# Git

## Git 简介

### 1、什么是Git?

什么是Git? 比如一个项目，两个人同时参与开发，那么就把这个项目放在一个****公共的地方****，需要的时候都可以去获取，有什么改动，都可以进行提交。  
  
为了做到这一点，就需要一个版本控制系统，Git就是这样一个免费、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。

### 2、提供Git服务的网站

国外常用的是: [http://github.com](http://github.com/)  
国内常用的是码云：[http://git.oschina.net](http://git.oschina.net/)

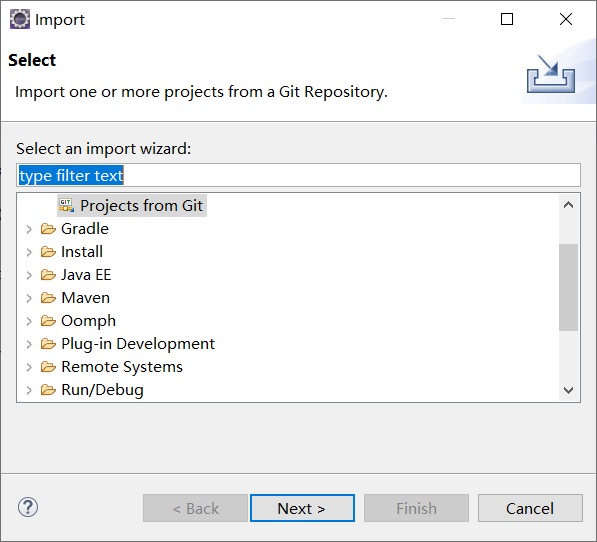
github慢，但是上面内容多，大神多，简历里也需要github.com  
码云快，但是内容少

### 3、注册

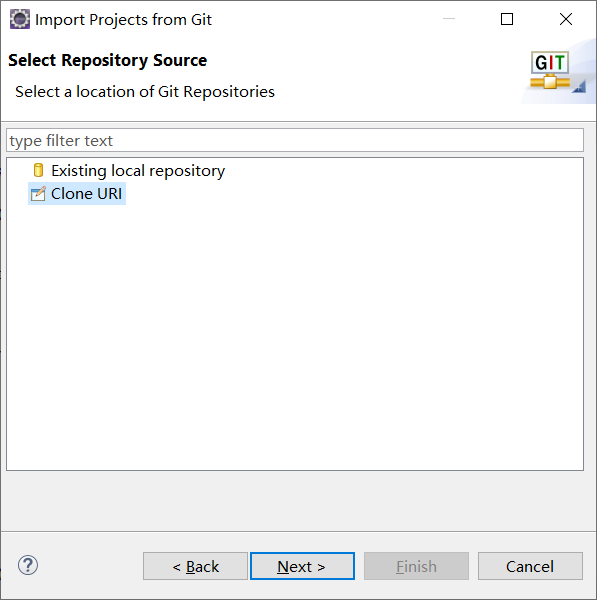
注册地址： [http://github.com](http://github.com/)

## Git pull项目

### 1、菜单-File->import->Git->Projects from Git



### 2、使用某个服务器上的git资源

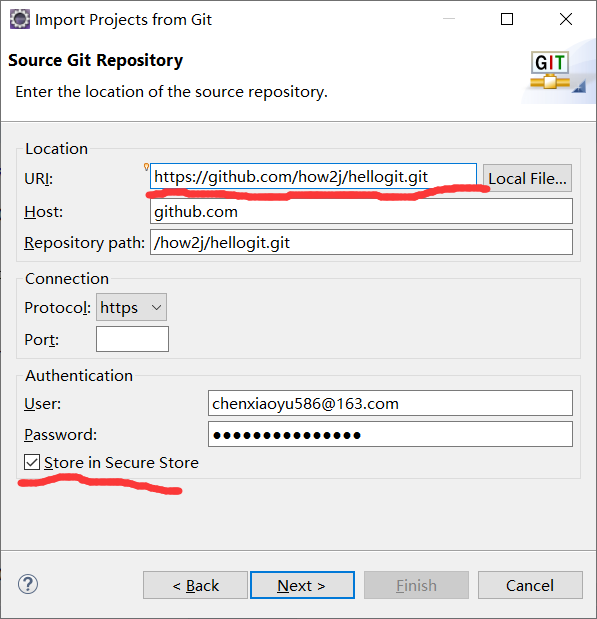


### 3、输入git参数

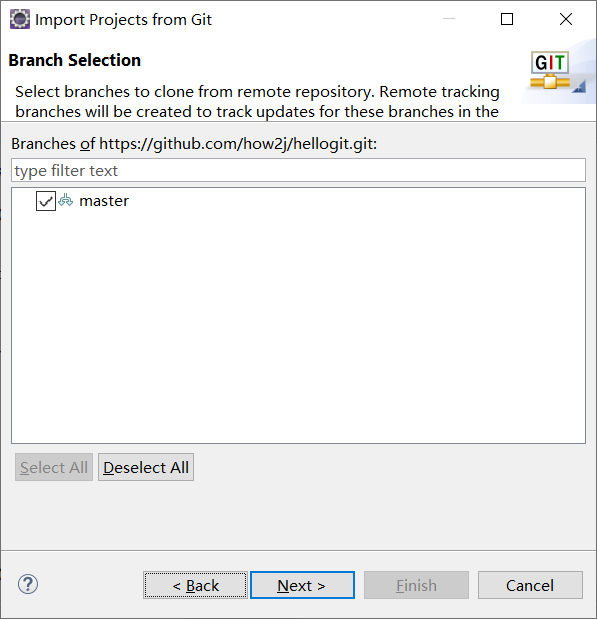
输入url之后，后面两个参数自动生成

然后输入帐号密码

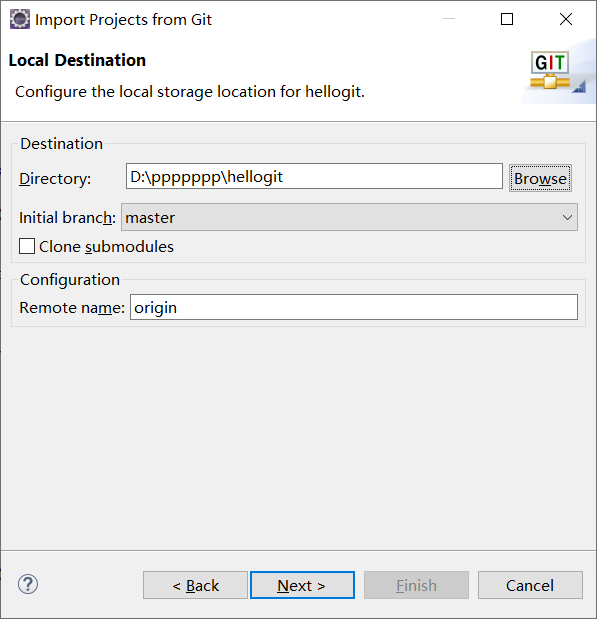
最后Store in Secure Store勾上，表示Eclipse会保持你的账号密码。



