



# Git 与 GitHub



# 目 录

- 版本控制系统介绍
- Git 与 GitHub 简介
- GitHub账户注册与管理
- 工作区、本地仓库、远程仓库
- 创建版本库
- 修改管理

# 引入

团队合作

代码量大

维护？

频繁迭代

# 版本控制系统介绍

- 版本控制  $\neq$  版本控制系统
- 版本控制
  - 它是一种软件工程借以在开发的过程中，确保由不同人所编辑的同一档案都得到更新，它通过文档控制记录程序各个模块的改动，并为每次改动编上序号，并且编辑错误之后还可以回溯到以前的版本

# 版本控制系统介绍

- 版本控制  $\neq$  版本控制系统
- 版本控制系统
  - 版本控制系统是一种记录一个或若干文件内容变化，以便将来查阅特定版本修订情况的系统
  - 版本控制系统不仅可以应用于软件源代码的文本文件，而且可以对任何类型的文件进行版本控制

# 版本控制系统介绍

- 为什么要使用版本控制系统？
- 场景：
  - 当某程序员修改了关于A功能的若干处代码，这时候，他发现B功能无法正常运作了。。。
- 解决策略：
  - 必须精确找出来刚改的这若干处代码，需要将代码与上一次还能正常运行时候的代码进行比较



版本  
控制

# 版本控制系统介绍

- 为什么要使用版本控制系统？
- 团队合作常见问题
  - 1.不小心删除了自己的项目文档和源代码文件
  - 2.不敢修改自己的源代码文件
  - 3.不知道如何把自己项目文档传递给他人
  - 4.不知道如何把自己的项目分发
  - 5.不断的拷贝和备份
  - 6.文档和源代码管理混乱，无法查找

# 版本控制系统

- 版本控制系统功能
  - 记录文件的所有历史变化
  - 随时恢复到任何一个历史状态
  - 多人协作开发或修改
  - 错误恢复
  - 多功能并行开发（分支）



# 版本控制系统

- 版本控制系统分类
  - 本地版本控制系统 ( VCS )
  - 集中式版本控制系统 ( VSS、VCS、SVN )
  - 分布式版本控制系统 ( Git )



# 目 录

- 版本控制系统介绍
- Git 与 GitHub 简介
- Git账户注册与管理
- 工作区、本地仓库、远程仓库
- 创建版本库
- 修改管理

# Git ( 分布式版本管理系统 )

- Git 由 Linux 创始人Linus Torvalds开发，作为Linux 内核代码管理系统使用
- Git是一款**免费、开源**的分布式版本控制系统，用于敏捷高效地处理任何**或小或大**的项目



# Git ( 分布式版本管理系统 )

- 分布式 VS 集中式
  - 分布式相比于集中式的最大区别在于开发者可以提交到本地，每个开发者通过克隆，在本地机器上拷贝一个完整的Git仓库

# Git ( 分布式版本管理系统 )

- Git的特点 ( 从开发者角度考虑 )
  - 从服务器上克隆完整的Git仓库 ( 包括代码和版本信息 ) 到单机上
  - 在自己的机器上根据不同的开发目的 , 创建分支 , 修改代码
  - 在单机上自己创建的分支上提交代码
  - 在单机上合并分支

# Git ( 分布式版本管理系统 )

- Git的特点 ( 从开发者角度考虑 )
  - 把服务器上最新版的代码fetch下来，然后跟自己的主分支合并
  - 查看邮件或者通过其它方式查看提交状态
  - 解决冲突

# Git ( 分布式版本管理系统 )

## • Git的优点

- 适合分布式开发，强调个体
- 速度快
- 简单的设计
- 对非线性开发模式的强力支持
- 有能力高效管理类似Linux内核一样的超大规模项目
- 离线工作

# Git ( 分布式版本管理系统 )

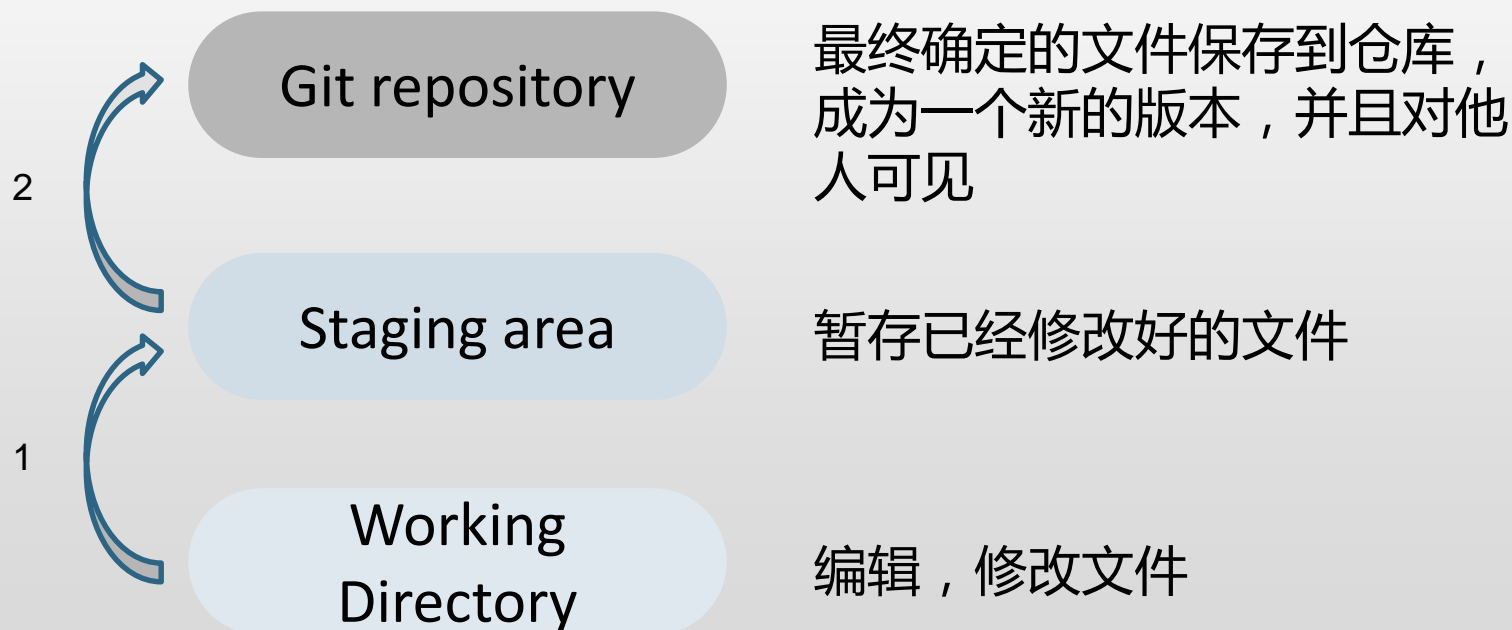
- Git的缺点

- 代码保密性差，一旦开发者把整个库克隆下来就可以完全公开所有代码和版本信息
- 资料少（中文资料很少）
- 学习周期相对而言比较长
- 不符合常规思维



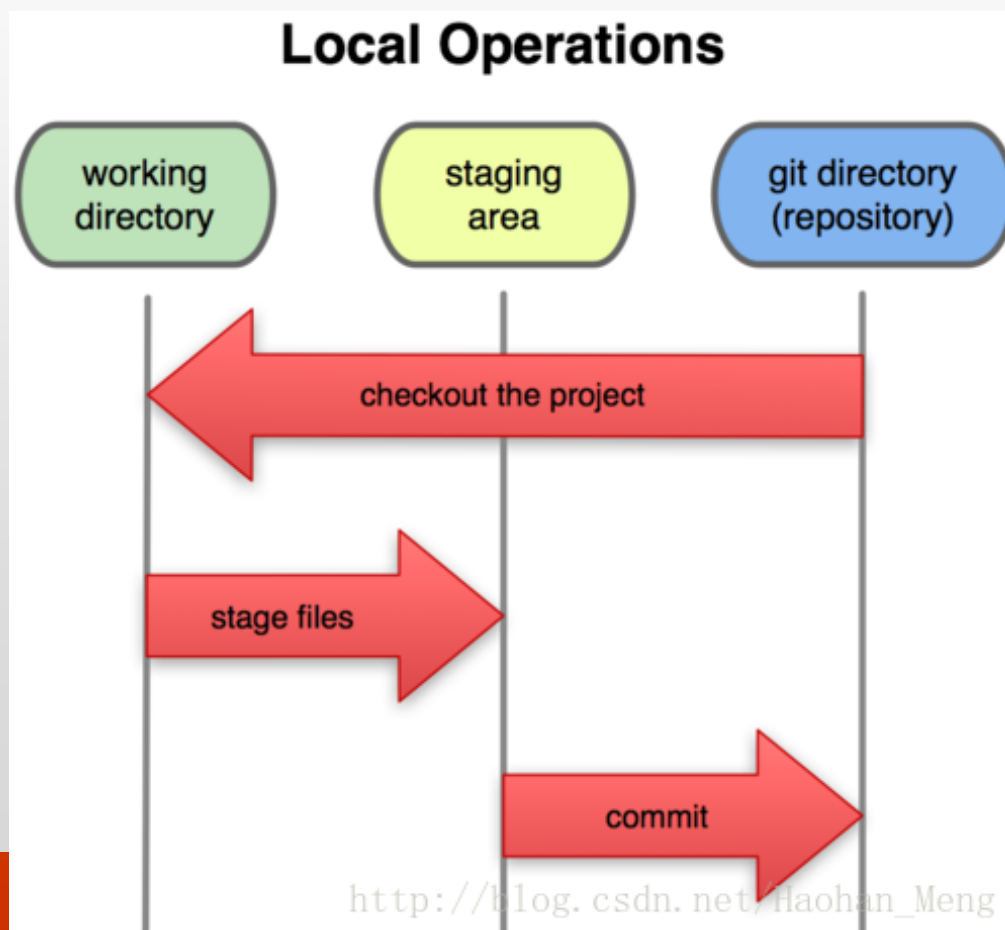
# Git ( 分布式版本管理系统 )

- Git 工作原理 - 文件状态及工作区域



# Git ( 分布式版本管理系统 )

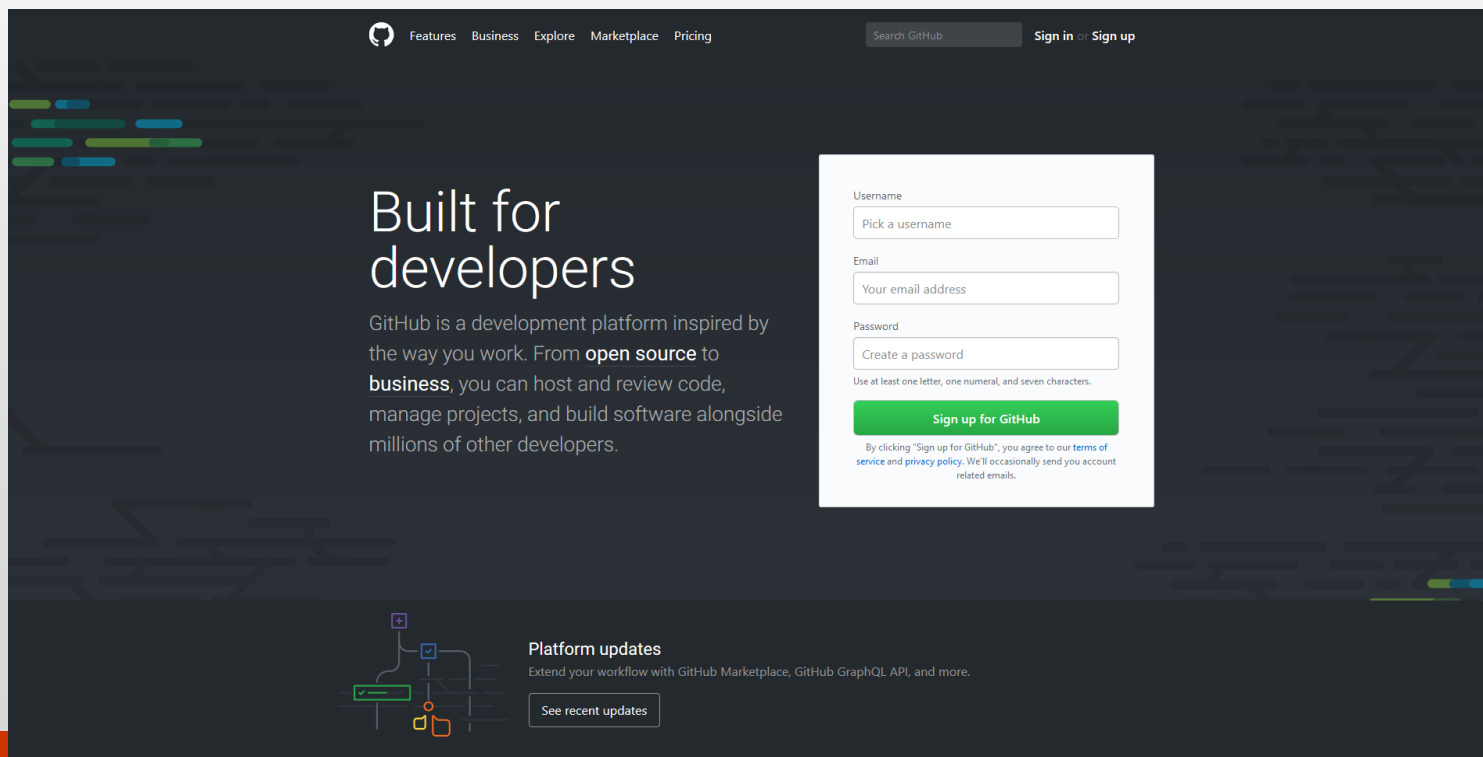
- Git 工作原理 - 文件状态及工作区域



# GitHub



- 面向开源及私有软件项目的托管平台
- 官网：<https://github.com/>



# GitHub ( 官网 : <https://github.com/> )



- GitHub 于 2008 年 4 月 10 日正式上线
- 功能
  - Git 代码仓库托管及基本的 Web 管理界面
  - 订阅、讨论组、文本渲染、在线文件编辑器、协作图谱 ( 报表 )、代码片段分享 ( Gist ) 等
- 注册用户
  - 超过900万 (2015年统计数据)
- 托管项目 : Ruby on Rails、jQuery、python 等



