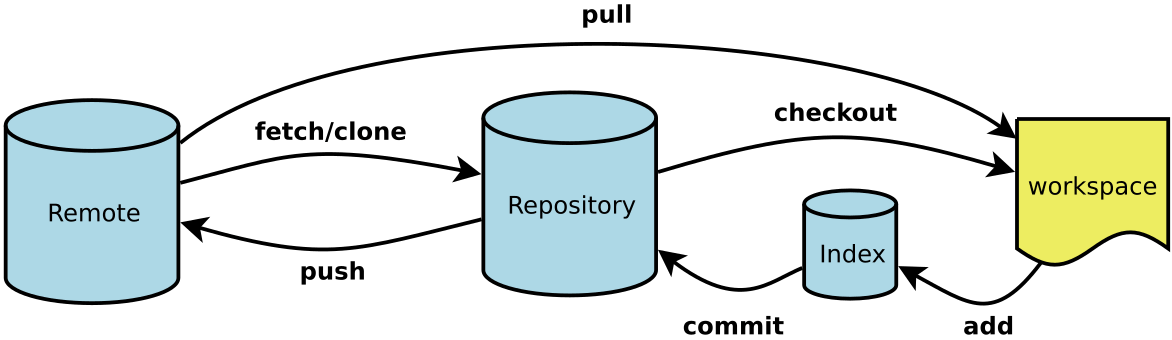
1. Git是目前世界上最先进的分布式版本控制系统。
2. 工作原理



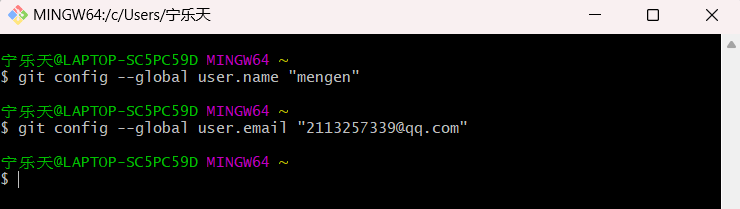
其中workspace是工作区，Index/stage是暂存区，Repository是仓库区，Remote是远程仓库

1. SVN与Git最主要区别：

SVN是集中式版本控制系统，版本库是集中放在中央服务器的，Git是分布式版本控制系统，那么它就没有中央服务器的。

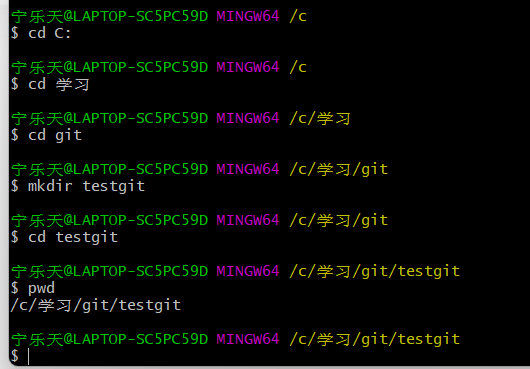
1. windows安装git

安装exe后找到git bash，如果弹出一个类似的命令窗口的东西，就说明Git安装成功，安装成功后指定用户名和邮箱完成设置。

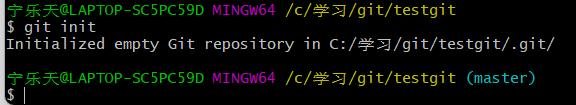


1. 创建版本库：

先cd进指定目录后mkdir name

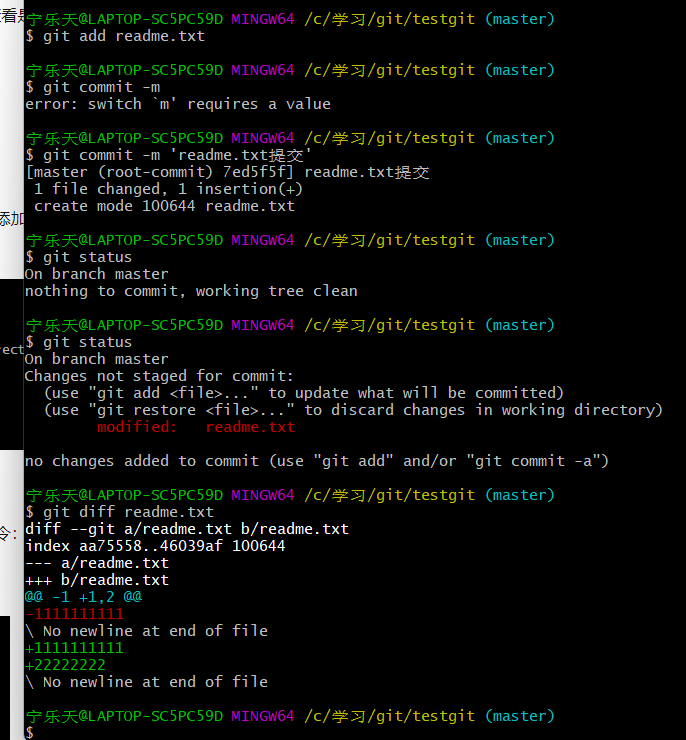


1. 通过命令 git init 把这个目录变成git可以管理的仓库。



1. 把文件添加到版本库中：

先使用命令 git add readme.txt添加到暂存区里面去，然后用命令 git commit告诉Git，把文件提交到仓库。



1. git log命令显示从最近到最远的显示日志，我们可以看到最近三次提交。



1. git reset --hard HEAD^ 那么如果要回退到上上个版本只需把HEAD改成HEAD以此类推。那如果要回退到前100个版本的话，使用上面的方法肯定不方便，我们可以使用下面的简便命令操作：git reset --hard HEAD~100 即可。