### 4、选择一个分支

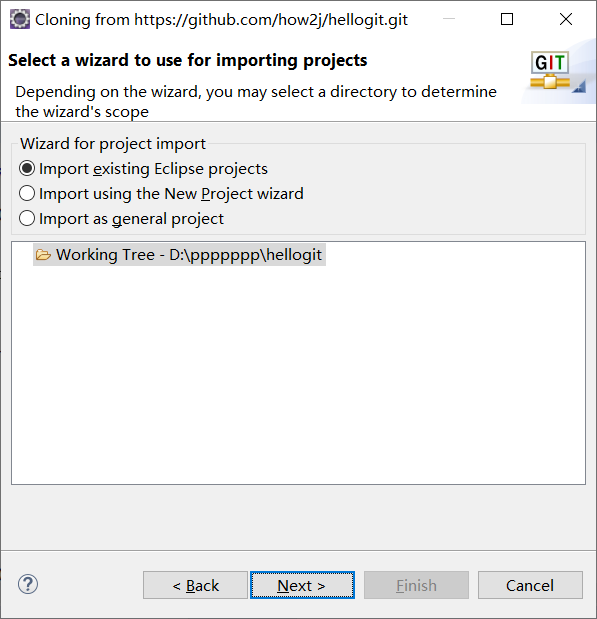


### 5、放在哪个路径

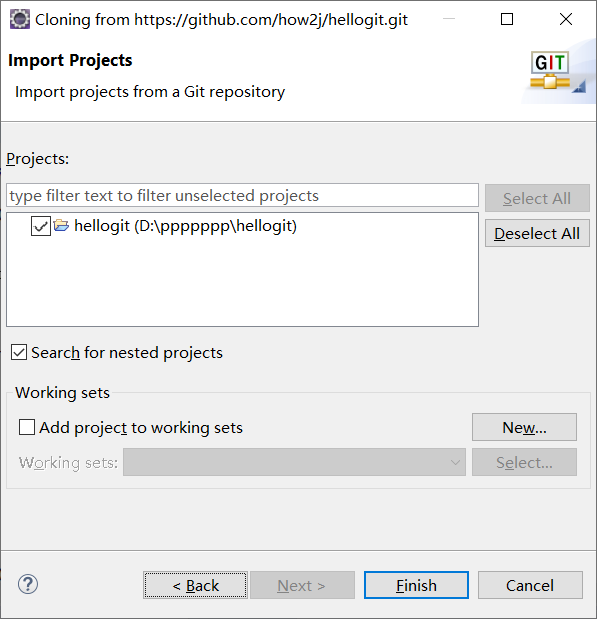


### 6、将项目导入到Eclipse中

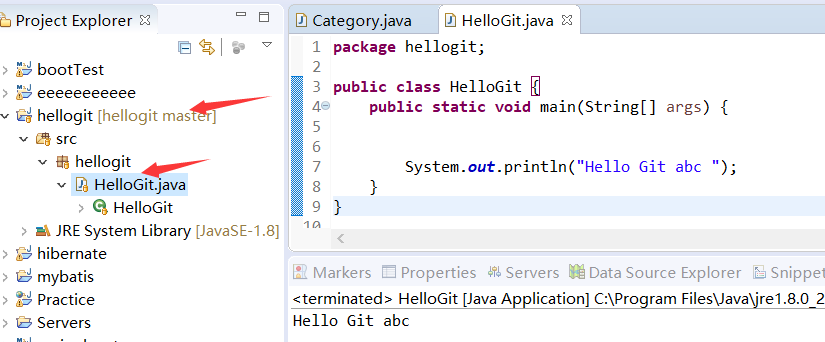
选择import existing Eclipse projects



### 7、确定位置



### 8、运行



## Git 创建项目

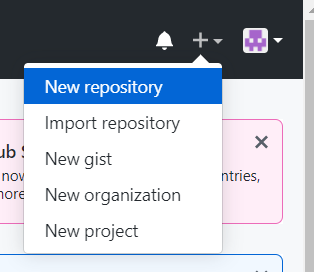
把自己的项目共享到Git上

### 在Git上新建仓库

登录

点击右上角的加号，点击New repository创建仓库