# 目 录

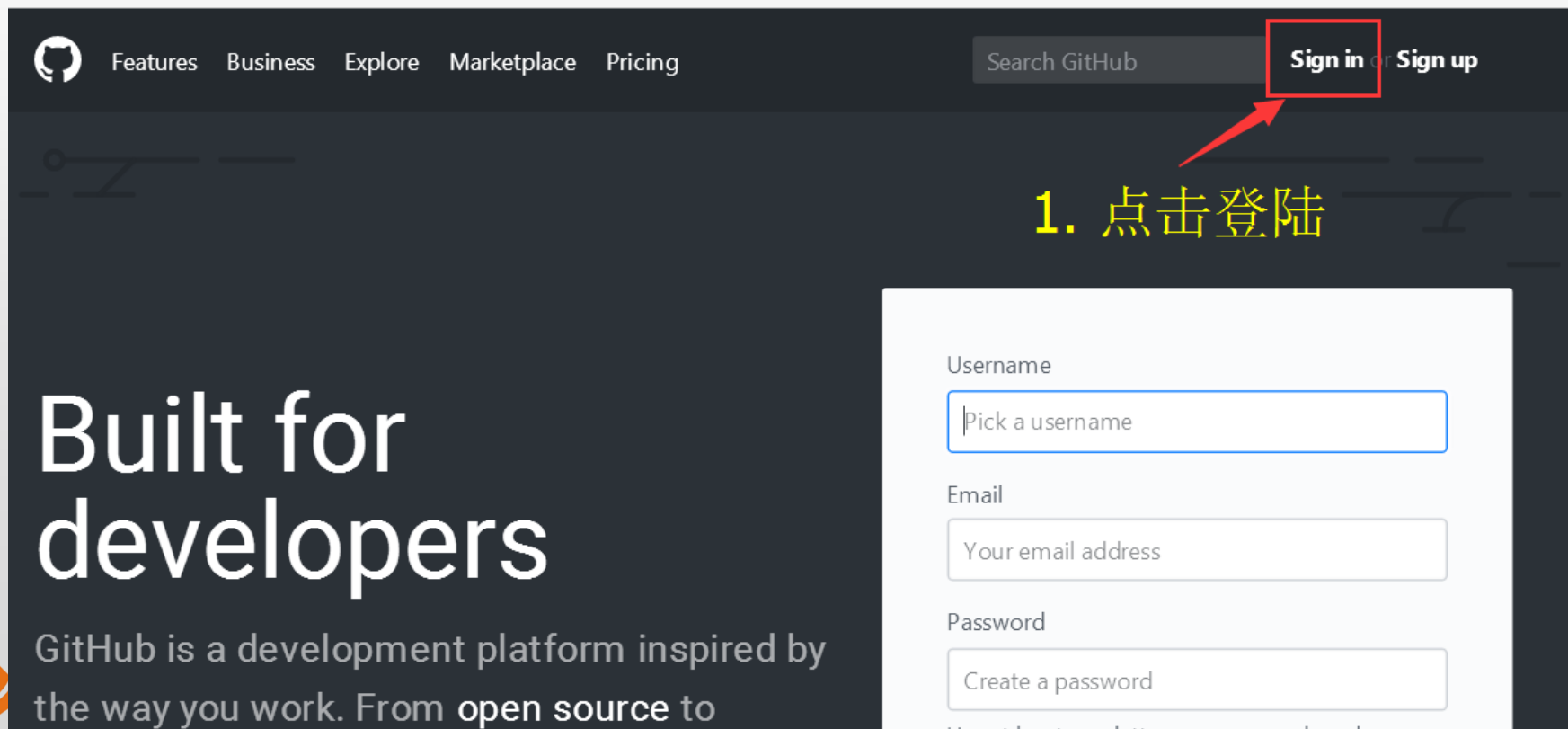
- 版本控制系统介绍
- Git 与 GitHub 简介
- Git账户注册与管理
- 工作区、本地仓库、远程仓库
- 创建版本库
- 修改管理

# Git账户注册与管理

- Git账户的注册
- 自定义账户头像
- 账户管理

# Git账户注册与管理

- Git账户的注册 ( <https://github.com/> )



The screenshot shows the GitHub homepage. The top navigation bar includes links for Features, Business, Explore, Marketplace, and Pricing. A search bar labeled 'Search GitHub' is present. The 'Sign in' button is highlighted with a red box, and a red arrow points to it from the text '1. 点击登陆' (1. Click login). Below the arrow, the login form is visible, containing fields for Username, Email, and Password.

1. 点击登陆

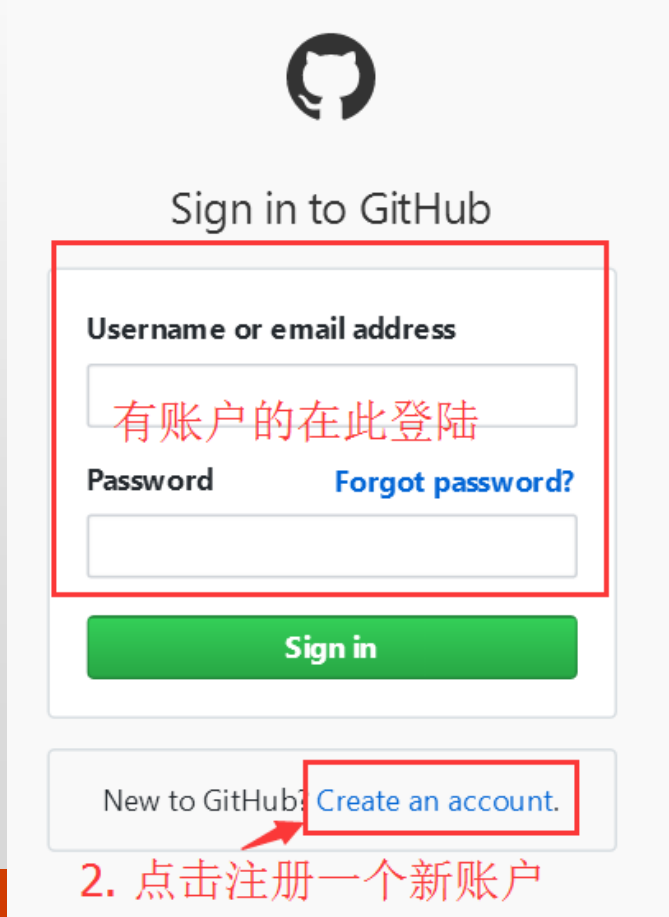
Username  
Pick a username

Email  
Your email address

Password  
Create a password

# Git账户注册与管理

- Git账户的注册 ( <https://github.com/> )



The image shows the GitHub login and registration page. At the top is the GitHub logo. Below it is the text "Sign in to GitHub". There are two main sections: a login section and a registration section. The login section is enclosed in a red box and contains a text input field for "Username or email address" with the red text "有账户的在此登陆" (Log in if you have an account) inside it, a text input field for "Password", and a blue link "Forgot password?". Below these is a green "Sign in" button. The registration section is below the login section and contains the text "New to GitHub?" followed by a blue link "Create an account." which is also enclosed in a red box. A red arrow points from the text "2. 点击注册一个新账户" (Click to register a new account) below the registration section to the "Create an account." link.

Sign in to GitHub

Username or email address

有账户的在此登陆

Password [Forgot password?](#)

Sign in


New to GitHub? [Create an account.](#)


2. 点击注册一个新账户




# Git账户注册与管理

## • Git账户的注册

**Step 1:**  
Create personal  
account

**Step 2:**  
Choose your plan

**Step 3:**  
Tailor your experience

### 3. 输入注册信息

#### Create your personal account

**Username**

输入用户名

This will be your username — you can enter your organization's username next.

**Email Address**

输入注册邮箱，不要使用163邮箱

You will occasionally receive account related emails. We promise not to share your email with anyone.

**Password**

输入密码

Use at least one lowercase letter, one numeral, and seven characters.

By clicking on "Create an account" below, you are agreeing to the [Terms of Service](#) and the [Privacy Policy](#).

Create an account

输入信息完成后，点击创建账户

#### You'll love GitHub

**Unlimited** collaborators

**Unlimited** public repositories

- ✓ Great communication
- ✓ Frictionless development
- ✓ Open source community

# Git账户注册与管理

## • Git

**Completed**  
Set up a personal account

**Step 2:**  
Choose your plan

**Step 3:**  
Tailor your experience

**Choose your personal plan**

☒ Unlimited public repositories for free.

☐ **Unlimited private repositories** for \$7/month.  
[\(view in CNY\)](#)

免费版，是项目公开的  
(默认选项)

非公开的项目是收费的

Don't worry, you can cancel or upgrade at any time.

☐ **Help me set up an organization next**  
Organizations are separate from personal accounts and are best suited for businesses who need to manage permissions for many employees.  
[Learn more about organizations.](#)

**Continue**

保持默认后，点击继续

**Both plans include:**

- ✓ Collaborative code review
- ✓ Issue tracking
- ✓ Open source community
- ✓ Unlimited public repositories
- ✓ Join any organization

# Git账户注册与管理

## • Git账户

✓ Completed  
Set up a personal account

🔌 Step 2:  
Choose your plan

⚙️ Step 3:  
Tailor your experience

定制自己的经验

How would you describe your level of programming experience?

☐ Very experienced    ☐ Somewhat experienced    ☐ Totally new to programming

描述编程经验

What do you plan to use GitHub for? (check all that apply)

使用GitHub的目的

☐ School projects    ☐ Research    ☐ Design

☐ Development    ☐ Project Management    ☐ Other (please specify)

Which is closest to how you would describe yourself?

对自己的身份定位

☐ I'm a hobbyist    ☐ I'm a student    ☐ I'm a professional

☐ Other (please specify)

What are you interested in?    感兴趣内容

e.g. tutorials, android, ruby, web-development, machine-learning, open-source

本页内容可不填写，点击跳过当前步骤  
若填写内容，点击“Submit”提交

# Git账户注册与管理

- Git账户的注册 ( <https://github.com/> )

4. 注册成功，但必须先去做邮箱验证

Learn Git and GitHub without any code!

Using the Hello World guide, you'll create a repository, start a branch, write comments, and open a pull request.

Read the guide

Start a project

点击阅读“向导”

开启一个新项目

Hi @

Help us secure your GitHub account by verifying your email address ( ). This lets you access all of GitHub's features.

