1. 介绍

该库内含小组分享的文章精选+bib文件+ppt（rd/\*/）、组员阅读笔记（notes/）、代码与实验报告（code/）等。

本地仓库是github仓库（远程库）的镜像，日常维护两者可以在控制台上用命令行完成。远程库的维护命令包括：

（1）git clone 把github当前版本克隆到本地；

（2）git pull 把github当前版本检出到本地；

（3）git push 把本地commits发布到github。

要对远程仓库进行编辑，首先要注册GitHub账号，收到仓库编辑的权限（即仓库作者给你发邮件，你接受邀请），才能对远程仓库进行修改

1. Git的安装方法

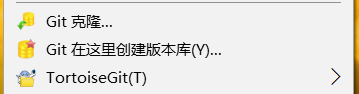
详情参见<https://www.cnblogs.com/xueweisuoyong/p/11914045.html>。

1. TortoiseGit使用方法

官网：https://tortoisegit.org/download/

1. 安装步骤

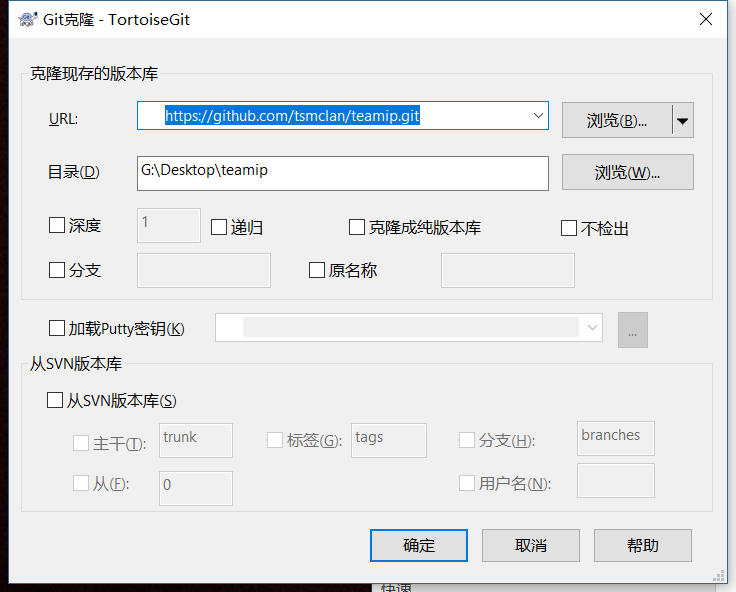
一路默认即可，安装好以后，右键，会发现菜单多了几项关于tortoisegit的选项。