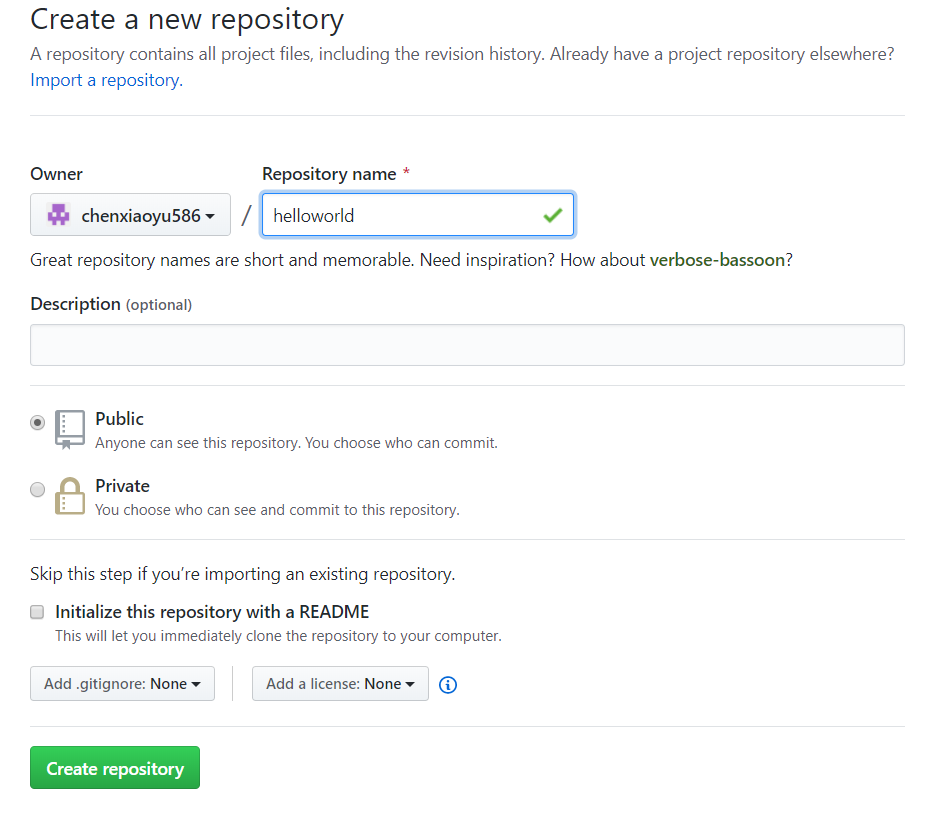
Git上的仓库就相当于项目的意思



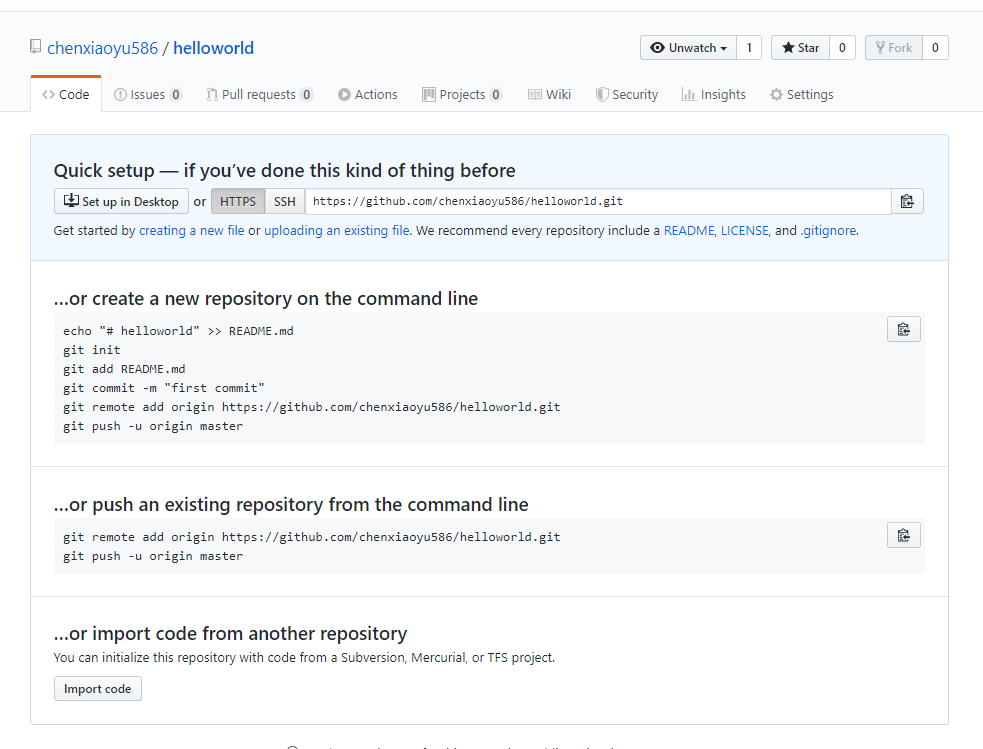
### 输入仓库名称

输入的是helloworld

****注：**** 不要选 Initialize this repository with a README, 否则会出来奇奇怪怪的问题

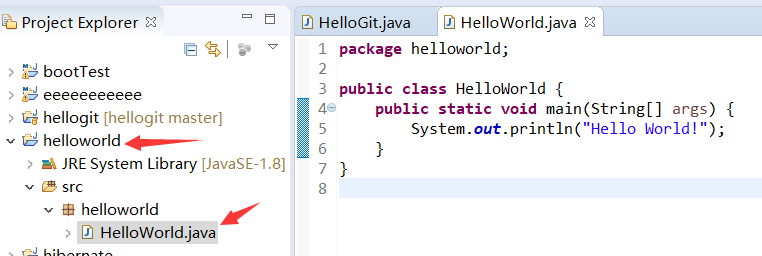


### 创建成功，得到Git地址



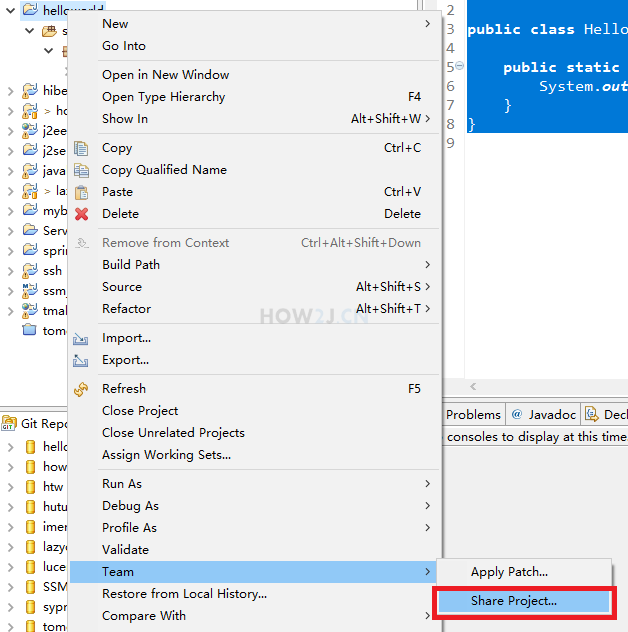
### 4、在eclipse中创建一个java项目

在Eclipse中创建一个java项目：helloword, 同时建立一个HelloWorld类



