



软件工程-2018

# Git的简单教程

助教：杨林，韩浩

本ppt参考廖雪峰的git教程  
<https://www.liaoxuefeng.com/wiki/0013739516305929606dd1836124857c67b8067c8c017b000>

# 目录

CONTENTS

- ▷ 第一部分 『Git简介与安装』
- ▷ 第二部分 『Git创建版本库』
- ▷ 第三部分 『Git提交修改』
- ▷ 第四部分 『Git版本查看与跳转』
- ▷ 第五部分 『Git远程仓库』
- ▷ 第六部分 『Git分支建立、推送、合并』

# 1

## Git简介与安装

Git是目前世界上最先进的**分布式版本控制系统**（没有之一）

### 版本控制



备份文件



版本	文件名	用户	说明	日期
1	readme	aa	删除了a行	10.01
2	readme	bb	增加了b行	10.02
3	readme	aa	修改了a行	10.04
4	readme	cc	修改了c行	10.11

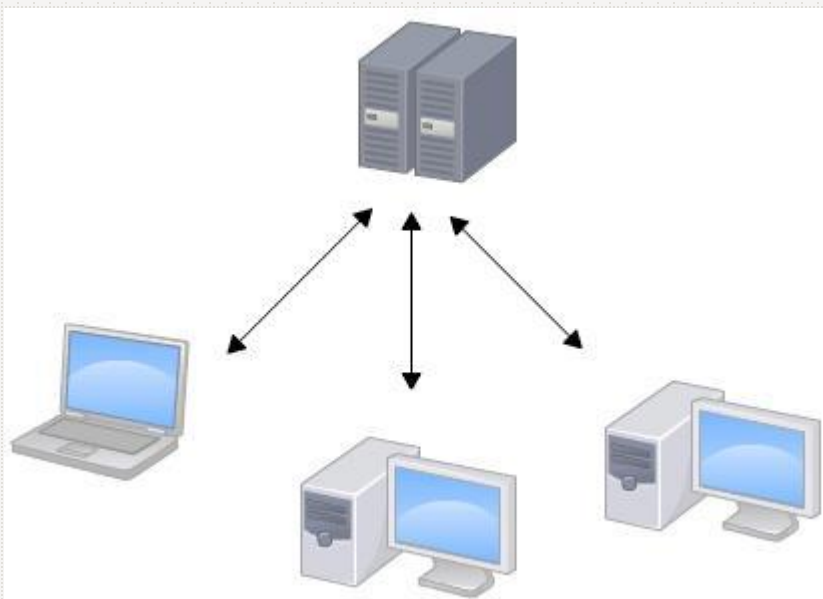
版本控制

# 1

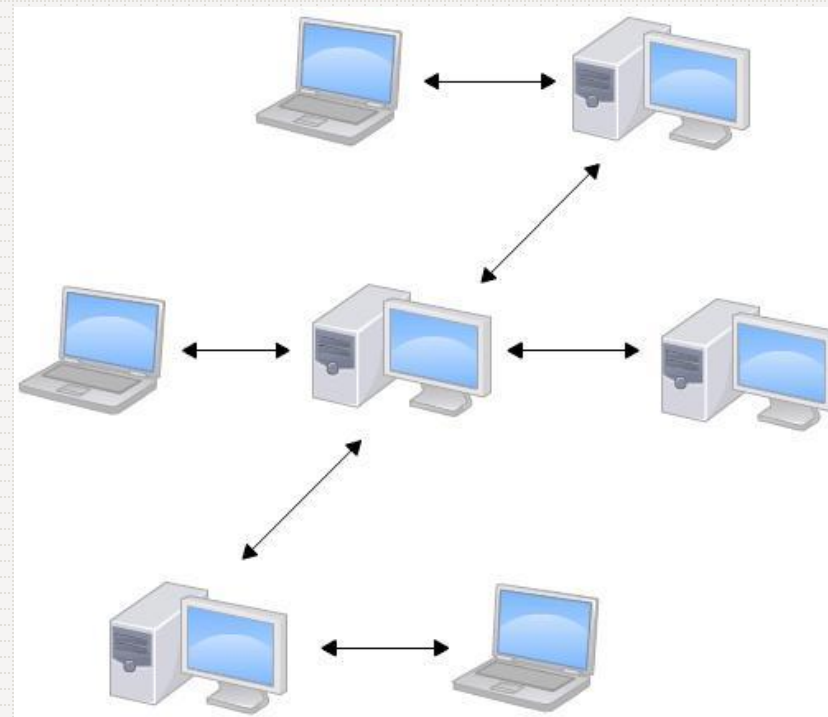
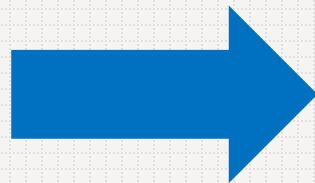
## Git简介与安装