[Verify email address](#)

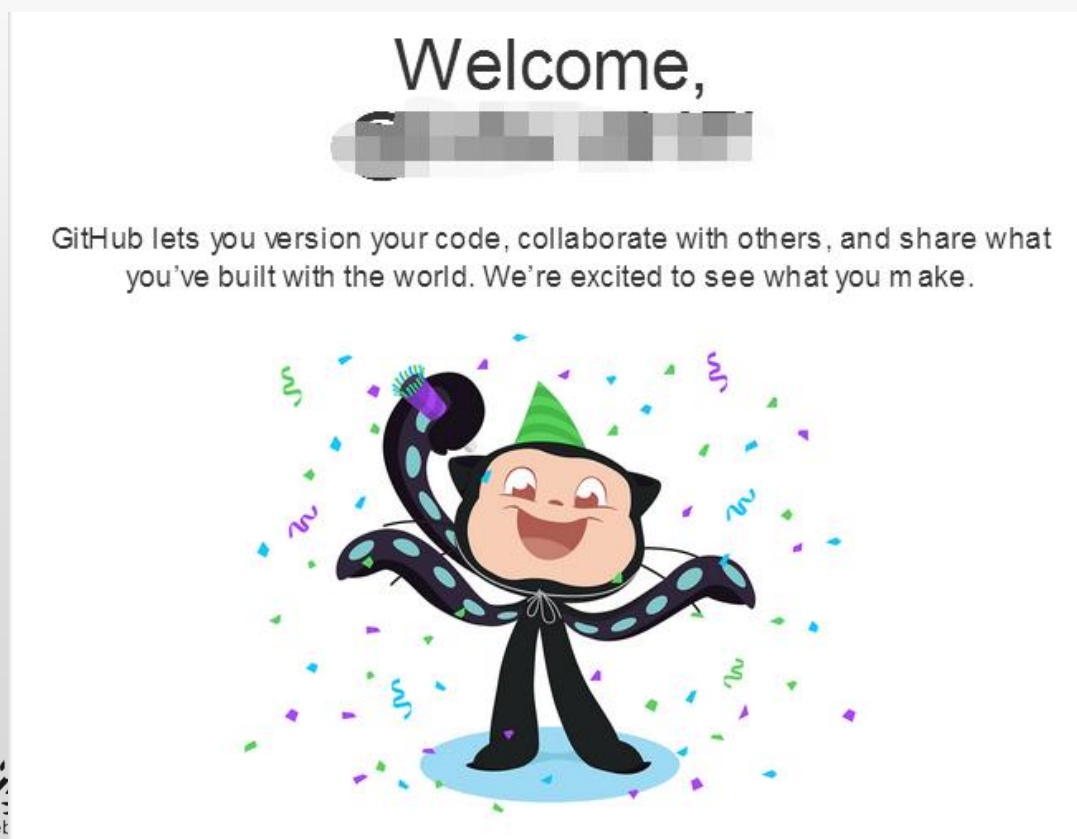
点击验证邮箱

Button not working? Paste the following link into your browser:

[https://github.com/users/lnfor-2017/emails/32900484/confirm\\_verification/8910058123c4530f69bb9dc4f3664d225fdb310](https://github.com/users/lnfor-2017/emails/32900484/confirm_verification/8910058123c4530f69bb9dc4f3664d225fdb310)

# Git账户注册与管理

- Git账户的注册 ( <https://github.com/> )

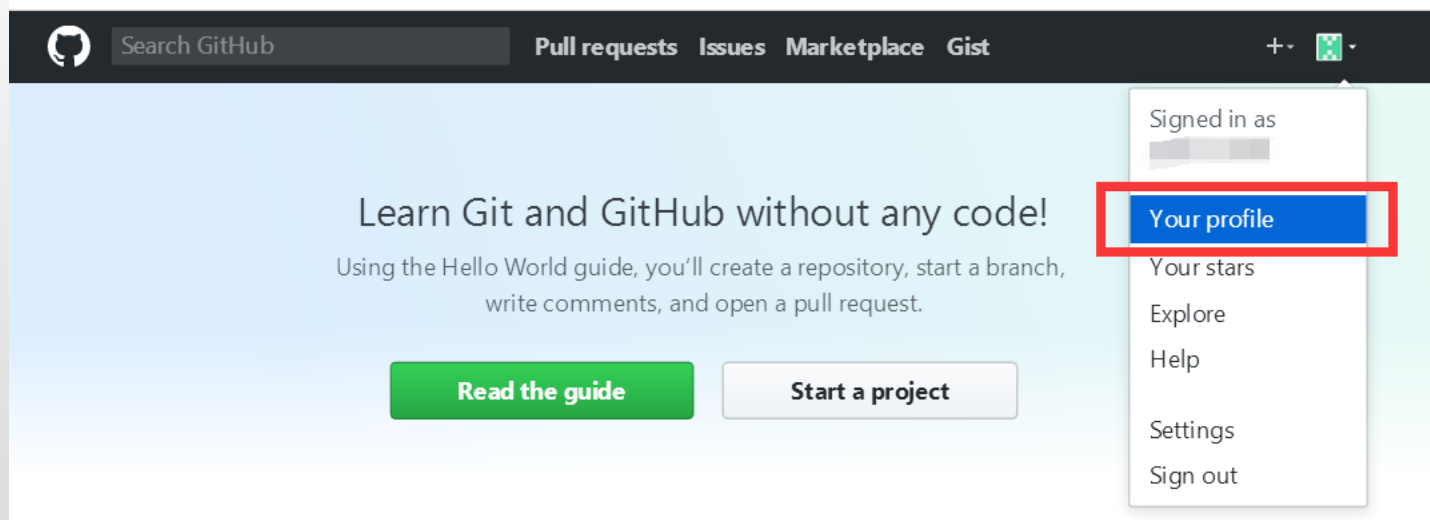


# Git账户注册与管理

- 查看向导页面（GitHub基本操作）
  - What is GitHub?
  - Step 1. Create a Repository
  - Step 2. Create a Branch
  - Step 3. Make and commit changes
  - Step 4. Open a Pull Request
  - Step 5. Merge your Pull Request

# Git账户注册与管理

- 自定义账户头像



# Git账户注册与管理

- 账户首页





# Git账户注册与管理

- GitHub默认头像
  - Github给无头像用户生成5x5像素的Identicons头像
  - 使用用户ID的哈希值，然后根据哈希值每一位的奇偶值来决定对应位置上的像素的开关。这样生成的图像，配上由哈希值决定的颜色，保证可生成大量唯一的图像



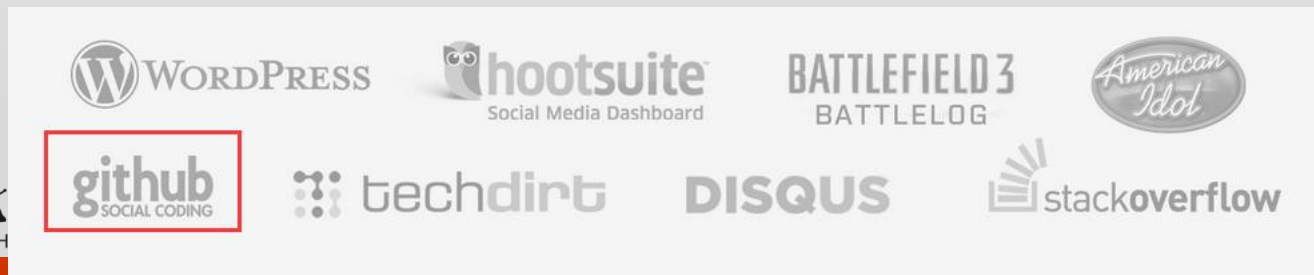
# Git账户注册与管理

- 自定义账户头像
  - 使用Gravatar关联头像与邮箱

# Git账户注册与管理

- Gravatar简介

- Globally Recognized Avatar ( 全球识别头像 )
- <http://www.gravatar.com>
- 提供的在线头像服务，主要用在博客和论坛留言中，依靠邮件地址识别判断，为每个用户展示个性化的通用头像服务，完全免费

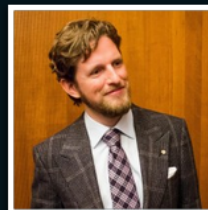


# Git账户注册与管理

- 自定义账户头像

- 1. 注册Gravatar账户 ( <http://www.gravatar.com> )

Your Gravatar is an image that follows you from site to site appearing beside your name when you do things like comment or post on a blog. Avatars help identify your posts on blogs and web forums, so why not on any site?



**Matt**

embarcadero and townsend, sf ca

In 2002 I started contributing to Open Source software, and life has just gotten better from there. Co-founder of WordPress, founder Automattic.

[View Complete Profile](#)

Introducing Gravatar

NAME

点击注册账户

Create Your Own Gravatar

# Git账户注册与管理

WordPress.com 也可以在 [简体中文](#) 中使用



Signing up for **Gravatar** with WordPress.com

[I already have a WordPress.com account!](#)

Not sure what this is all about?  
[We can help clear that up for you.](#)

E-mail Address

在GitHub注册的邮箱 ✓

**Triple-check** your email. It's the only way we can contact you.

Username

自动提取用户名，可自定义 ✓

This is what we'll call you. It needs to be a least four letters or numbers.

Password

..... 输入密码

Don't be afraid to use symbols like **!"£\$%^&()** along with numbers and letters.

You agree to the [fascinating terms of service](#) by submitting this form.

点击创建

Sign up →

不接受中文

# Git账户注册与管理

- 自定义账户头像
  - 2. 邮件确认

## We sent you a confirmation email!

*We sent an email to yang\_xin\_jia@126.com when you signed up for Gravatar.*

In this email you will find a link that, when clicked, will bring you back to a confirmation page. Once you've confirmed your email, you can start using Gravatar.

If for some reason you did not receive the email, you can click the link to resend the confirmation email. You've now sorted!

## Welcome to WordPress.com (and Gravatar)!

You created a WordPress.com account with your Gravatar sign up. Gravatar is just one of several glorious doodads brought to you by the jolly people at Automattic. Click the button below to activate your account.

[Activate Account](#)

Your new WordPress.com account also gives you access to our other fabulous contraptions, which include [Polldaddy](#), and [Akismet](#), [VaultPress](#)

# Git账户注册与管理

- 自定义账户头像

- 3. 登陆账户

从邮箱登陆



从网站登陆



Thanks for confirming your email address.

You created your WordPress.com account while signing up for Gravatar, so you'll likely want to head back there...

Sign in to Gravatar

点击登陆



What is Gravatar?

How to Use Gravatar

Help



WordPress.com

Sign In

## We sent you a confirmation email!

*We sent an email to yang\_xin\_jia@126.com when you signed up for Gravatar.*

In this email you will find a link that, when clicked, will bring you back to a confirmation page. Once you've confirmed your email, you can start using Gravatar.


# Git账户注册与管理

- 自定义账户头像
- 4. 关联头像

My Gravatars My Profile Help 126.com

## Manage Gravatars

1. Pick email to modify

 @126.com  
Primary address (cannot delete)

[Add email address](#)

**Whoops, looks like you don't have any images yet. [Add one by clicking here!](#)**

If you don't assign a Gravatar to your email address, a custom one will be dynamically generated for you based on the settings of the website it appears on.

To use your own custom image, click the link above and then upload one, provide a link to something online or take a picture using your webcam.

[Add a new image](#)

点击给当前邮箱关联一个头像



# Git账户注册与管理

- 自定义账户头像
  - 4. 关联头像

## Upload image



# Git账户注册与管理

- 自定义账户头像
  - 4. 关联头像

## Upload new image

We will automatically resize large images, but please note that there is a 1MB file size limit.

You will have a chance to crop this image in the next step.



# Git账户注册与管理

- 自定义账户头像
  - 4. 关联头像

Choose a rating for your Gravatar



选择头像级别：  
G 普通级、PG 辅导级  
R 和 X 为限制级


By clicking on one of these ratings

选择G即可



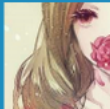
# Git账户注册与管理

- 自定义账户头像
  - 5. 头像设置成功

 My Gravatars My Profile Help

## Manage Gravatars


1. Pick email to modify



yang\_xin\_jia@126.com  
Primary address (cannot delete)  
[Remove image](#)  
[View rating](#)

[Add email address](#)

2. Pick new image



rated G X

[Add a new image](#)