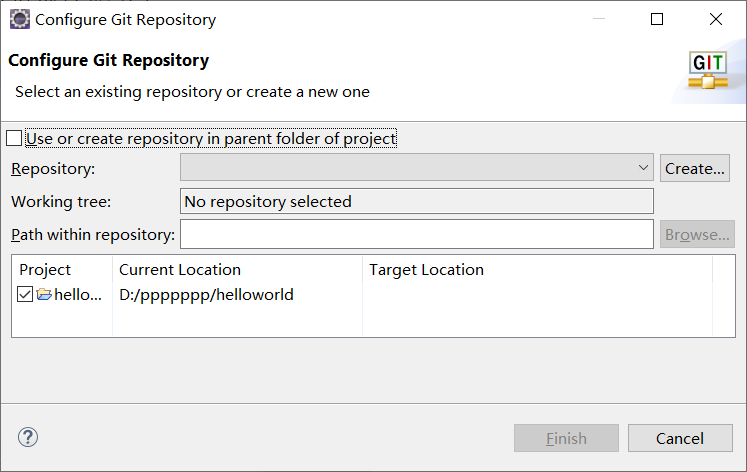
### 5、Share Project

右键项目->Team->Share Project



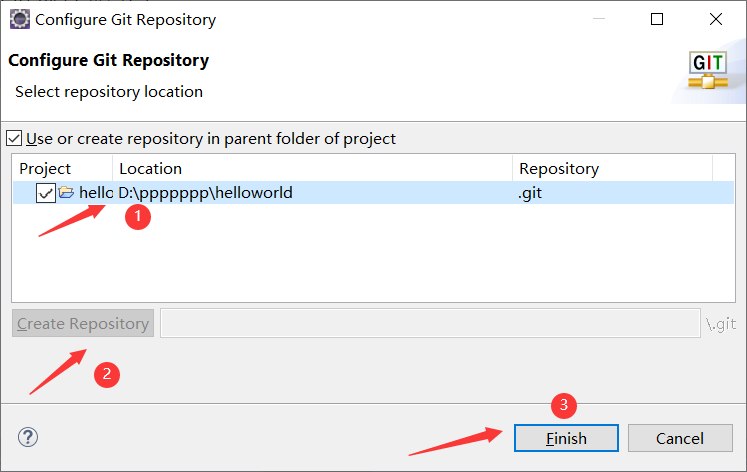
### 6、配置仓库

配置仓库，点击一下 Use or create repository in parent folder of project



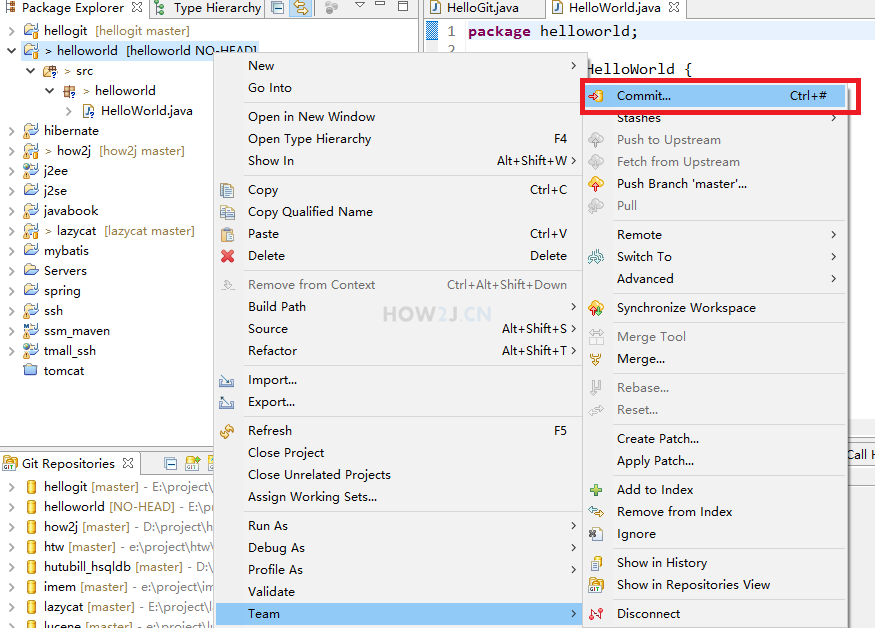
### 7、创建仓库

按顺序点击

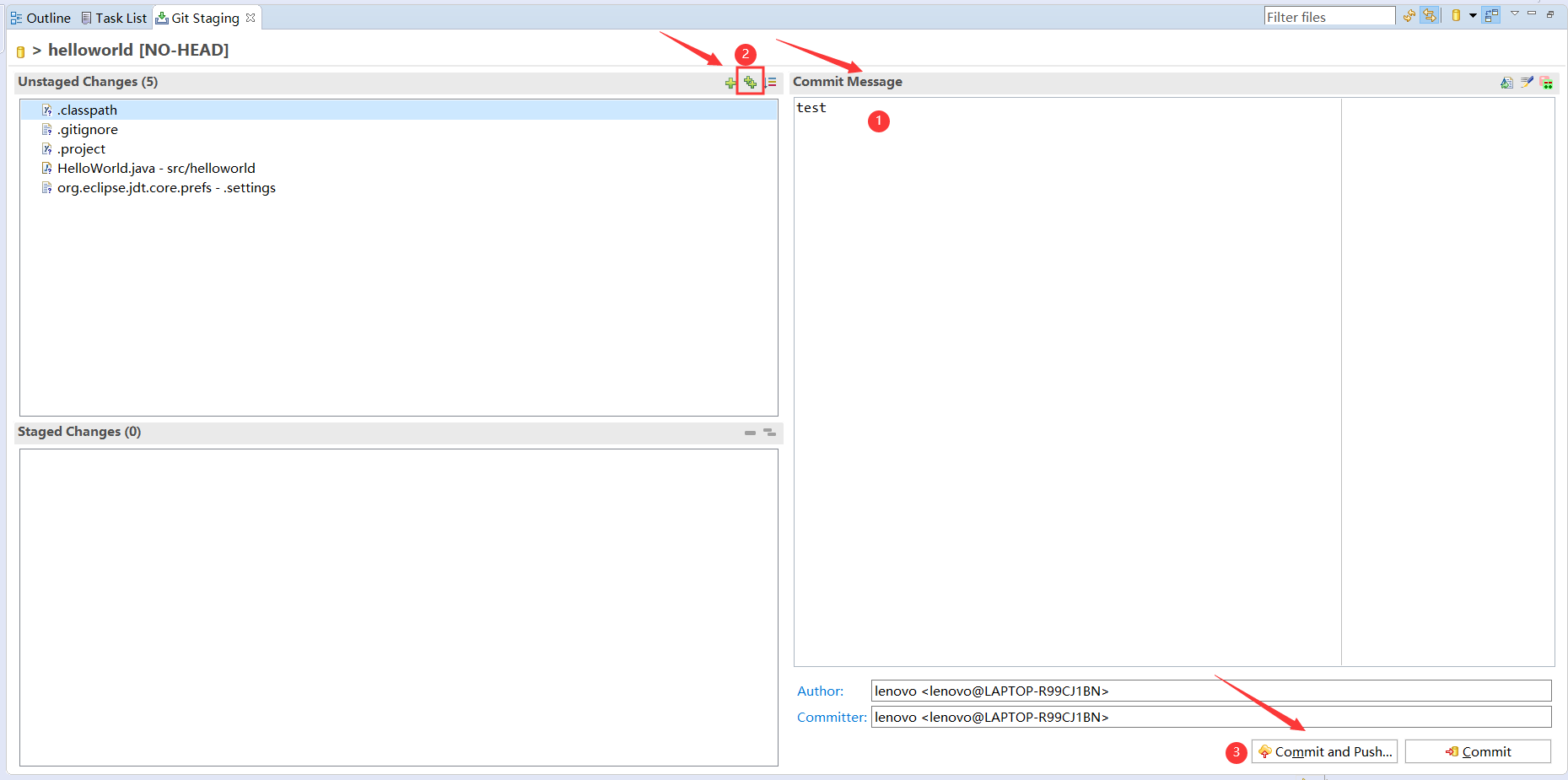


### 8、提交项目

右键项目->Team->Commit