到此，安装算完成了，相比tortoiseSVN，安装完tortoiseGIT不需要重启电脑。

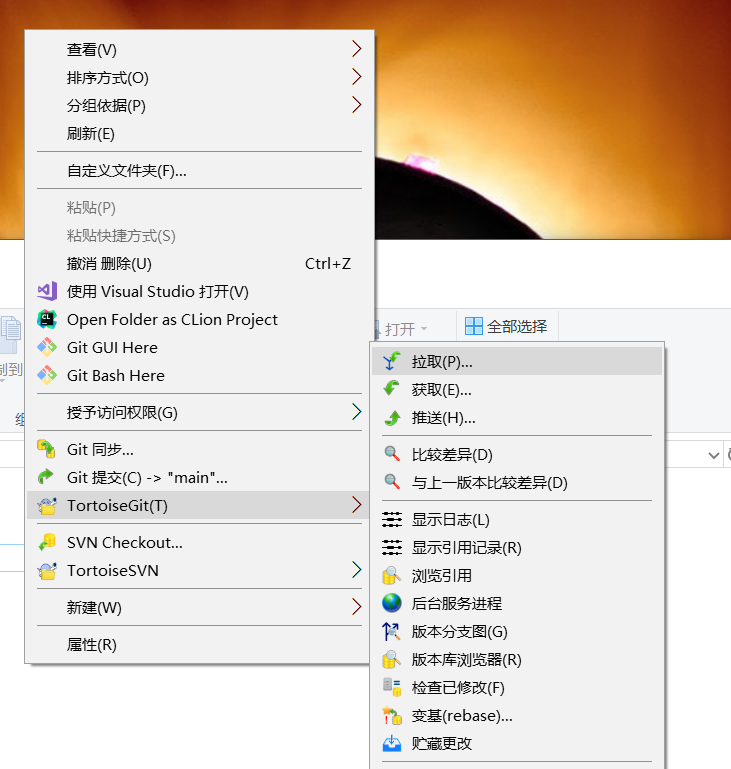
1. 下载仓库

在右键菜单中选择git clone，出现如下界面



url中填入teamip私有仓库链接<https://github.com/tsmclan/teamip.git>，目录选择自己想要存放文件的位置，最后点击确认即可下载仓库到本地。