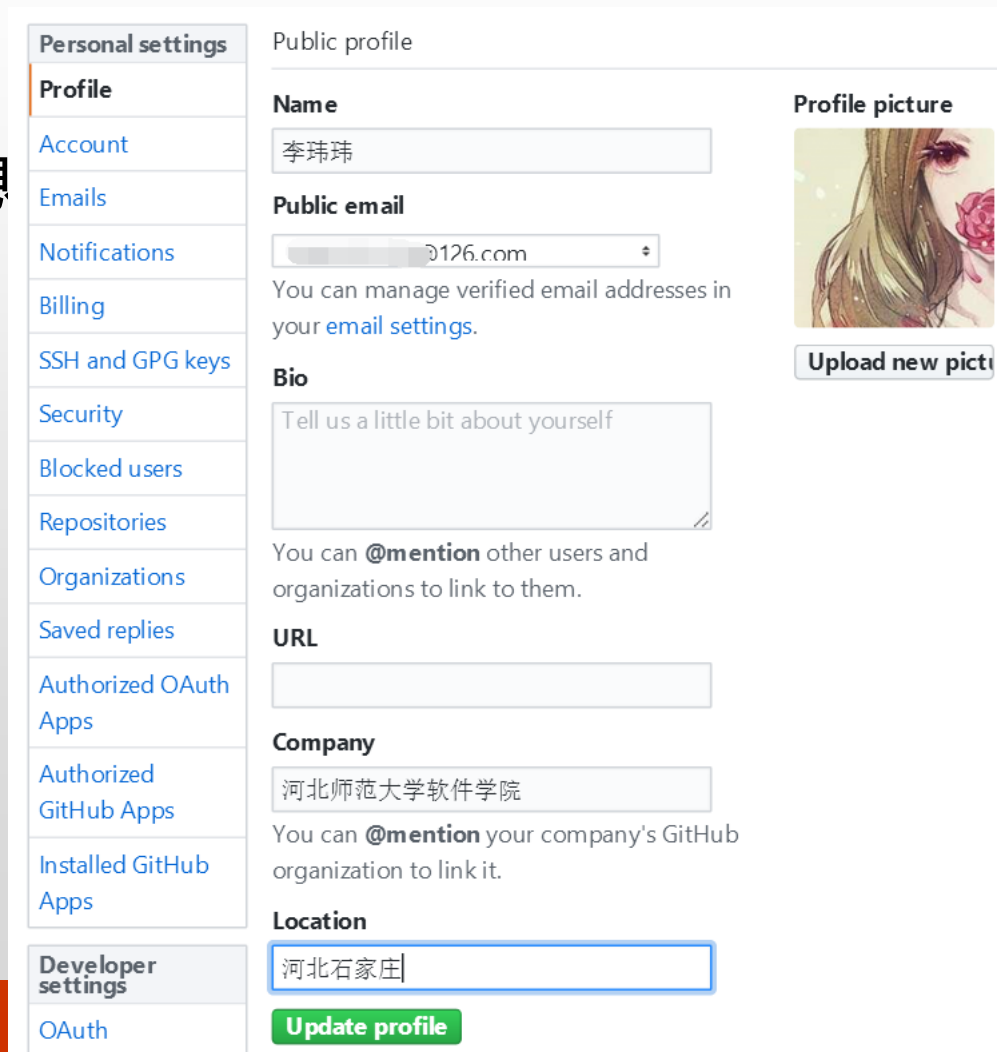
# Git账户注册与管理

- 自定义账户头像
  - 6. 查看GitHub



# Git账户注册与管理

- 自定义账户头像
  - 6. 更新GitHub个人信息



**Personal settings**

- Profile**
- Account
- Emails
- Notifications
- Billing
- SSH and GPG keys
- Security
- Blocked users
- Repositories
- Organizations
- Saved replies
- Authorized OAuth Apps
- Authorized GitHub Apps
- Installed GitHub Apps

**Developer settings**

- OAuth

**Public profile**

**Name**

李玮玮

**Public email**

126.com

You can manage verified email addresses in your [email settings](#).

**Bio**

Tell us a little bit about yourself

You can **@mention** other users and organizations to link to them.

**URL**

**Company**

河北师范大学软件学院

You can **@mention** your company's GitHub organization to link it.

**Location**

河北石家庄

**Profile picture**

Upload new picture

**Update profile**



# 目 录

- 版本控制系统介绍
- Git 与 GitHub 简介
- Git账户注册与管理
- 工作区、本地仓库、远程仓库
- 创建版本库
- 修改管理

# 工作区、本地仓库、远程仓库

- Git管理中文件的三种状态
  - 对于任何一个文件，在Git内部都只有三种状态：已修改，已暂存和已提交。
  - 已修改：表示修改了某个文件，但还没有提交保存
  - 已暂存：表示把已修改的文件放在下次提交时要保存的清单中了
  - 已提交：表示该文件已被安全的保存在本地数据库中了

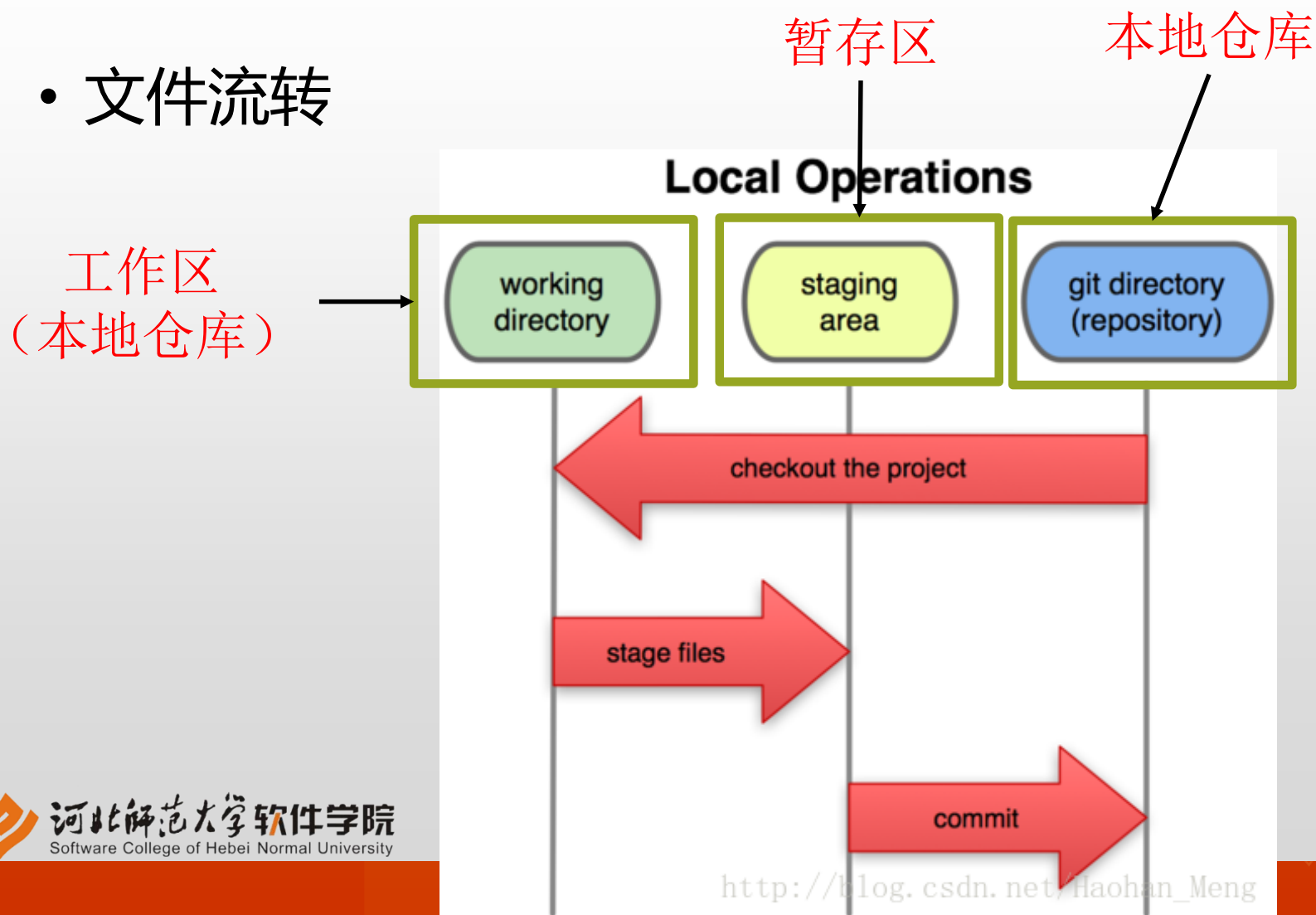


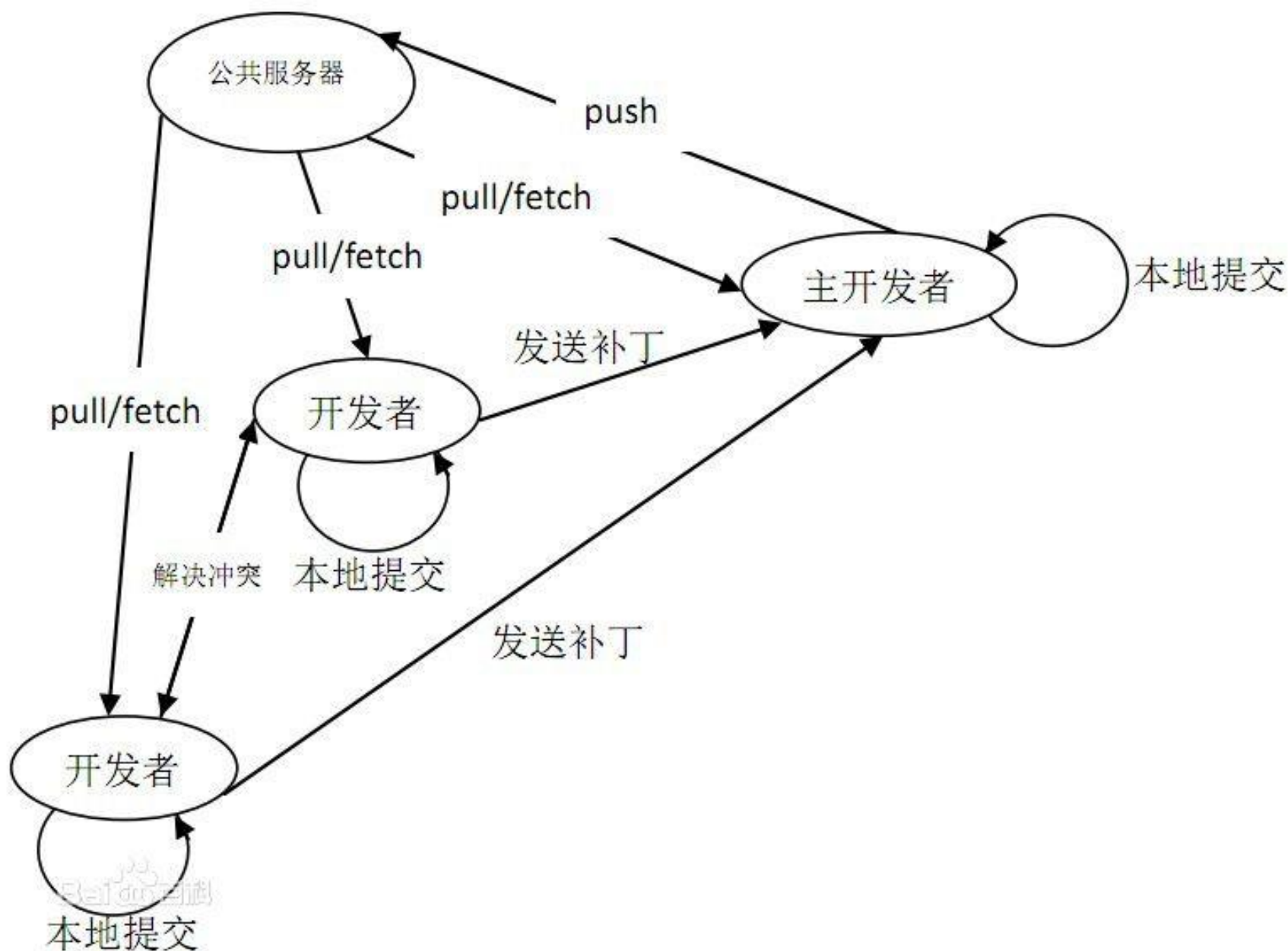
# 工作区、本地仓库、远程仓库

- Git在管理项目时，**文件流转**的三个工作区域是：  
本地仓库(即工作目录，也就是项目的源文件)->暂存区域->Git的工作目录
  - 在本地的的工作目录修改某些文件；
  - 对修改后的文件进行快照，保存到暂存区域；
  - 提交更新，将保存在暂存区域中的文件快照永久转存到Git的工作目录中

# 工作区、暂存区、远程区

- 文件流转







# 目 录

- 版本控制系统介绍
- Git 与 GitHub 简介
- Git账户注册与管理
- 工作区、本地仓库、远程仓库
- 创建版本库
- 修改管理

# 创建版本库

- 1. Git 安装 ( Windows )
- 2. Git 操作

# 创建版本库

- 1. Git 安装 ( Windows )
  - 下载地址
    - <https://git-for-windows.github.io/>
    - <https://pan.baidu.com/s/1kU5OCOB?errno=0&errmsg=Auth%20Login%20Sucess&&bduss=&ssnerror=0#list/path=%2Fpub%2Fgit>