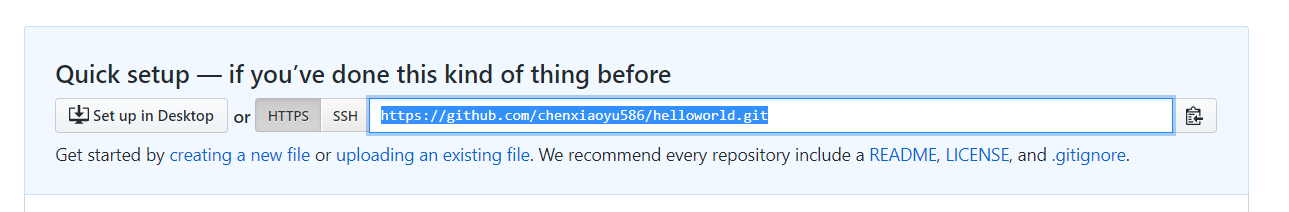
### 9、设置提交信息

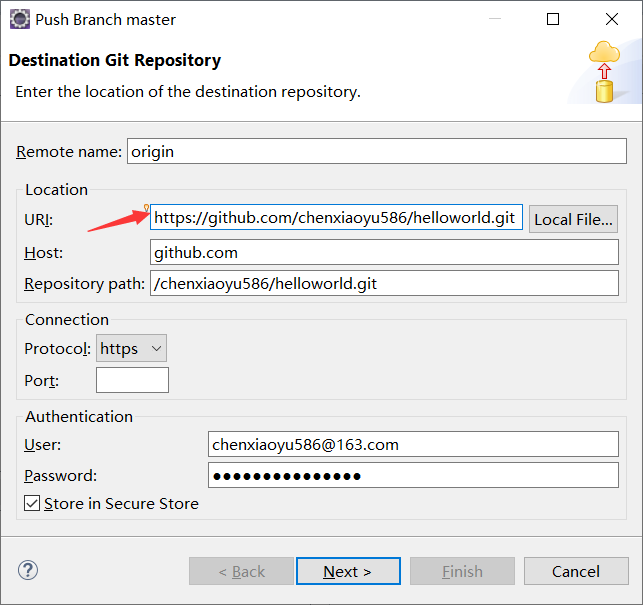


### 10、设置Git参数

复制Git地址

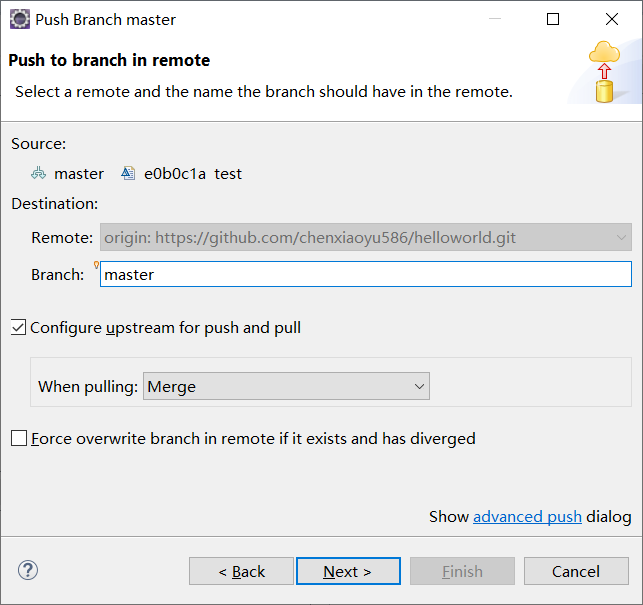
粘贴进去



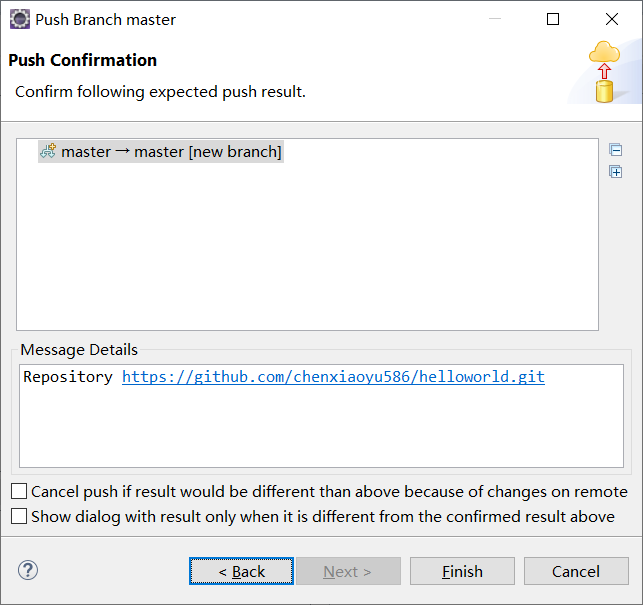


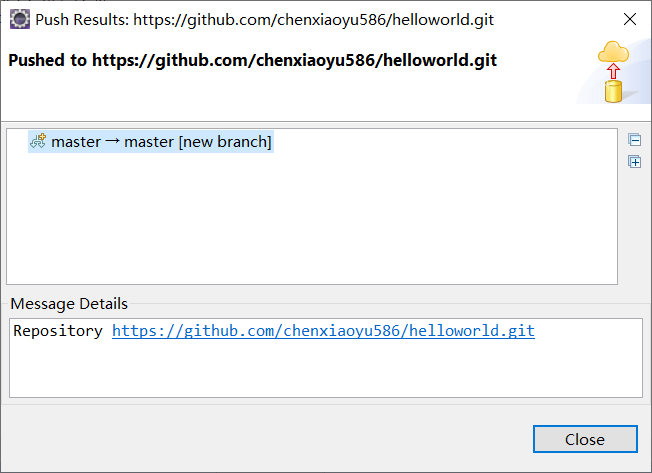
### 11、指定分支

使用默认的master分支

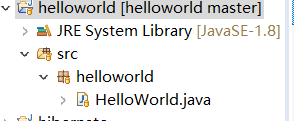


### 12、提交成功

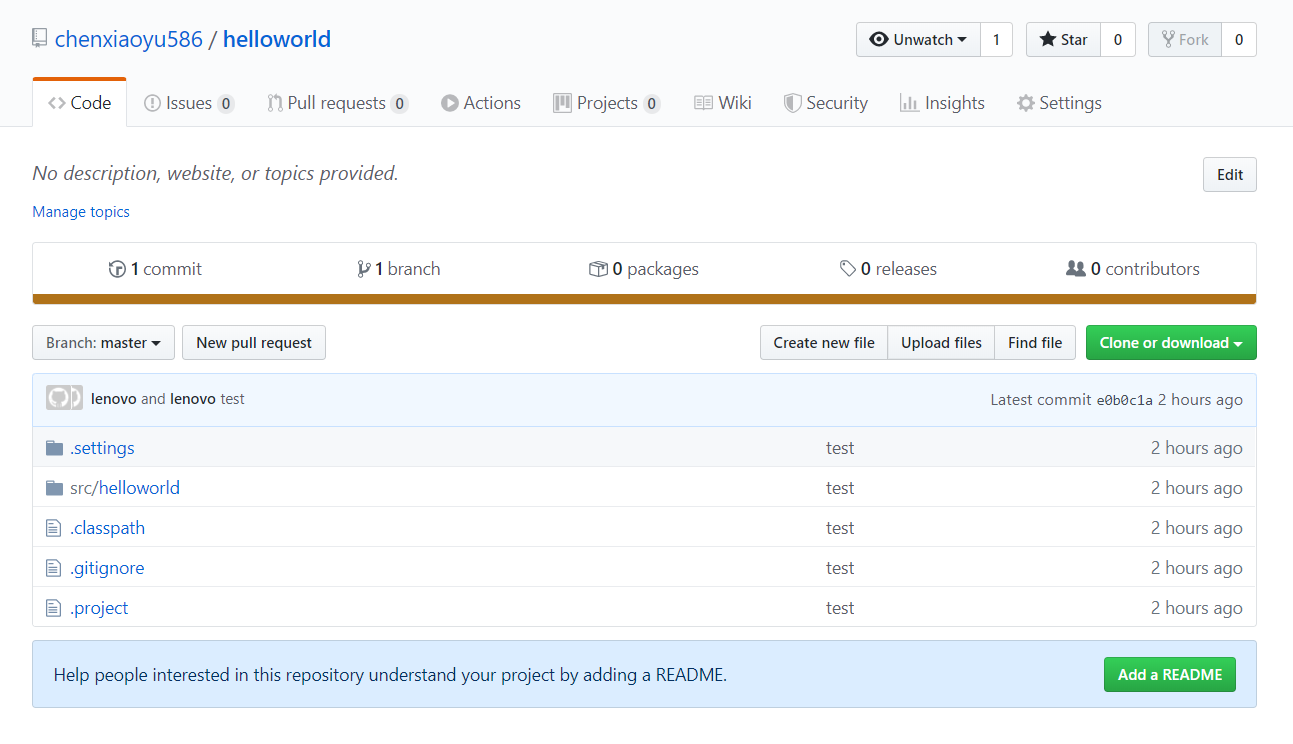