Git是目前世界上最先进的**分布式版本控制系统**（没有之一）

**分布式**



集中式版本控制系统



分布式版本控制系统

# 1

## Git简介与安装

- Windows 和 Mac OS
  - <https://git-scm.com/downloads>



- LINUX
  - Ubuntu : apt-get install git
  - Centos : yum install git
- 默认安装即可，安装后要填入个人信息设置，推荐使用命令行进行操作
  - `git config --global user.name "Your Name"`
  - `git config --global user.email "email@example.com"`

# 1

## Git简介与安装

- Windows在win键菜单输入git即可找到git bash的应用

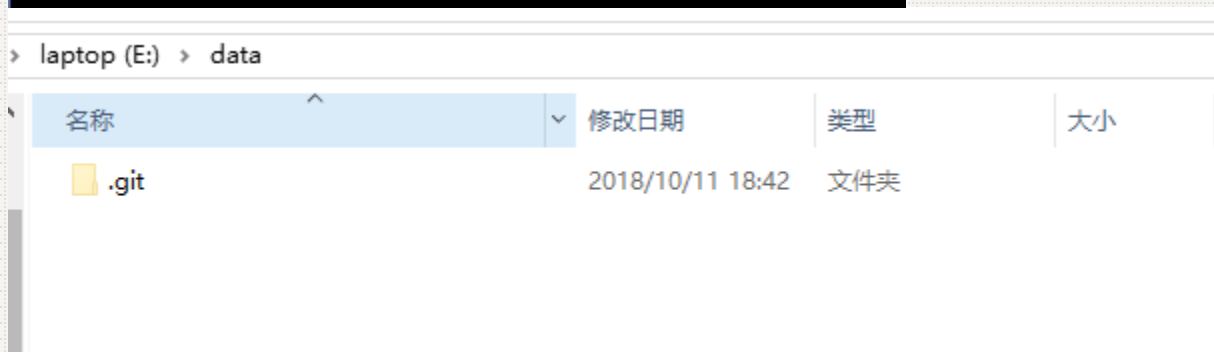


- 在文件夹空白处右键，即可选择在该文件路径下打开



- git init命令把这个目录变成Git可以管理的仓库
  - Git就把仓库建好了，而且告诉你是一个空的仓库（empty Git repository）
  - 目录下多了一个.git的隐藏目录，这个目录是Git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。

```
yanglin@DESKTOP-S57QABK MINGW64 /e/data
$ git init
Initialized empty Git repository in E:/data/.git/
```