# 创建版本库

## • 2. Git 操作

### • 操作方式

- 小乌龟工具：TortoiseGit

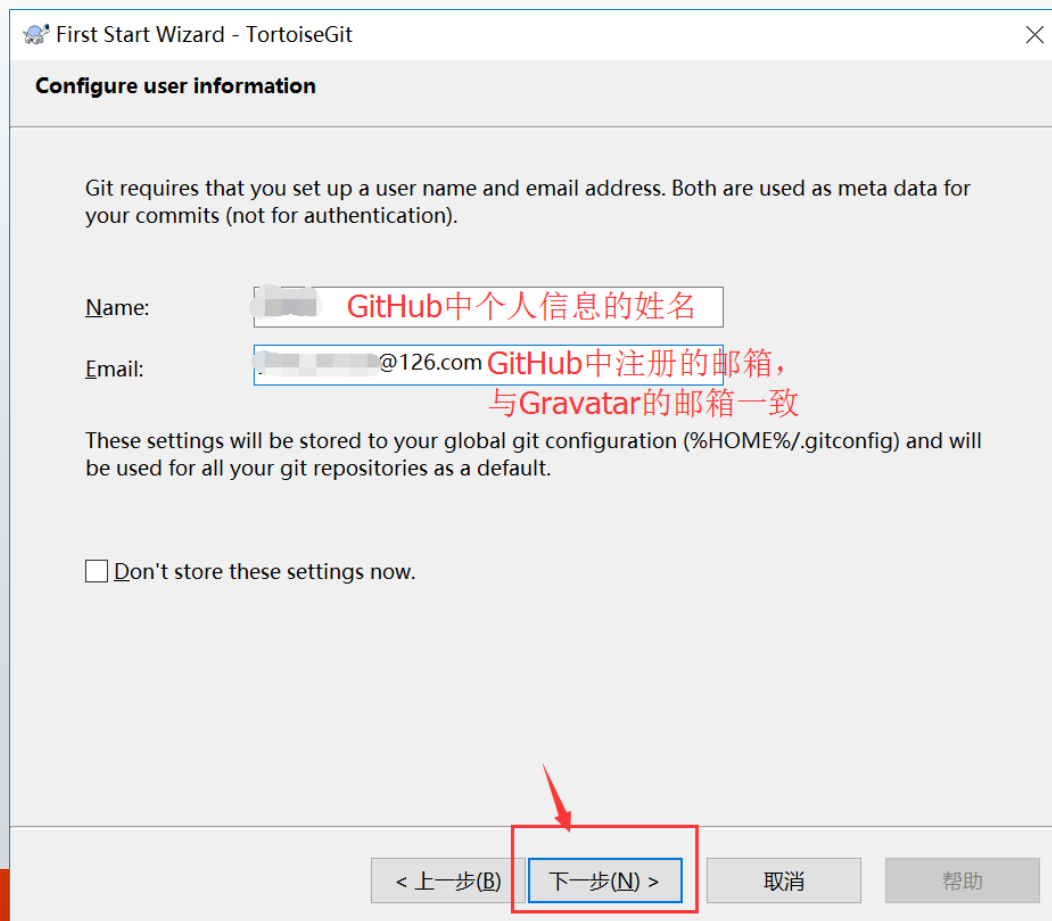
<https://tortoisegit.org/download/>

- 命令行：Git Bash
- 图形界面：Git GUI



# 创建版本库

## • 小乌龟工具 ( TortoiseGit ) 安装关键步骤说明



First Start Wizard - TortoiseGit

**Configure user information**

Git requires that you set up a user name and email address. Both are used as meta data for your commits (not for authentication).

Name:  GitHub中个人信息的姓名

Email:  @126.com GitHub中注册的邮箱，  
与Gravatar的邮箱一致

These settings will be stored to your global git configuration (%HOME%/.gitconfig) and will be used for all your git repositories as a default.

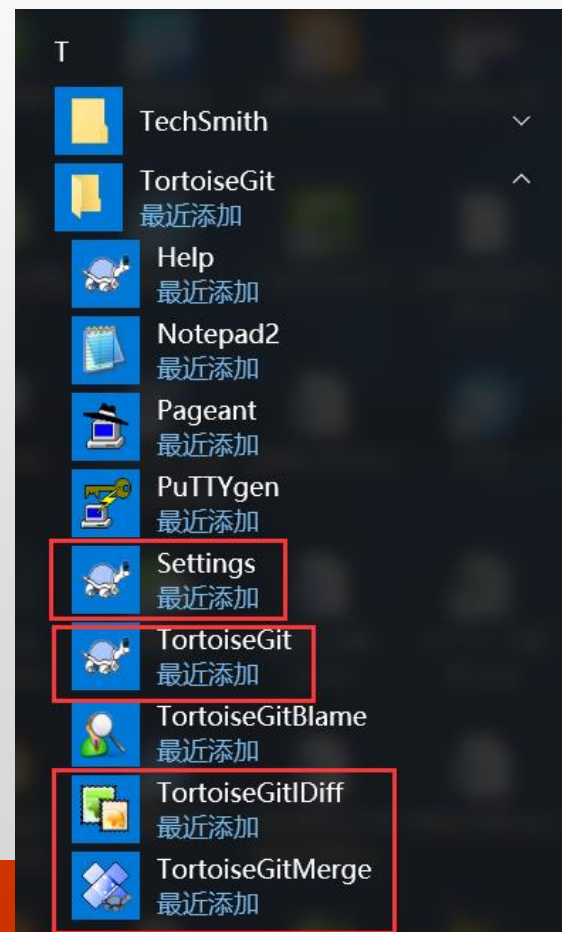
☐ Don't store these settings now.

< 上一步(B)   **下一步(N) >**   取消   帮助



# 创建版本库

- 小乌龟工具 ( TortoiseGit ) 安装
- 功能：
  - 设置
  - 维护项目
  - 管理版本区别
  - 冲突合并



# 创建版本库

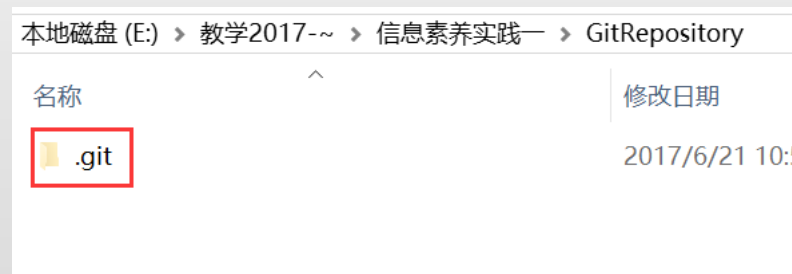
- 小乌龟工具 ( TortoiseGit ) 使用
  - 新建
  - 克隆

# 创建版本库

- 小乌龟工具 ( TortoiseGit ) 使用

- 新建

鼠标  
右键



# 创建版本库

- 小乌龟工具 ( TortoiseGit ) 使用
  - 新建
    - 2 ) 在GitHub上创建仓库



李玮玮



# 创建版本库

Owner

 Infor-2017 ▾

Repository name

InfoTest

自定义一个仓库的名字 ✓

Great repository names are short and memorable. Need inspiration? How about **curly-doodle**.

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.

生成readme文件

Add .gitignore: None ▾

Add a license: None ▾



Create repository

创建仓库



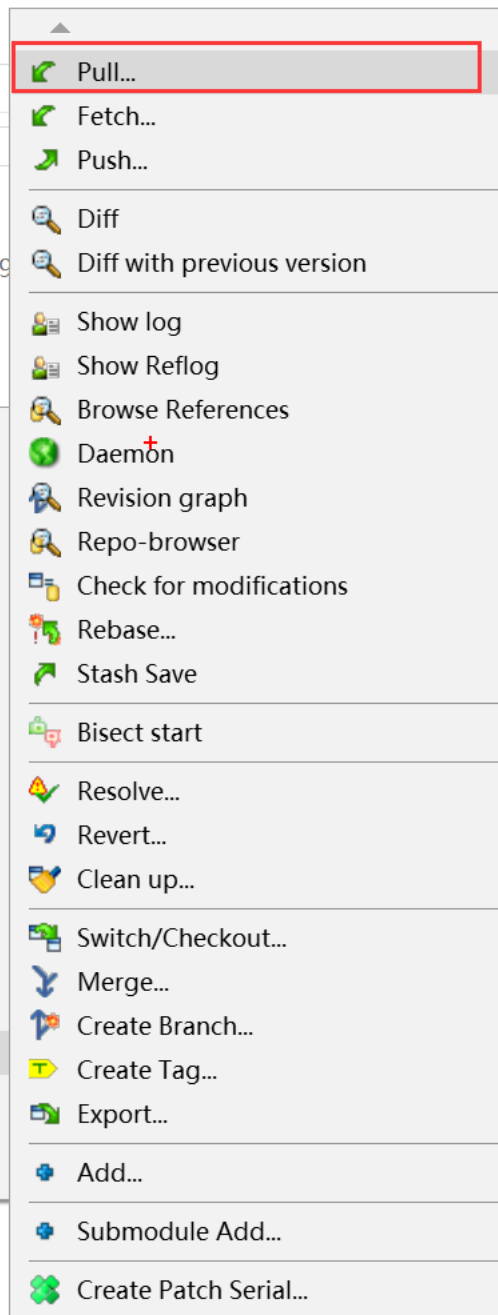
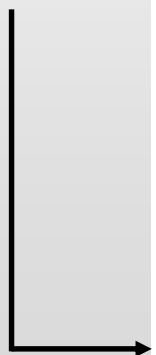
# 创建片

## • 小乌龟

### • 新建

### • 4

鼠标  
右键



E:\教学2017-~\信息素养实践一\copy - Settings - TortoiseGit



- General
  - Context Menu
  - Set Extend Menu Item
  - Dialogs 1
  - Dialogs 2
  - Dialogs 3
  - Colors 1
  - Colors 2
  - Colors 3
  - Alternative editor
- Git
  - Remote
  - Credential
- Hook Scripts
  - Issue Tracker Integration
  - Issue Tracker Config
- Icon Overlays
  - Icon Set
  - Overlay Handlers
- Network
  - Email
- Diff Viewer
  - Merge Tool
- Saved Data
- TortoiseGitBlame
- TortoiseGitUDiff
- Advanced

Remote

Remote:

origin

Remote:

URL:

Push URL:

Putty Key:

Tags:  ☐ Push Default

☒ Prune ☐ Prune (All remotes)

1 粘贴远程仓库地址

3

2

确定

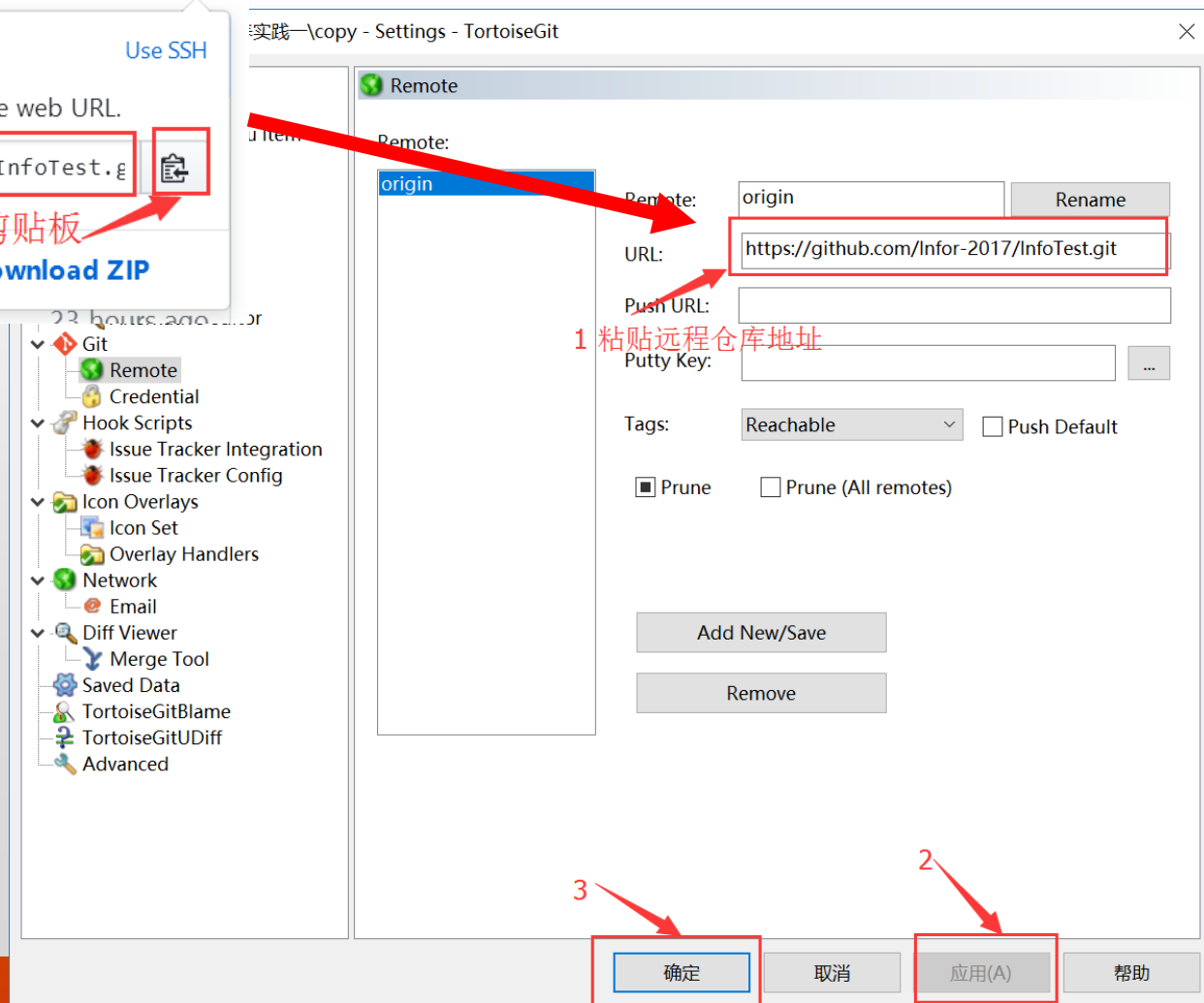
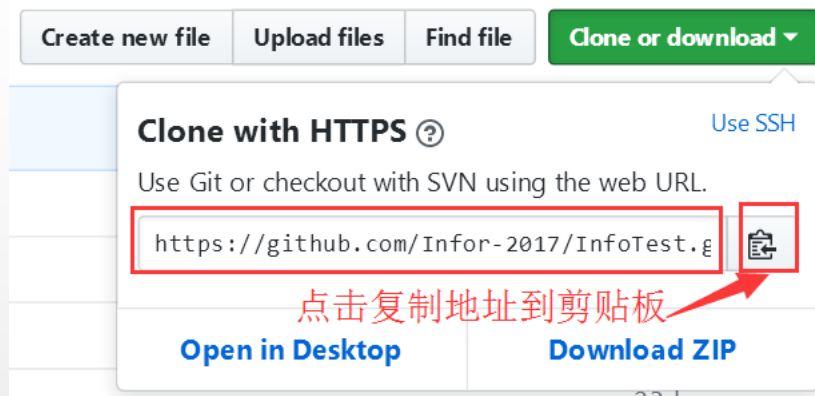
取消