1. 上传文件
2. 首先在仓库目录右键菜单中选择git pull（拉取），此操作作用：其他同学更改文件提交后，如果直接git push很容易造成冲突。

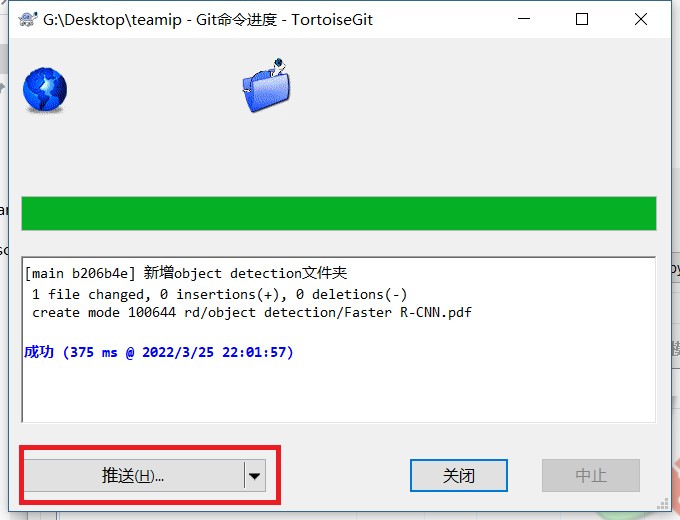


1. 右键菜单中找到git summit，在下方图片红色区域选择自己提交的文档，并在日志信息中写出更改信息，然后选择提交（summit）

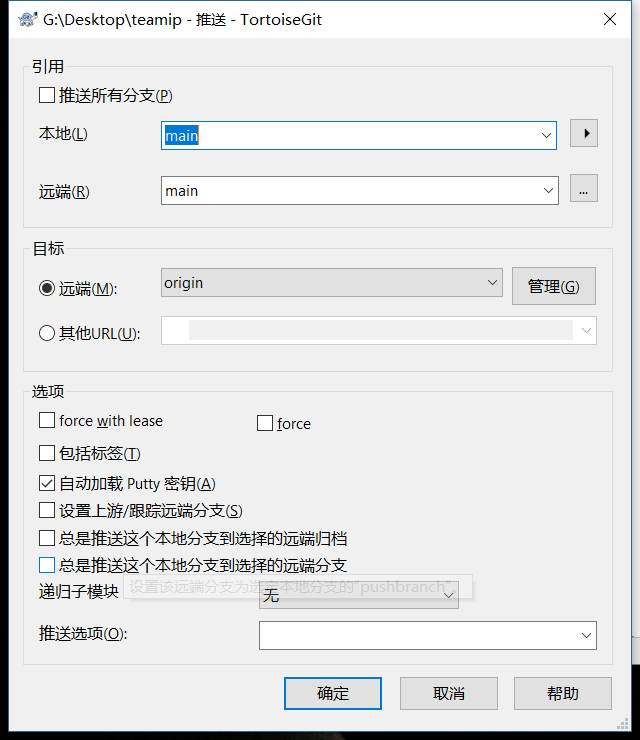


1. Push

提交（summit）完成后会弹出如下窗口，点击红色框中的推送（push）



弹出如下窗口，点击确定即可

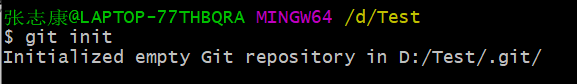
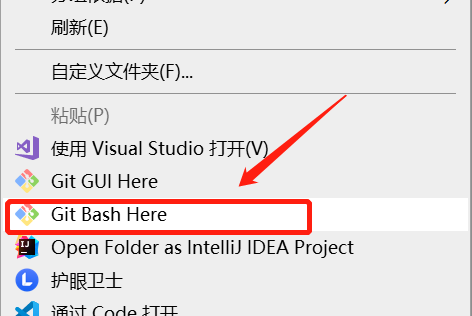


1. 说明

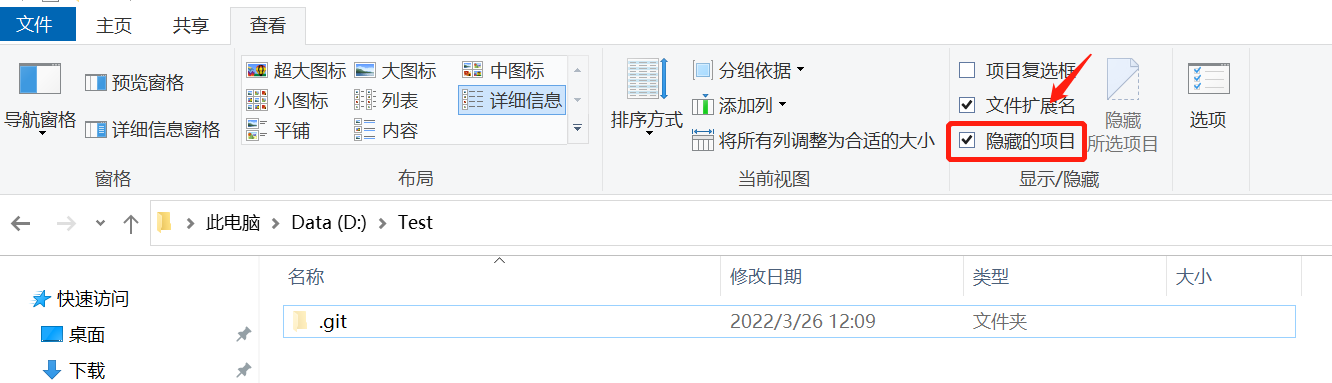
仓库默认分支为main，直接选择即可。也可自己重新创建新分支，但push完后需要在网页上多一步merge操作。

1. 命令行使用方法

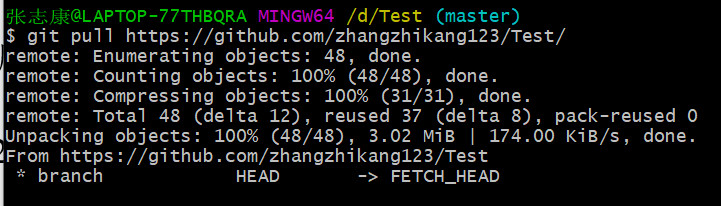
第一步：新建文件夹起名为Test，进入文件夹中右键Git Bash Here输入git init以此创建一个工程目录：