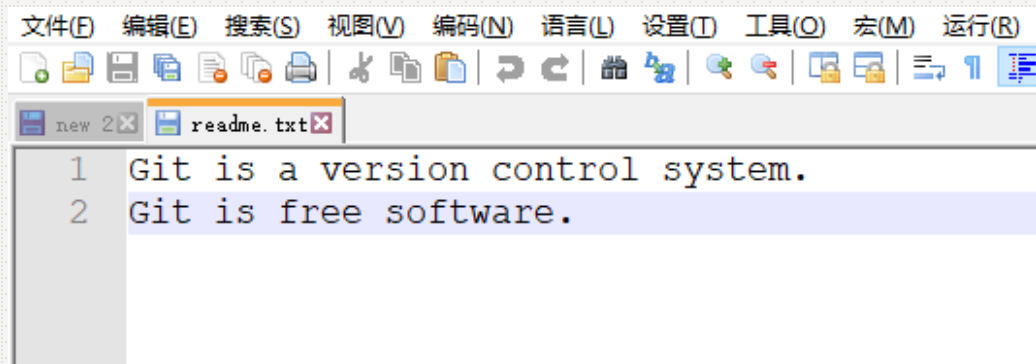
- 所有的版本控制系统，其实只能**跟踪文本文件的改动**，比如TXT文件，网页html，所有的程序代码等等，而图片、视频这些二进制文件，虽然也能由版本控制系统管理，但**没法跟踪文件的变化**，Microsoft的Word格式是二进制格式，因此，版本控制系统是没法跟踪Word文件的改动的。
- **建议使用标准的UTF-8编码**，所有语言使用同一种编码，既没有冲突，又被所有平台所支持。
- Windows同学最好不要使用自带的记事本编辑任何文本文件，在每个文件开头添加了0xefbbbf（十六进制）的字符，**建议你下载Notepad++代替记事本**，记得把Notepad++的默认编码设置为UTF-8 without BOM即可



- 用命令git add告诉Git，把文件添加到仓库
- 用命令git commit告诉Git，把文件提交到仓库
  - git commit命令执行成功后会告诉你，1 file changed：1个文件被改动（我们新添加的readme.txt文件）；2 insertions：插入了两行内容（readme.txt有两行内容）。

```
yanglin@DESKTOP-S57QABK MINGW64 /e/data (master)
$ git add readme.txt

yanglin@DESKTOP-S57QABK MINGW64 /e/data (master)
$ git commit -m "add readme.txt"
[master (root-commit) 479533f] add readme.txt
1 file changed, 2 insertions(+)
create mode 100644 readme.txt
```

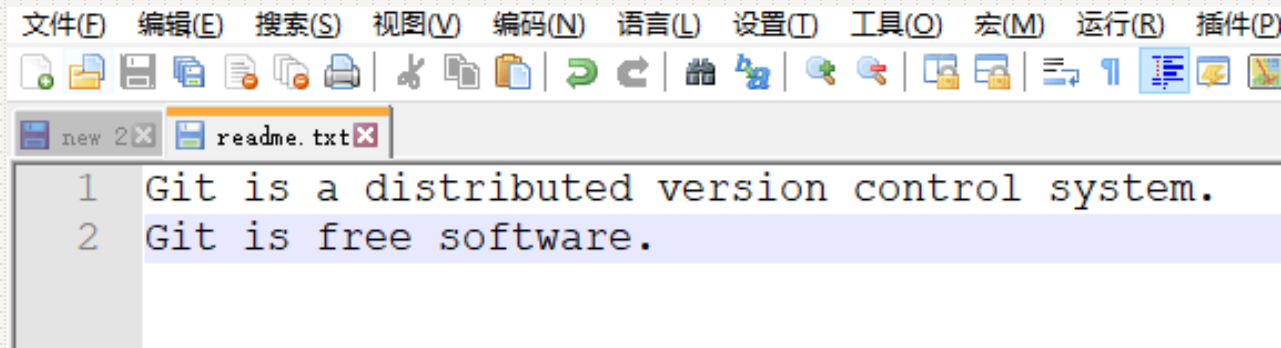


- 使用命令git add <file>，注意，可反复多次使用，添加多个文件
- 使用命令git commit -m <message>，完成

### 3

## Git提交修改

- 已经成功地添加并提交了一个readme.txt文件，现在，是时候继续工作了，于是，我们继续修改readme.txt文件，改成如下内容



- 使用命令git status查看仓库当前状态

```
yanglin@DESKTOP-S57QABK MINGW64 /e/data (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   readme.txt

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

# 3

## Git提交修改

- 使用命令git diff 查看修改文件详情