应用(A)

帮助



# 创建版本库

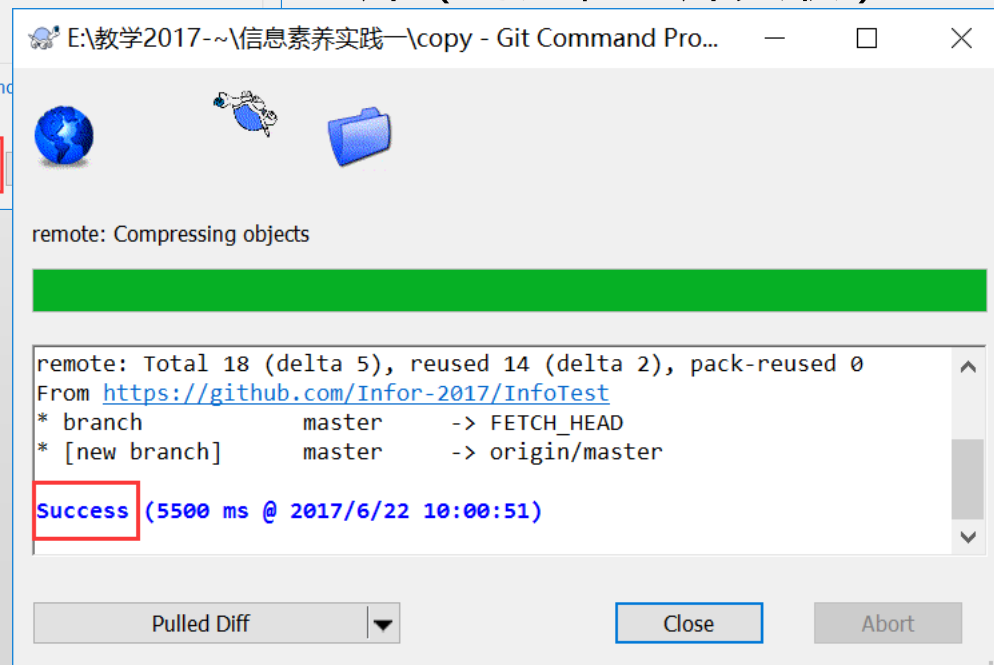
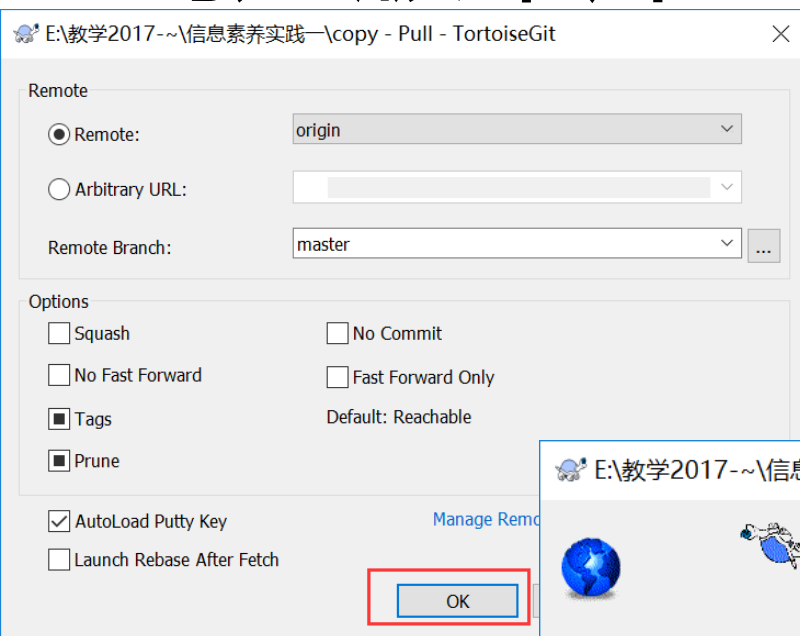


# 创建版本库

iseGit ) 使用

程仓库 ( 与远程仓库关联 )

- .git
- test
- 新增目录
- 20170622.txt
- developer.txt
- README.md
- 新增文件.txt

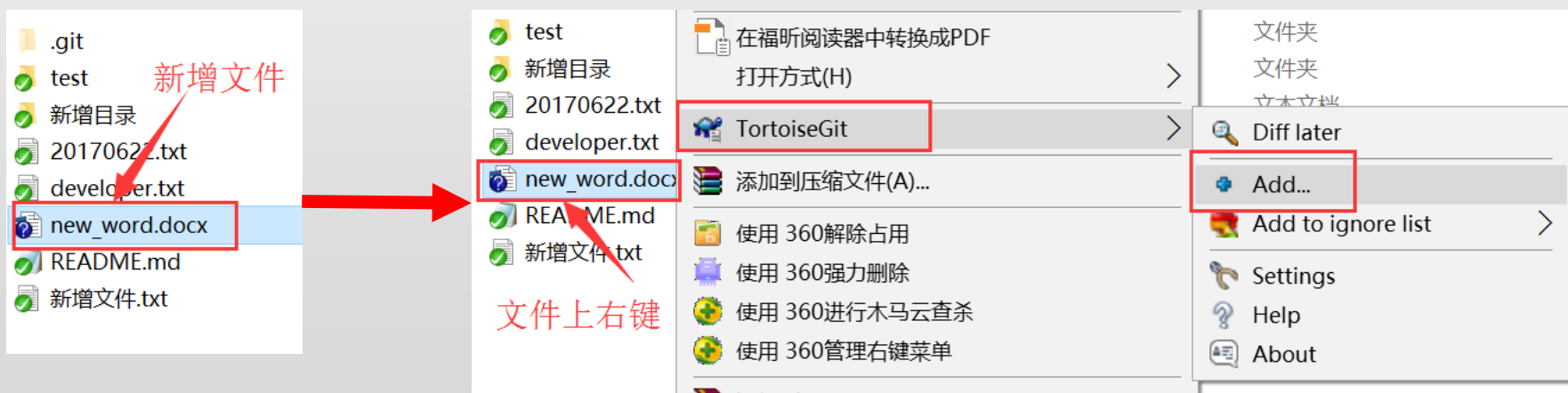


# 创建版本库

- 小乌龟工具 ( TortoiseGit ) 使用

- 新建

- 5 ) 提交文件 ( 先添加<add>到文件 , 再提交<commit>到本地仓库 , 最后推<push>到远程仓库 )



Add Finished!

Action	Path
Command	Add
Added	new_word.docx
Finished!	Success (31 ms @ 2017/6/22 10:45:56)

文件已添加

地仓库，

本地磁盘 (E:) > 教学2

名称

- .git
- test
- 新增目录
- 20170622.txt
- developer.txt
- new\_word.docx
- README.md
- 新增文件.txt

文件上的图标变化，  
说明已经添加到  
工作区

E:\教学2017-~\信息素养实践一\copy - Commit - TortoiseGit

Commit to: master ☐ new branch

Message:

添加word文档

1 必须填写提交说明

☐ Amend Last Commit  
☐ Set author date  
☐ Set author

Add Signed-off-by

Changes made (double-click on file for diff):

Check: **All** None Unversioned **Versioned** Added Deleted Modified **Files** Submodules

Path	Extension	Status	Lines added	Lines removed
<input checked="" type="checkbox"/> new_word.docx	docx	Added	0	0

勾选要提交到本地仓库的文件

☒ Show Unversioned Files  
☐ Do not autoselect submodules

2 点击提交

☐ Show Whole Project  
☐ Message only

Commit

Cancel

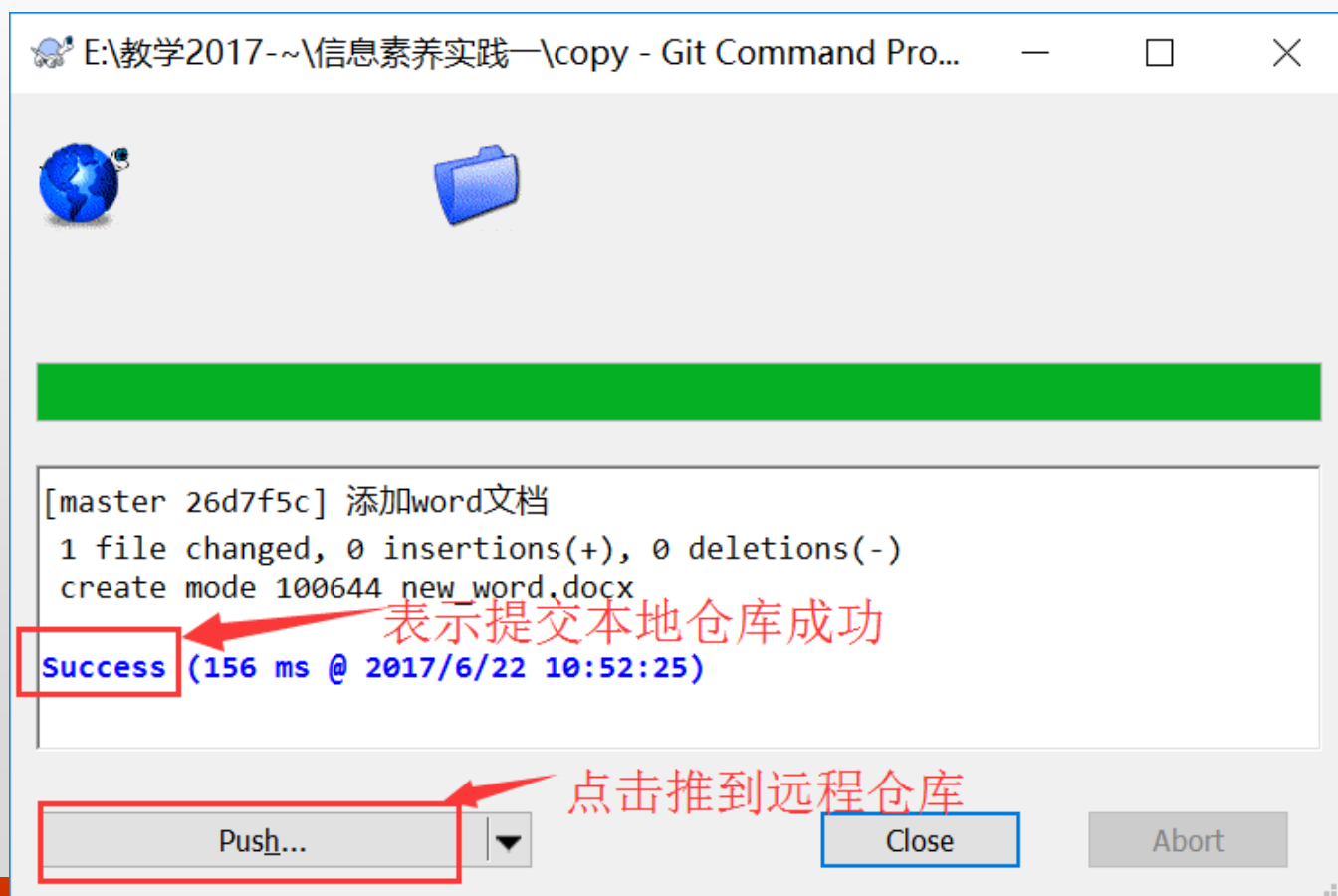
Help

1 files selected, 1 files total

[View Patch>>](#)

# 创建版本库

- 小乌龟工具 ( TortoiseGit ) 使用



git>到本

# 创建版本库

(Git) 使用

The image shows a sequence of steps for pushing code to a remote repository using TortoiseGit. The background window is the 'Push' dialog in TortoiseGit, with 'Remote:' set to 'developer' and 'Destination' set to 'Remote: origin'. A red box labeled '1' highlights the 'Remote:' field. In the foreground, the 'Git Command Prompt' window shows the execution of the push command. The output indicates that the push was successful, with a green bar and the text 'Success (173015 ms @ 2017/6/22 10:58:07)'. A red box labeled '2' highlights the 'OK' button in the TortoiseGit dialog. Another red box labeled '成功' (Success) points to the success message in the command prompt. A red arrow points from the 'Close' button in the command prompt to the 'OK' button in the TortoiseGit dialog. To the right, two 'Password' dialog boxes are shown, with red boxes and text indicating where to enter the GitHub username and password.