### 13、图标变化

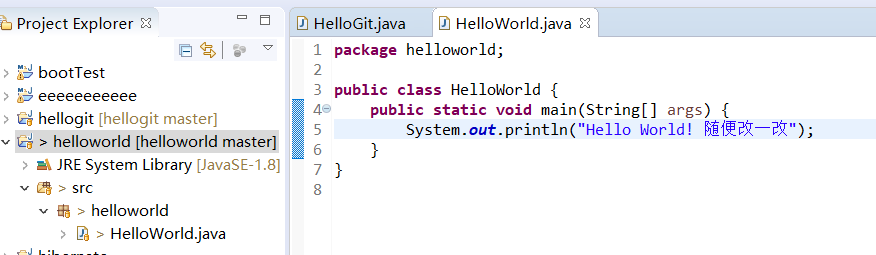


### 14、观察GitHup上的项目



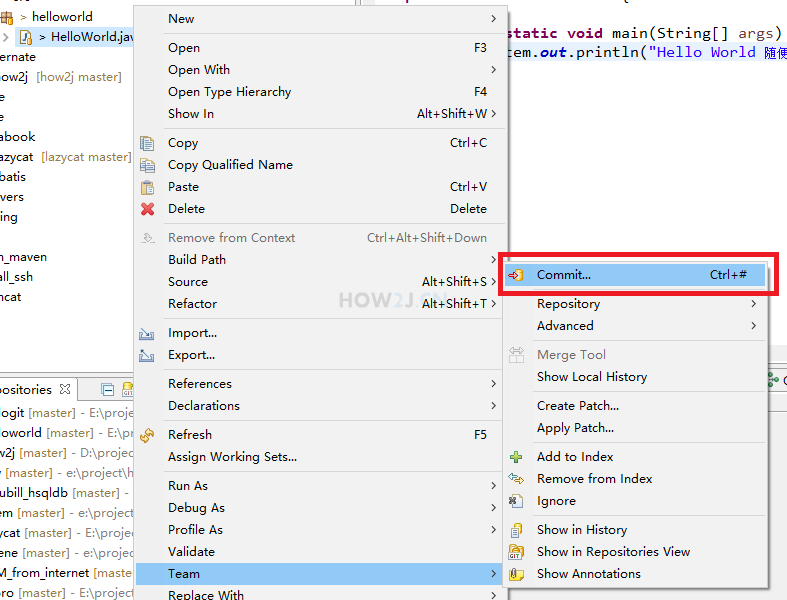
## Git修改自己的代码

### 1、对项目helloworld修改一下

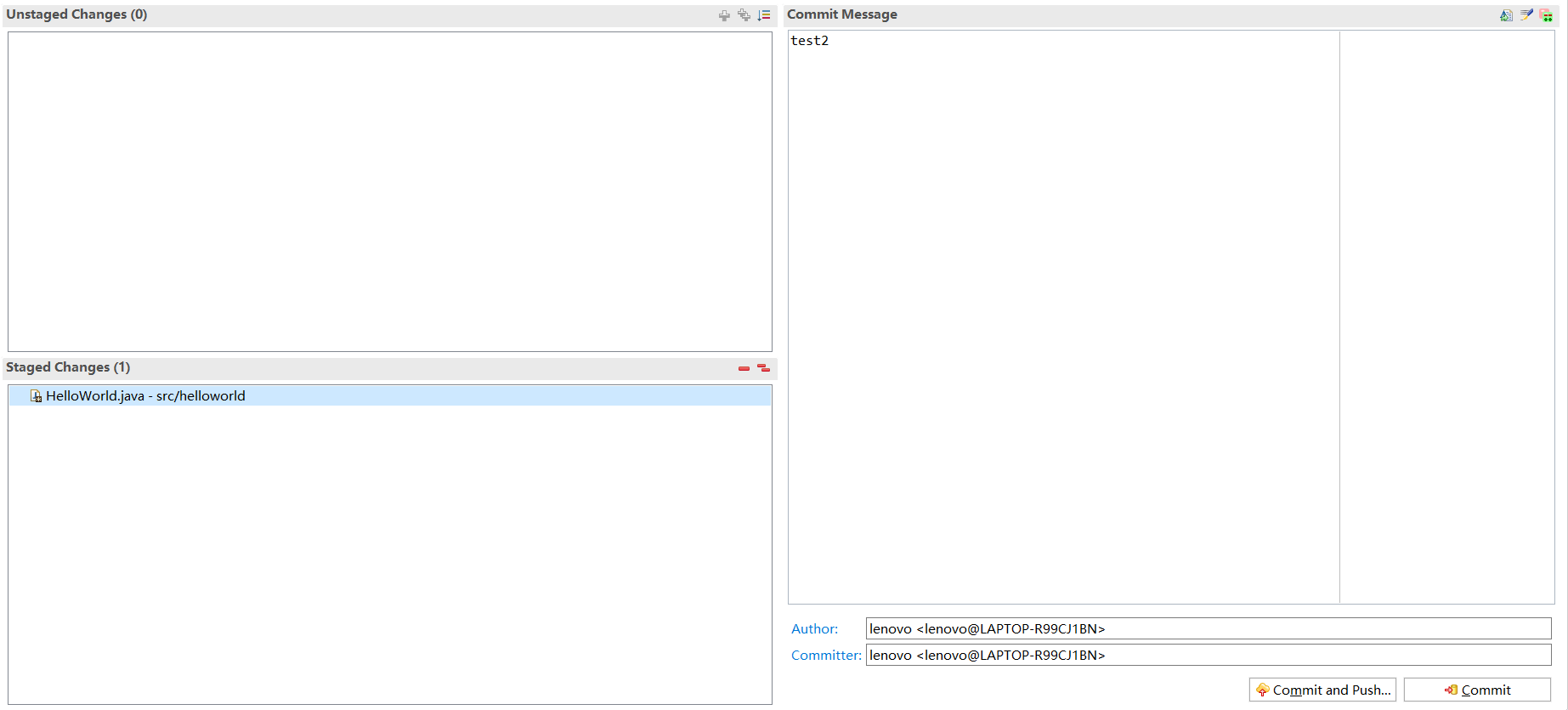


### 2、提交修改

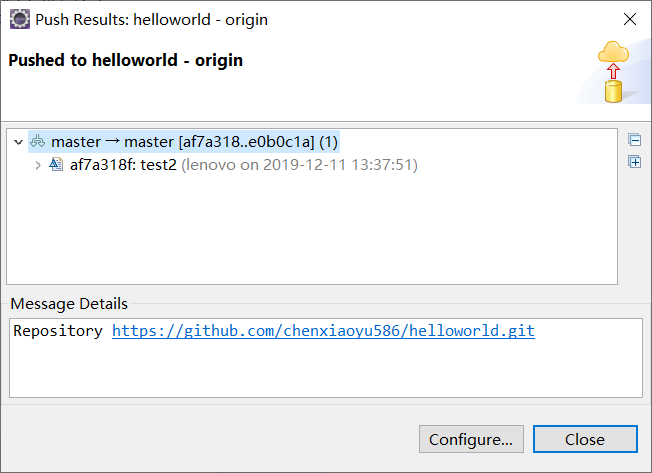
右键HelloWorld.java->Team->Commit