```
yanglin@DESKTOP-S57QABK MINGW64 /e/data (master)
$ git diff readme.txt
diff --git a/readme.txt b/readme.txt
index d8036c1..013b5bc 100644
--- a/readme.txt
+++ b/readme.txt
@@ -1,2 +1,2 @@
-Git is a version control system.
+Git is a distributed version control system.
 Git is free software.
 \ No newline at end of file
```

- 之后使用git add与git commit 即可提交修改，之后在用git status查看 可以发现没有要修改的地方

# 4

## Git版本查看与跳转

直接回跳上一个版本

```
git reset --hard HEAD^
```

跳转制定版本 输入版本号前几位即可

```
git reset -hard bf6a
```

```
yanglin@DESKTOP-S57QABK MINGW64 /e/data (master)
$ git log
commit 998e86f73f6502acb60c589de08d8b1b574d68a2 (HEAD -> master)
Author: yanglin <yanglincs17@gmail.com>
Date: Thu Oct 11 19:25:21 2018 +0800

    GPL

commit bf6a69724cc434cf3c5df947ff83d45639e130a2
Author: yanglin <yanglincs17@gmail.com>
Date: Thu Oct 11 19:22:35 2018 +0800

    add distributed

commit 479533f6c51a921a0189c33855529af106abb402
Author: yanglin <yanglincs17@gmail.com>
Date: Thu Oct 11 18:57:57 2018 +0800

    add readme.txt
```

# 4

## Git版本查看与跳转

版本1

Git is a version control system.

Git is free software.

版本2

Git is a distributed version control system.

Git is free software.

版本3

Git is a distributed version control system.

Git is free software distributed under the GPL.

```
yanglin@DESKTOP-S57QABK MINGW64 /e/data (master)
$ git log
commit 998e86f73f6502acb60c589de08d8b1b574d68a2 (HEAD -> master)
Author: yanglin <yanglincs17@gmail.com>
Date: Thu Oct 11 19:25:21 2018 +0800

    GPL

commit bf6a69724cc434cf3c5df947ff83d45639e130a2
Author: yanglin <yanglincs17@gmail.com>
Date: Thu Oct 11 19:22:35 2018 +0800

    add distributed

commit 479533f6c51a921a0189c33855529af106abb402
Author: yanglin <yanglincs17@gmail.com>
Date: Thu Oct 11 18:57:57 2018 +0800

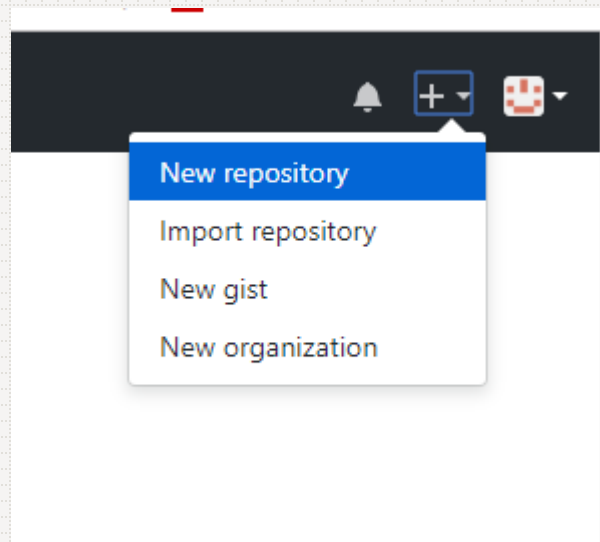
    add readme.txt
```

使用**git log**即可查看到最近几次的提交信息

# 5

## Git远程仓库

作为中转的仓库一般24小时开机在线，  
可以选择在GitHub上建立新仓库（公开）



在一台24小时开机的主机上建立私有仓库（阿里云、腾讯云、校内的主机，有服务器的可以自己尝试搭建）

## 5

## Git远程仓库

生成密钥，生成一个公钥，一个私钥，公钥可以交给别人，私钥留给自己