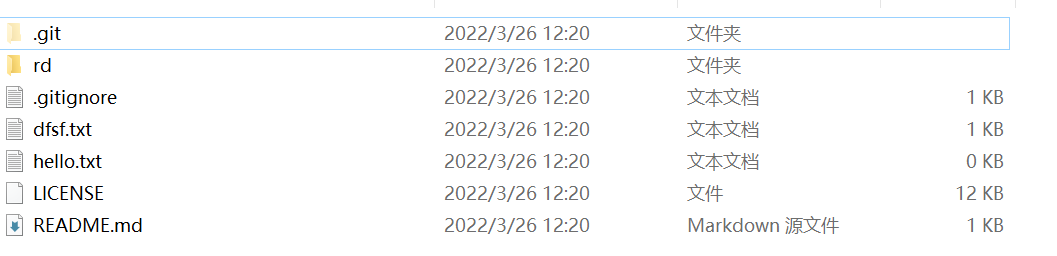
打开文件夹中隐藏的项目按钮，若出现.git文件夹，则工程目录创建成功：



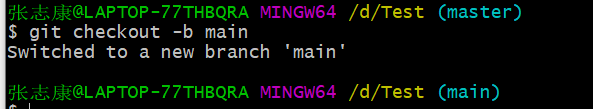
第二步：将GitHub需要编辑的仓库通过pull操作移入到你的本地仓库中，以我在GitHub中新建的Test仓库为例，通过输入git pull https://github.com/zhangzhikang123/Test/，即可将远程仓库中所有文件移入到你的本地仓库中：



上图显示远程仓库传送成功，查看你的本地仓库，此时文件也全被移了进来：

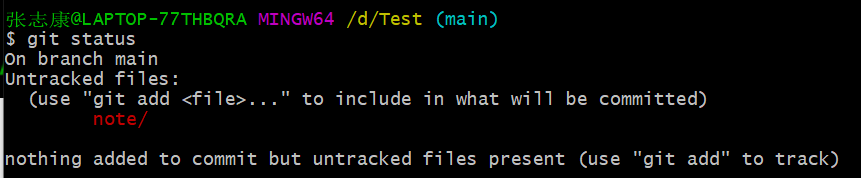


由于我想要修改的远程仓库只有一个main分支，而git中默认是master分支，所以需要将分支进行更改，通过输入git checkout -b main即可更改分支(可以将main修改为你想要更改的分支)：

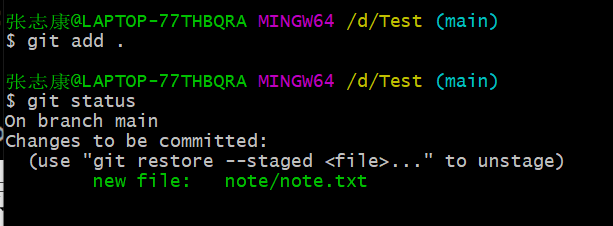


以上分支即可修改成功。

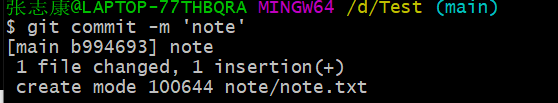
第三步：在本地仓库中新建一个note文件夹，并新建note.txt文件以作例子，将其通过命令行方式传回远程仓库。通过输入git status查看本地仓库文件状态：



可以看到提示需要你将note文件夹下的文件未被追踪，使用命令git add .添加到暂存区里面去，不要忘记后面的小数点“.”，意为添加文件夹下的所有文件，再次通过git status查看文件状态，文件已经被追踪。



第四步：把修改的内容提交到本地仓库中。用命令git commit -m ‘note’告诉Git，把文件提交到仓库。引号内为提交的说明：



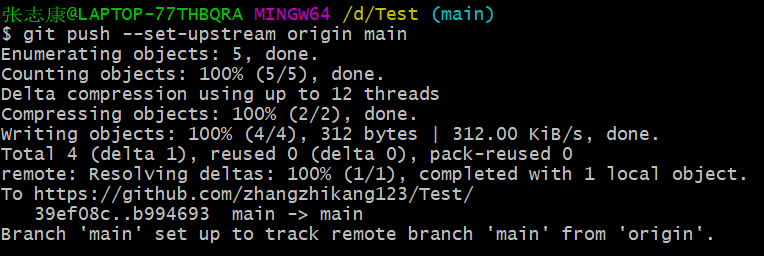
然后把修改好的本地仓库内容提交到GitHub的远程仓库中，使用命令行git remote -v可以查看并无任何显示，表示本地仓库此时未与远程仓库相关联：