### 3、输入提交信息



### 提交成功



## Git修改别人的代码

如何修改别人的代码呢？ 严格得讲，是不能够直接修改别人的代码的，只能基于别人的项目，克隆一个项目出来到自己的账号上，到了自己的账号上之后，就可以修改了。  
修改了之后，再发起一次合并请求，倘若原项目作者接受了你的合并请求，那么你就为原项目做了贡献，也就修改了原项目了。

### 1、Fork

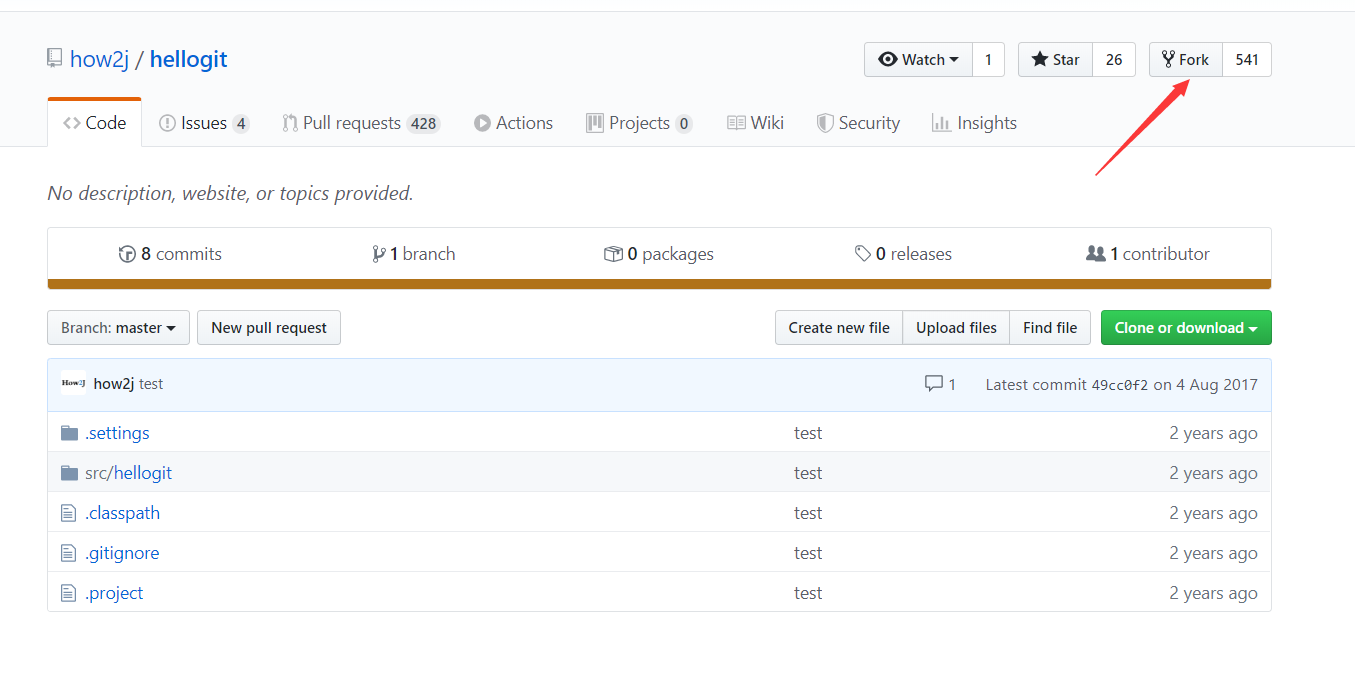
基于别人的项目，做一次克隆，克隆出一个一模一样的项目到自己的账号下，这个叫Fork

Fork英文是叉子的意思，表示在原有的项目基础上，出现了分支

### 2、在github上进行Fork操作

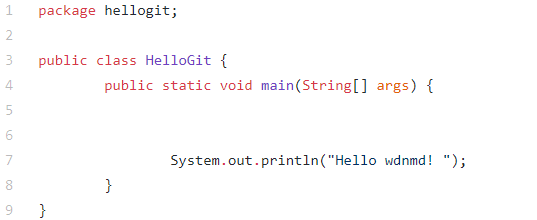
访问项目：<https://github.com/how2j/hellogit>