```
yanglin@DESKTOP-S57QABK MINGW64 ~  
$ ssh-keygen  
Generating public/private rsa key pair.  
Enter file in which to save the key (/c/Users/yanglin/.ssh/id_rsa): |
```

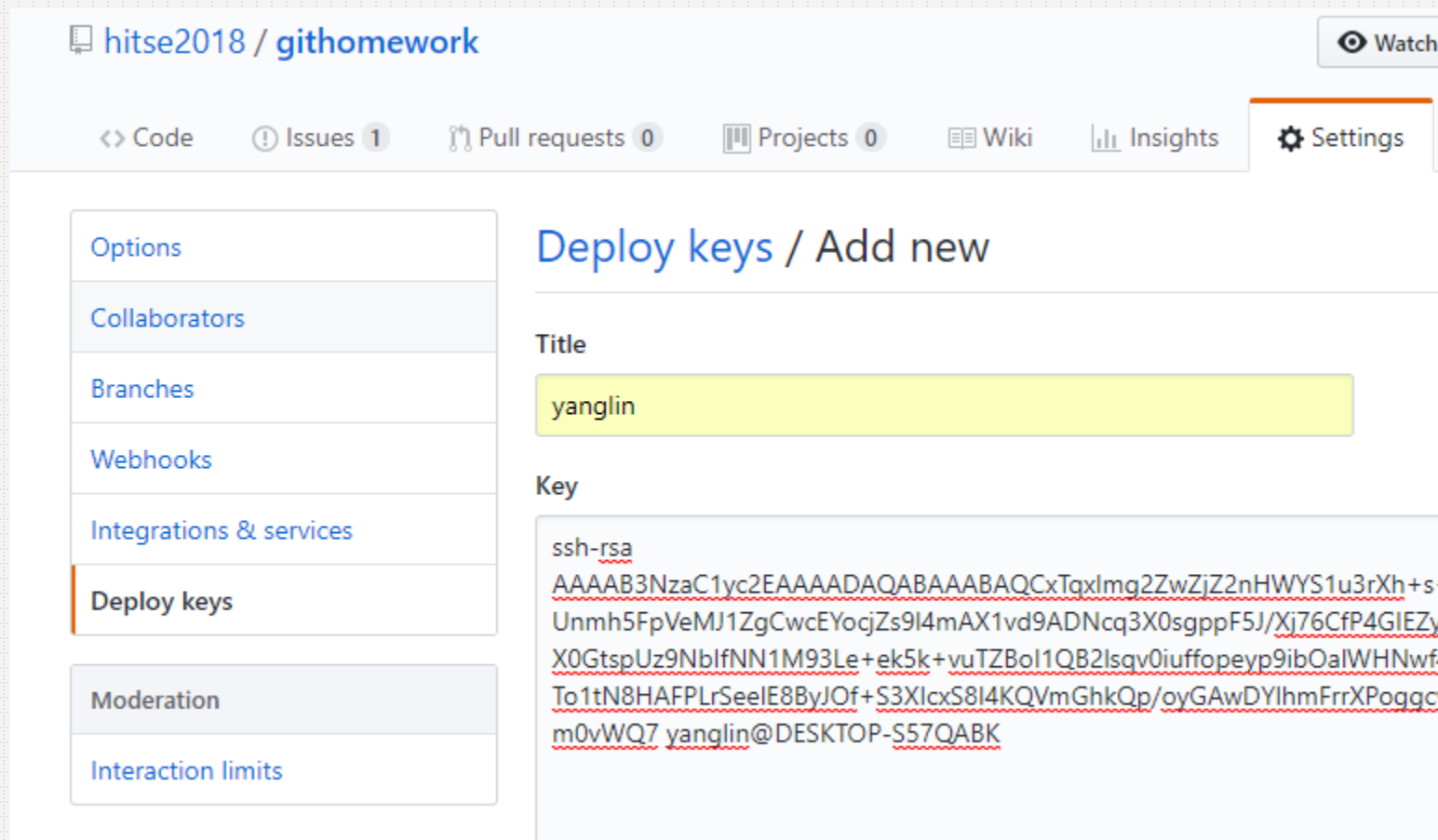
```
id_rsa.pub  
1 ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQxCxTqxlmg2ZwZjZ2nHWYS1u!
```

## 5

## Git远程仓库

进行密钥布置，即可在本地推送到远程仓库

可以添加多个人的公钥，这样多人可以推送到该仓库，也可以不添加公钥，让对方发起提交申请再自己同意合并



The screenshot shows the GitHub repository settings for 'hitse2018 / githomework'. The 'Settings' tab is selected, and the 'Deploy keys / Add new' section is active. The 'Title' field contains 'yanglin'. The 'Key' field contains an SSH key: 'ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQBAQCxTqxImg2ZwZjZ2nHWYS1u3rXh+s+Unmh5FpVeMJ1ZgCwcEYocjZs9I4mAX1vd9ADNcq3X0sgppF5J/Xj76CfP4GIEZyX0GtspUz9NblfNN1M93Le+ek5k+vuTZBoI1QB2lsqv0iuffopeyp9ibOaIWHNwfrTo1tN8HAFPLrSeelE8ByJOf+S3XlCxS8I4KQVmGhkQp/oyGAwDYIhmFrrXPoggc m0vWQ7 yanglin@DESKTOP-S57QABK'. The left sidebar shows navigation options: Options, Collaborators, Branches, Webhooks, Integrations & services, Deploy keys (selected), Moderation, and Interaction limits.



# 5

## Git远程仓库

设置远程仓库，推送、拷贝

本地设置好远程仓库地址