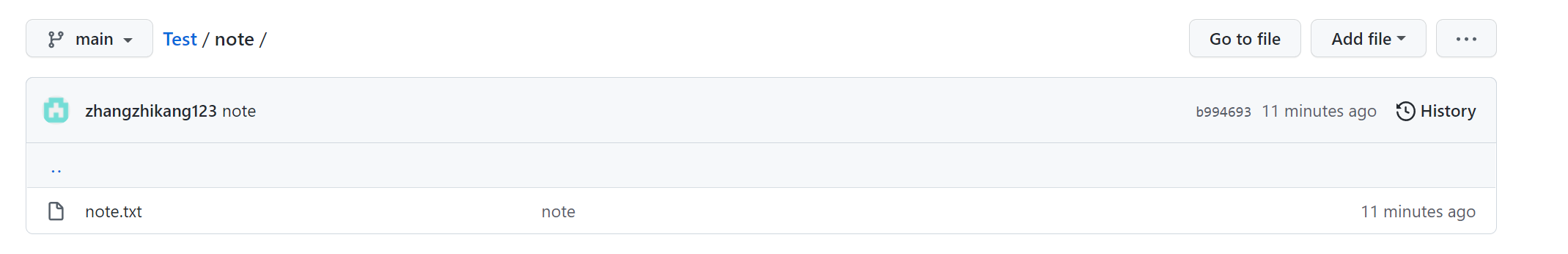
第五步：输入git remote add origin https://github.com/zhangzhikang123/Test/（origin表示你给后面仓库地址起的别名），即能关联远程仓库，再次输入git remote -v查看情况：



最后一步，输入git push --set-upstream origin main将本地仓库上传到远程仓库中，执行此命令后会要求输入用户名、密码，验证通过后即开始上传（若想要每次跳过用户名和密码输入，可以参照<https://www.jianshu.com/p/ab5977d763e9>）：

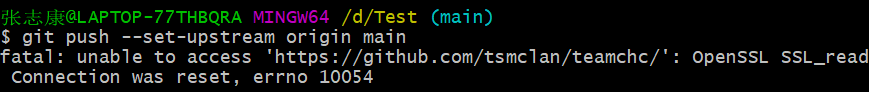


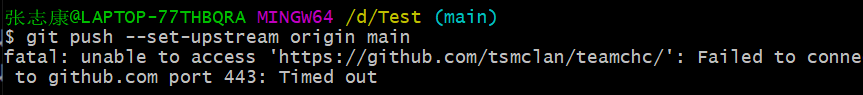
此时上传完成，在GitHub上进行查看：



远程仓库也已经出现新加入的文件。

**注意**：有时候进行pull或者push操作会出现以下两种报错情况：





由于国内GitHub连接不稳定，多重复试几次，能解决问题。

1. 文章的文件名约定

【综述】年yyyy期刊缩写\_title e.g. 2015CSR\_Crowdsourcing-A taxonomy and systematic mapping study.pdf

【期刊】年yy期刊缩写\_title e.g. 20KBS\_Batch allocation for decomposition-based complex task crowdsourcing e-markets in social networks.pdf

【会议】年yy会议缩写\_title e.g. 19www\_Leveraging Peer Communication to Enhance Crowdsourcing.pdf

【报告PPT】年yy期刊缩写\_报告人名字缩写\_title.pptx e.g. 15aaai\_wyj\_15aaai\_wyj\_Novel\ Mechanisms\ for\ Online\ Crowdsourcing\ with\ Unreliable,\ Strategic\ Agents.pptx

【阅读笔记】年yy期刊缩写\_notes\_关键词v版本号.docx e.g. 19OE\_notes\_Object-independent\ image-based\ wfs.docx

【实验报告】yy年mm月exp\_报告人名字缩写\_关键词v版本号.docx e.g. 2202Oexp\_报告人名字缩写\_专家核提取v1.0.docx。