点击右上角Fork



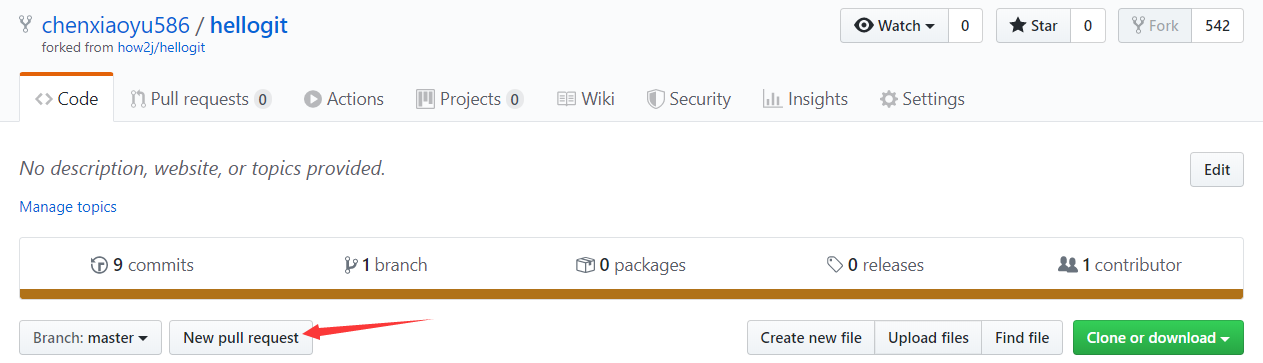
### 3、修改

Fork之后，hellogit就会被克隆一份到你的账号上，接着对HelloGit随便做点什么修改



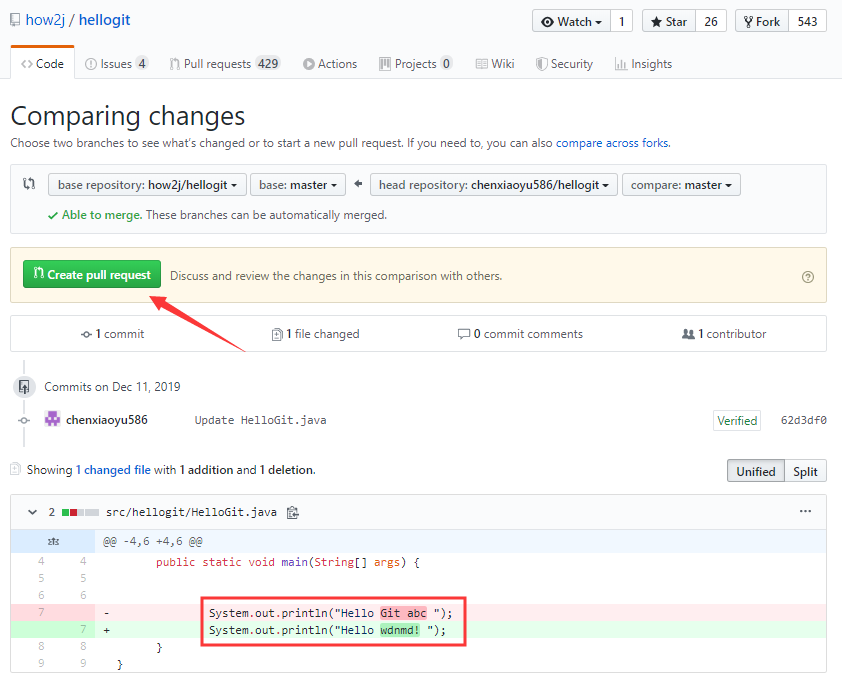
### 4、发起New Pull Request

回到github的hellogit项目下，然后点击New Pull Request，即发起一次合并请求。  
  
既然是合并请求，为什么叫Pull Request呢？ 这是站在原项目主的角度的一个动词，即发起一次请求，希望原项目主，可以把你的改动，****拉进****到他的项目里去。



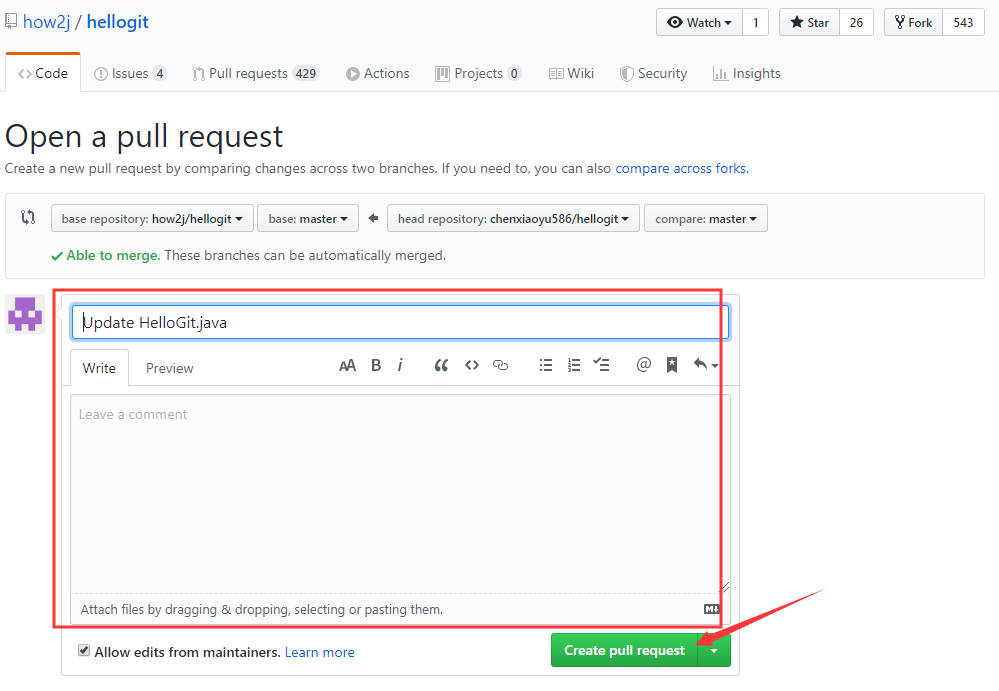
### 比较改变的地方

比较文件和原项目中的文件有什么区别，然后点击Create pull request



### 填写修改原因

填写修改文件的原因，然后点击Create pull request



### 修改提交完成

这样pull request就发出去了，然后就没你什么事了。  
剩下的就是原项目主自己判断，是否要接受你的这次pull request了。