```
git remote add origin git@github.com:hitse2018/githomework.git
```

推送主分支到远程仓库

```
git push origin master
```

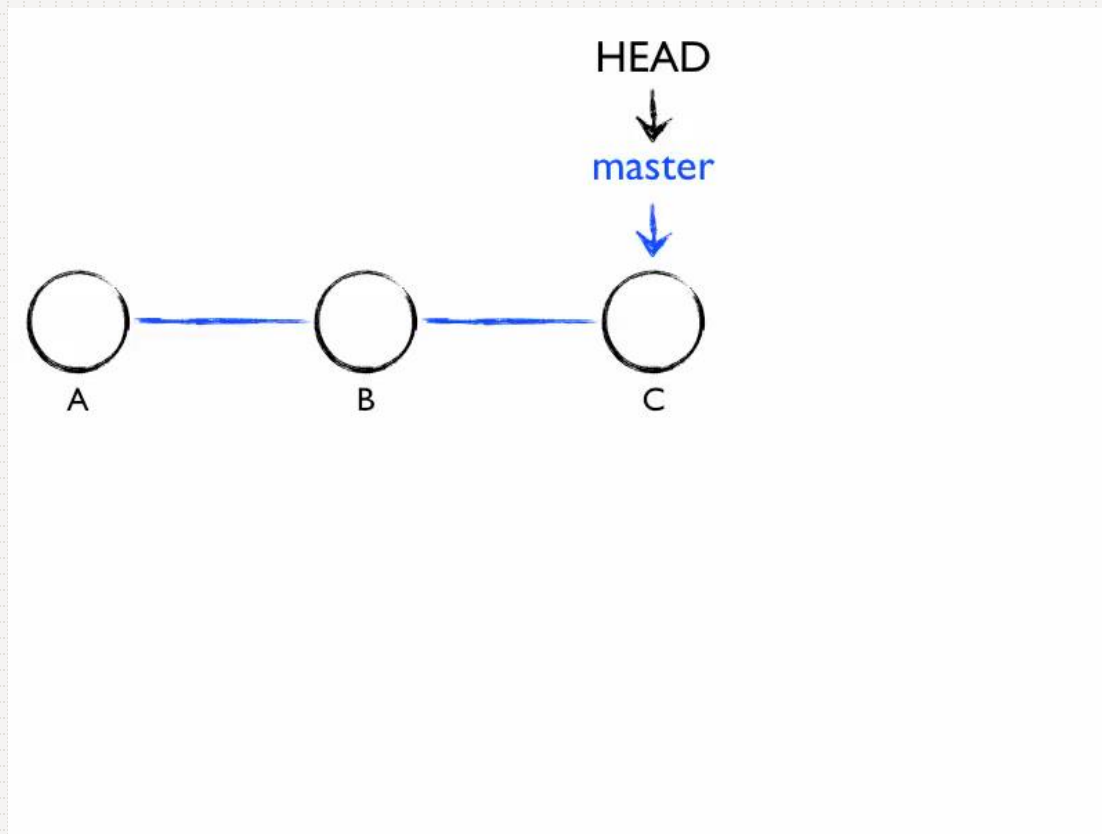
克隆别人的github仓库时无需验证

