 赞 13

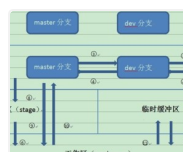


git简明使用手册

git作为时下最流行的代码管理工具, Git权威指南总结了十条喜欢Git的理由: 异地协同工作; 现场版本控制; 重...



古猷布衣 阅读 940 评论 0 赞 10

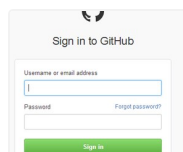


github入门到上传本地项目

一、创建github repository(仓库) 1-1 登录github github的官方网址: https:...



字短情长 阅读 720 评论 0 赞 0



Git命令学习笔记

1. 安装 Github 查看是否安装git: \$ git config --global user.name "...



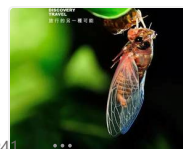
Albert_Sun 阅读 9,519 评论 9 赞 161

夏至 追忆田舍雅趣

每年6月21或者22日, 阳光直射到它在地球的最北端, 几近北回归线。太阳的运动过于明显, 因此夏至是二十四节气中最早...



自在行摄 阅读 70 评论 0 赞 0



写下你的评论...

评论0

赞4

写下你的评论...

 评论0  赞41 ...

写下你的评论...

 评论0  赞41 ...



写下你的评论...

 评论0  赞41 ...

写下你的评论...

 评论0  赞41 ...

Title

写下你的评论...

写下你的评论...

 评论0  赞41 ...

写下你的评论...

 评论0  赞41 ...

写下你的评论...

 评论0  赞41 ...

wangjiax9 / beautifulDay

Unwatch

1

Star

0

Fork

0

Code

Issues0

Pull requests0

Wiki

Pulse

Graphs

Settings

this is beautiful day — Edit

1 commit

1 branch

0 releases

1 contributor

Branch: master

New pull request

New file

Find file

SSH

git@github.com:wangjiax9/b

Download ZIP

wangjiax9 提交文件

Latest commit 61c1615 10 minutes ago