Ref

☐ Push all branches

Local: master

Remote: developer

Destination

☒ Remote: origin

☐ Arbitrary URL: 1

Options

Force: May discard

☐ Use Thin Pack (For slow network)

☐ Include Tags

☒ Autoload Putty Key

☐ Set upstream/track remote branch

☐ Always push to the selected remote branch

☐ Always push to the selected remote branch for this local branch

Recurse submodule

2

OK

Cancel

Help

remote: Resolving deltas

Total 2 (delta 1), reused 0 (delta 0)

remote: Resolving deltas: 100% (1/1), completed with 1 local object.

To <https://github.com/Infor-2017/InfoTest.git>

d4a4a23..26d7f5c master -> developer

Success (173015 ms @ 2017/6/22 10:58:07)

成功

Create pull request

Close

Abort

输入GitHub的用户名

Cancel

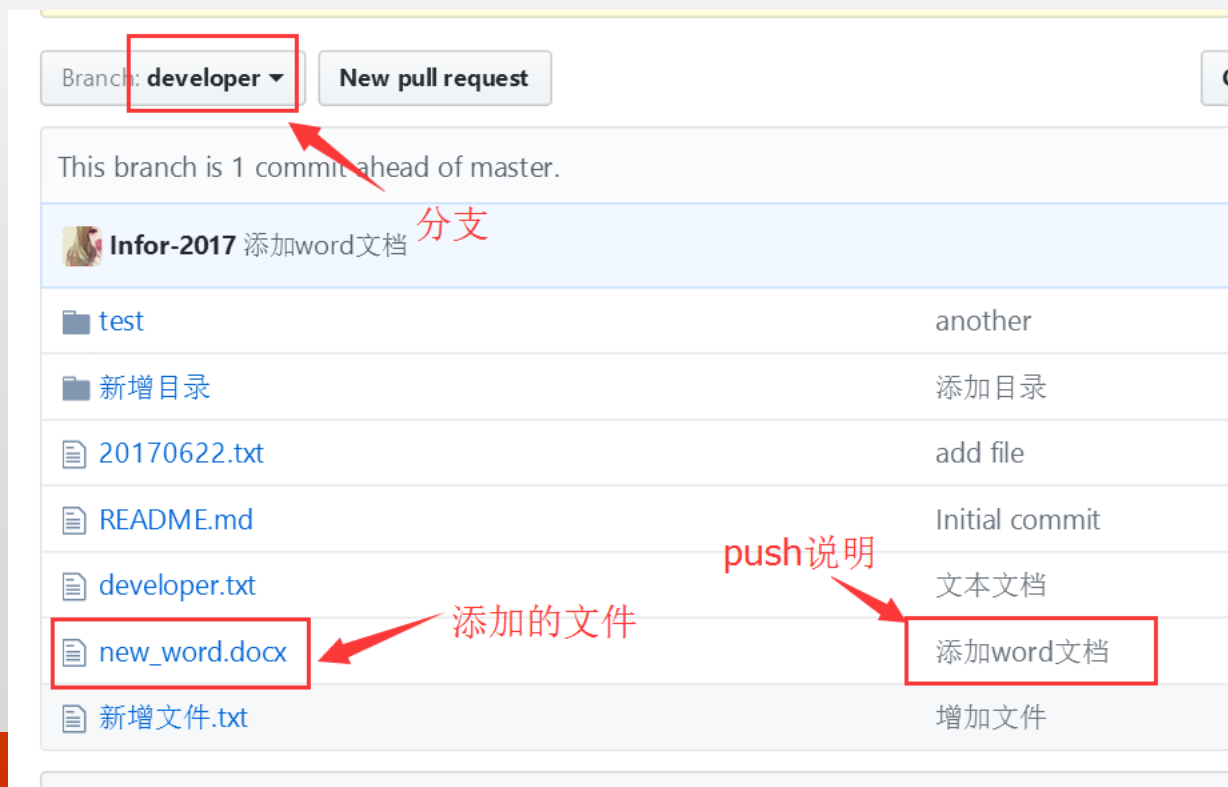
输入GitHub的密码

OK

Cancel

# 创建版本库

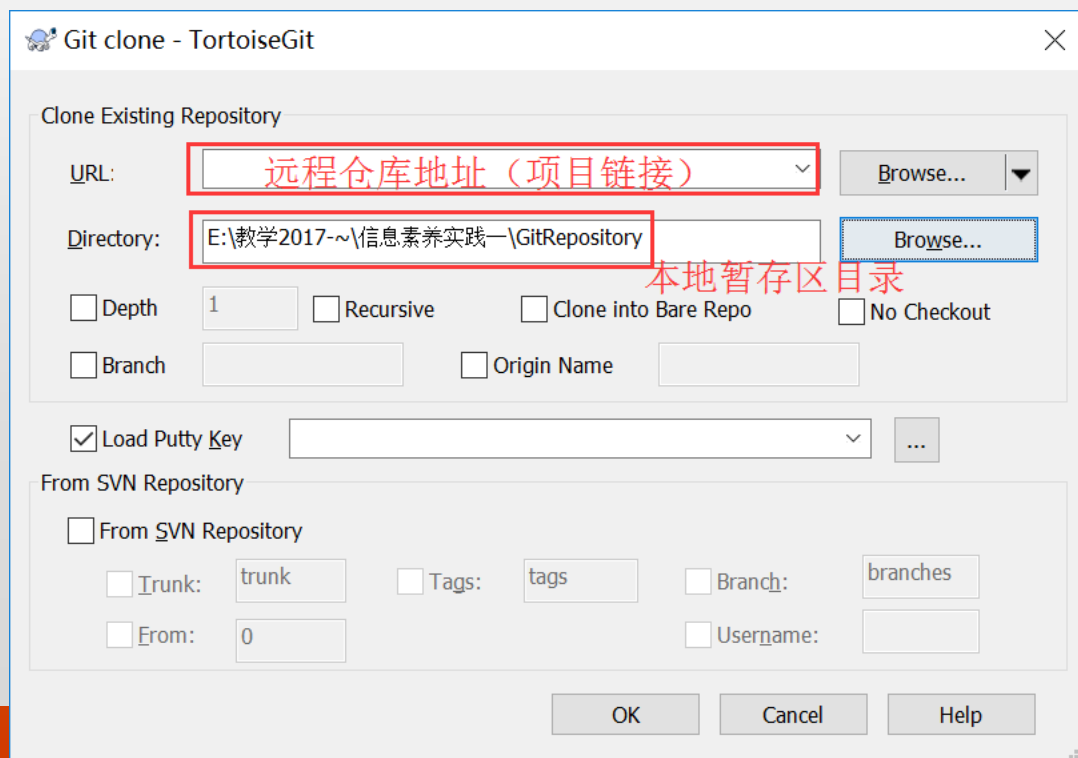
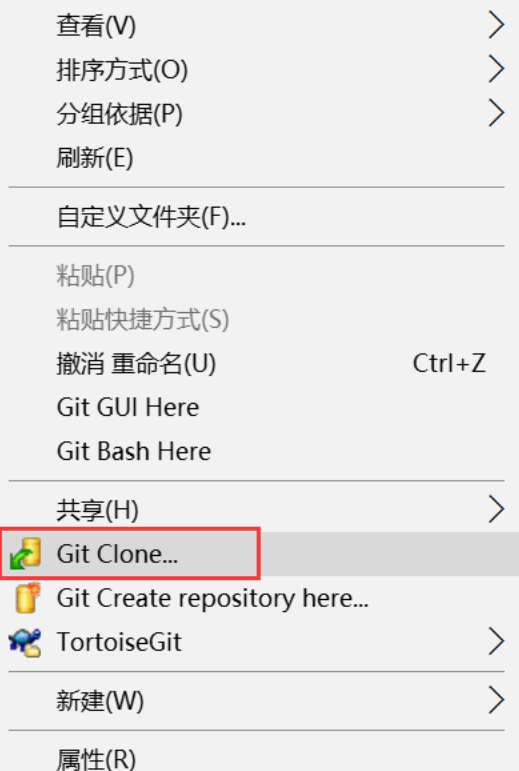
- 小乌龟工具 ( TortoiseGit ) 使用
  - 查看GitHub内容变化