```
git clone git@github.com:hitse2018/githomework.git
```

## 6

## Git分支建立、推送、合并

在master分支上可新建多个其他分支，每个人在自己的分支上工作，之后在合并至master上



## 6

## Git分支建立、推送、合并

两人合作代码

	Lihua	Hanmeimei
本地	<code>git clone git@github.com:...</code>	<code>git clone git@github.com:....</code>
本地	<code>Git checkout -b lihua</code>	<code>Git checkout -b hanmeimei</code>
本地	<code>Git commit -m "add lihua"</code>	<code>Git commit -m "add hanmeimei"</code>
推送至远程	<code>Git push origin lihua</code>	<code>Git push origin hanmeimei</code>
远程合并	远程的master分支上就可以 <code>git merge lihua</code> , <code>git merge hanmeimei</code>	

### 冲突与解决

为尽量避免冲突，多人合作最好别多人对同一文件操作，前期约定好每个人写的模块与调用接口，仅合并对方的文件

避免多人操作一个文件可有效减少冲突产生

有冲突产生多沟通交流

谨慎使用git push -f origin xxxx

目前，码农持枪杀人的动机仍然是个谜。有人猜测道：“同事不写注释，不遵循驼峰命名，括号换行，最主要还天天git push -f等因素”激怒了这名行凶者。

看到上面的原因，感觉太假了，“天天git push -f”，开什么玩笑，第一天这么干就被打断腿了好不好？

<https://github.com/hitse2018/githomework>

本次作业以小组为单位

### 1.建立git仓库，权限管理

组长新建一个git仓库，自己命名，在readme文件中说明队伍名称及组成人员，并将git仓库推送到github上，并将各个组员的公钥导入到该仓库允许组员进行操作

### 2.分支管理

各组各个成员均将仓库克隆到本地，各自新建一个自己名字（英文）的分支，并在自己的分支下新建一个学号-名字.txt的文件，并将自己的分支推送到组长的github上

### 3.合并分支生成新版本

组长将各个分支合并到主分支上

完成后将小组的仓库地址及小组名称发送到[hitse2018@163.com](mailto:hitse2018@163.com)邮箱，待截止日期后24小时可删除该仓库

本ppt仅作为简单操作与简介，想进一步学习了解git的知识的同学可以查看廖雪峰的git教程

(<https://www.liaoxuefeng.com/wiki/0013739516305929606dd18361248578c67b8067c8c017b000>)及网上的相关资料