# 创建版本库

- 小乌龟工具 ( TortoiseGit ) 使用
  - 克隆

鼠标  
右键

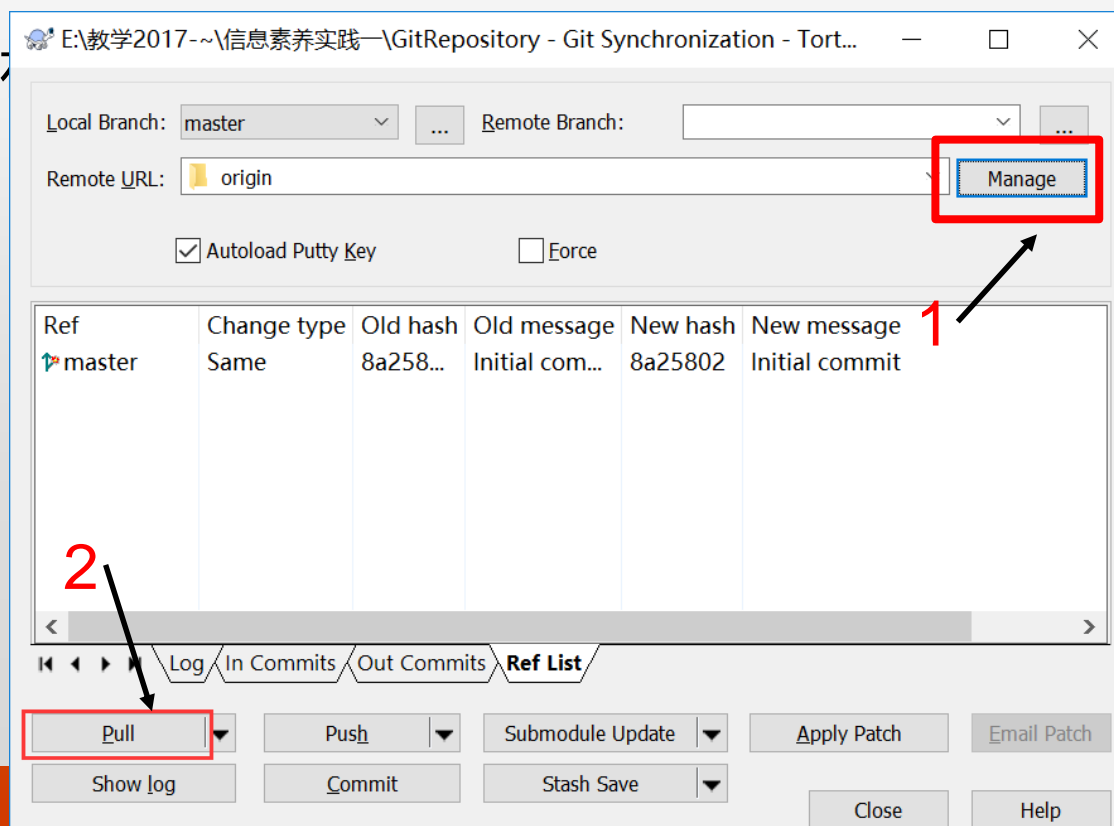




# 创建版本库

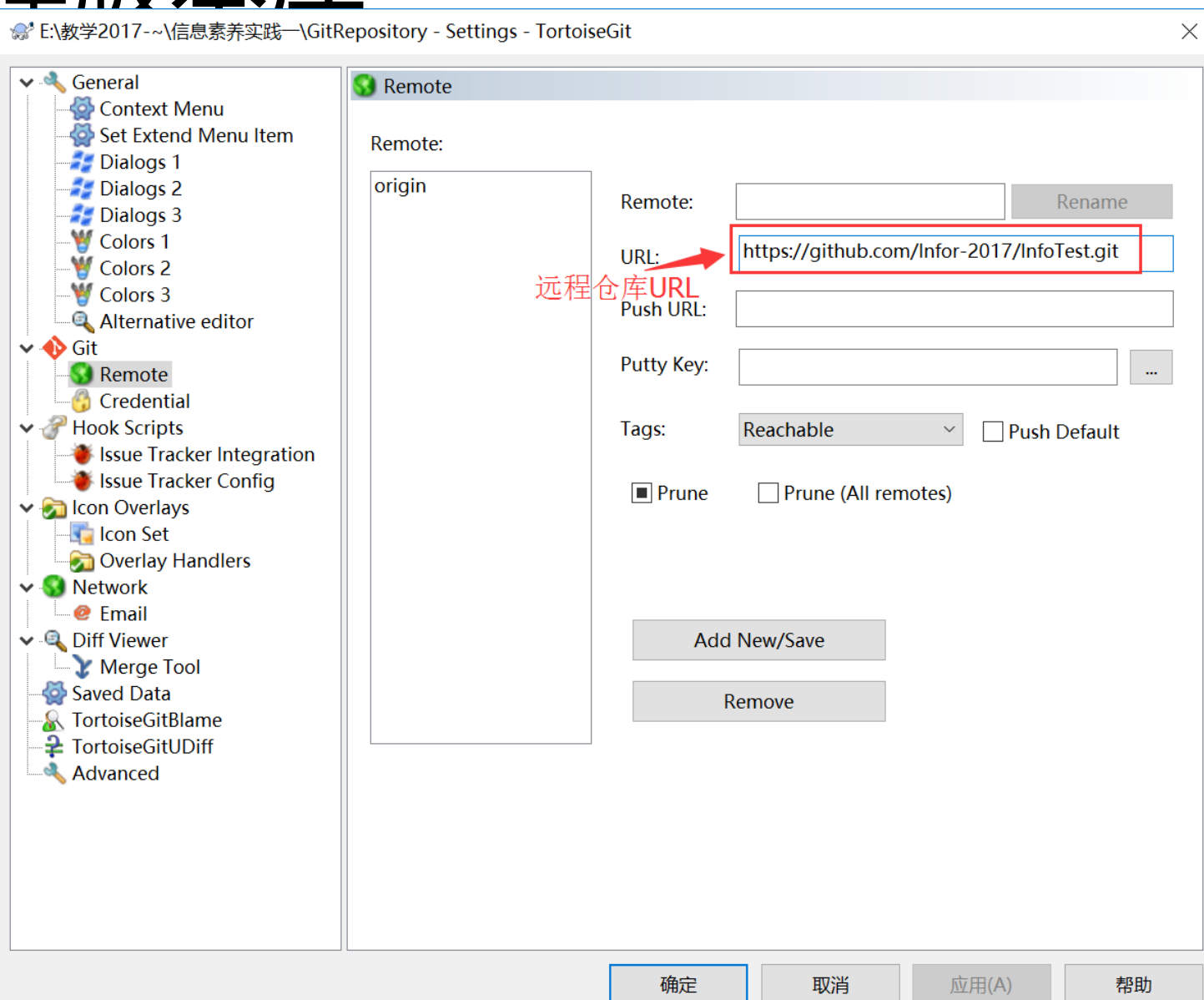
- 小乌龟工具 ( TortoiseGit ) 使用
  - 克隆
    - 若本地已经新建暂

鼠标  
右键



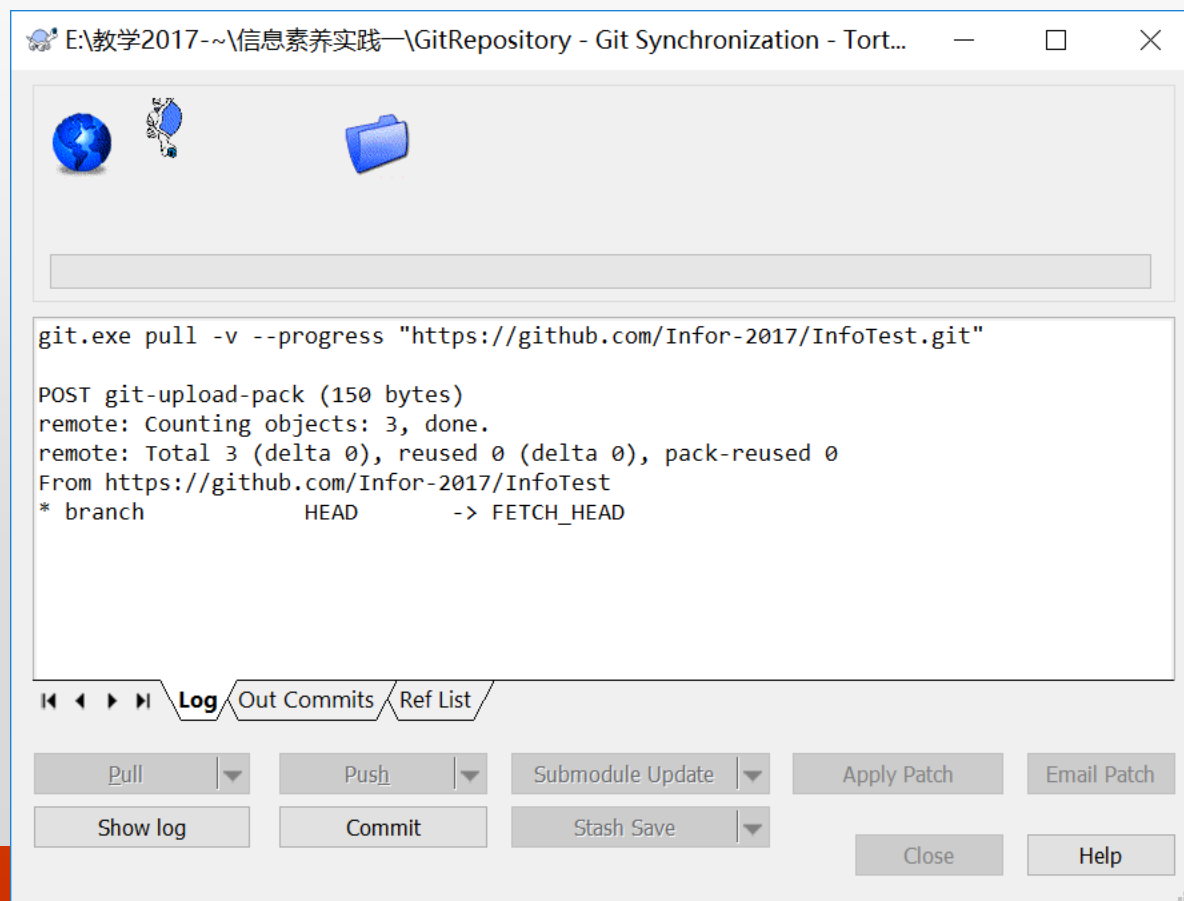
# 创建版本库

• 点



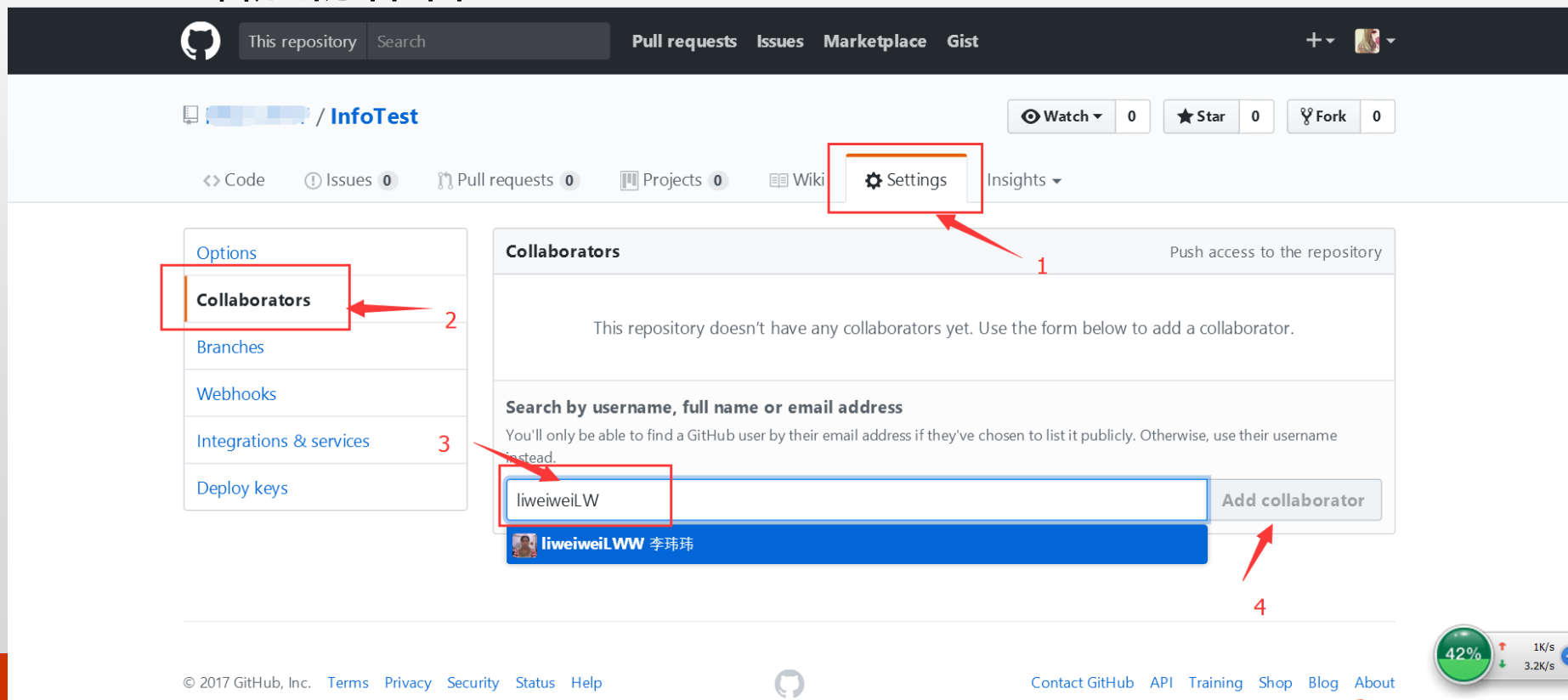
# 创建版本库

- 点击Pull之后的对话框



# 创建版本库

- 小乌龟工具 ( TortoiseGit ) 使用
- 增加协作者



The screenshot shows the GitHub repository 'InfoTest' settings page. The 'Settings' tab is selected in the top navigation bar (1). In the left sidebar, the 'Collaborators' tab is selected (2). The 'Collaborators' section is displayed, showing that the repository currently has no collaborators. A search bar is present with the text 'liweiweiLW' entered (3). The 'Add collaborator' button is highlighted with a red arrow (4). The search results show a user named 'liweiweiLWW 李玮玮'.

# 创建版本库

- 2. Git 操作

## Git 常用命令

- 设置用户名

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

- 设置用户邮箱

```
git config --global user.email "**@xxx.com"
```

- 查看设置

```
git config --list
```

# 创建版本库

- Git 本地操作

- 1) 初始化一个新的 Git 仓库

- 创建一个文件夹（或使用已有的文件夹）

- `mkdir demo`

- 进入刚创建的文件夹内初始化 Git，创建 Git 仓库

- `cd demo`

- `git init`



# 目 录

- 版本控制系统介绍
- Git 与 GitHub 简介
- Git账户注册与管理
- 工作区、暂存区、远程区
- 创建版本库
- 修改管理

# 创建版本库

- Git修改操作

1) 在本地工作区修改或新增文件

2) 空白处右键→Git Bash Here

3) 添加新增加/修改的所有内容

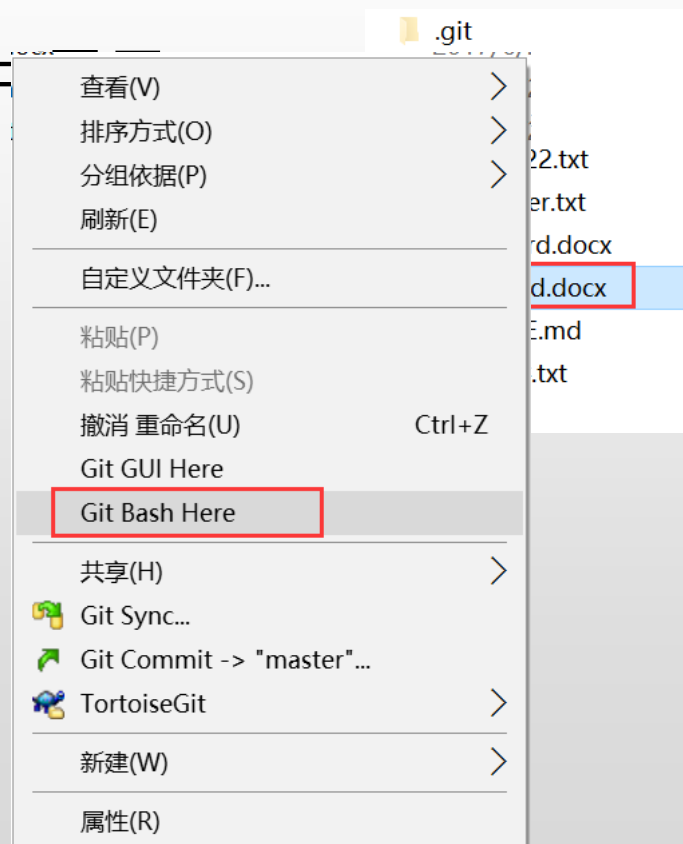
`git add *`

4) 提交到本地仓库

`git commit -m "新增文件"`

5) 提交到远程仓库

`git push`







# Thanks