**理解工作区和暂存区：**

工作区：工作区是你在本地计算机上看到和编辑的实际文件。你可以在工作区中创建、修改和删除文件。工作区中的文件可能包括已跟踪的文件（已添加到Git仓库）和未跟踪的文件（未添加到Git仓库）。

暂存区：暂存区是Git仓库的一部分，用于存储即将提交到版本控制的更改。在提交更改之前，你需要将工作区中的更改先添加到暂存区。暂存区是一个缓冲区，类似于一个临时存储区域，可以帮助你准备要提交的更改。

**Git撤销修改和删除文件操作：**

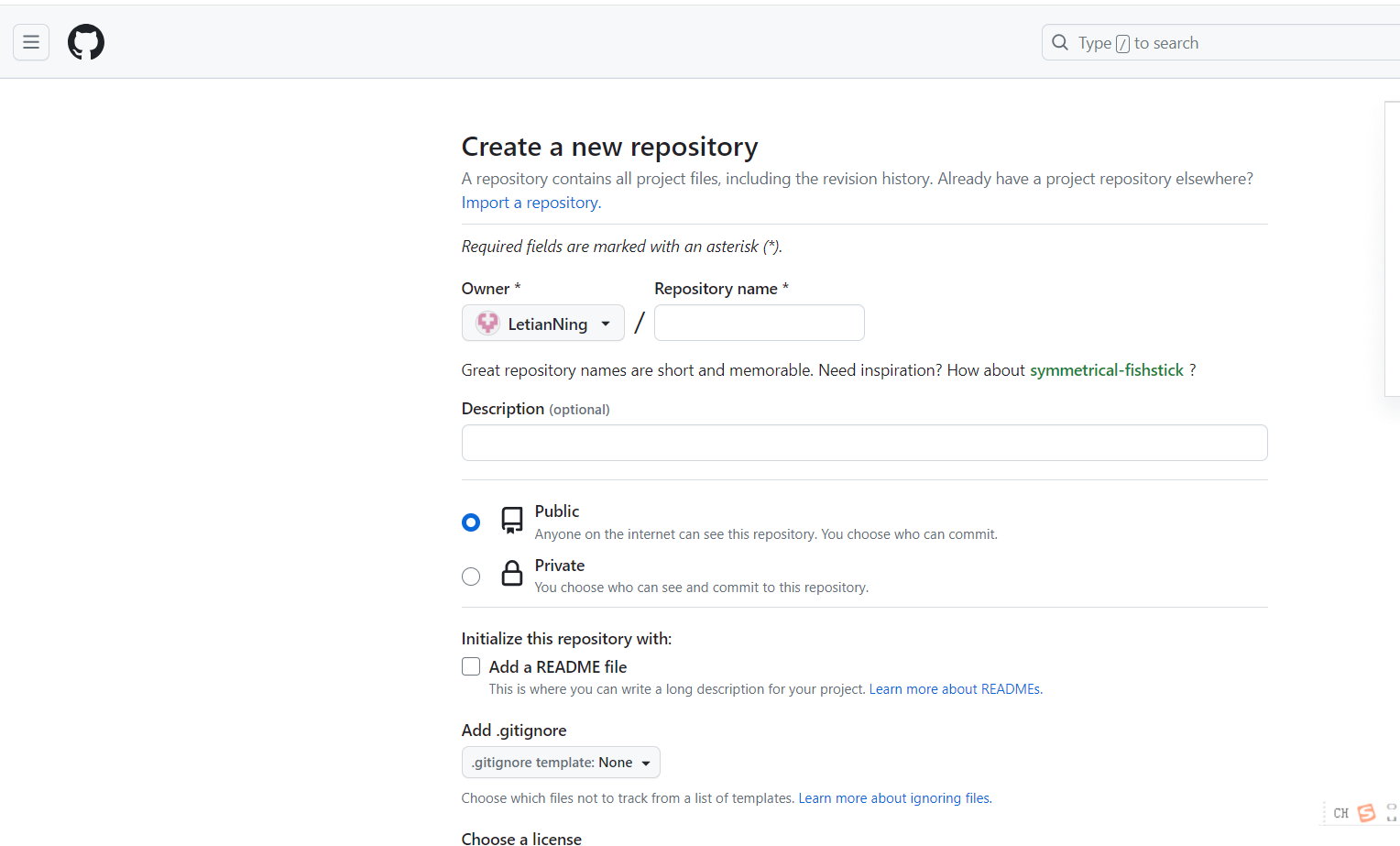
1. 如果我知道要删掉那些内容的话，直接手动更改去掉那些需要的文件，然后add添加到暂存区，最后commit掉。
2. 如果你对文件进行了修改，但尚未将修改添加到暂存区，可以使用命令git checkout -- <file>来撤销对该文件的修改。这将会将文件还原到最近一次提交的状态。
3. 如果你已经将修改添加到暂存区，但还没有提交，可以使用命令git reset HEAD <file>来将暂存区中的修改撤销，将文件还原为最近一次提交的状态。然后可以使用git checkout -- <file>来撤销对文件的修改。

**删除文件：**

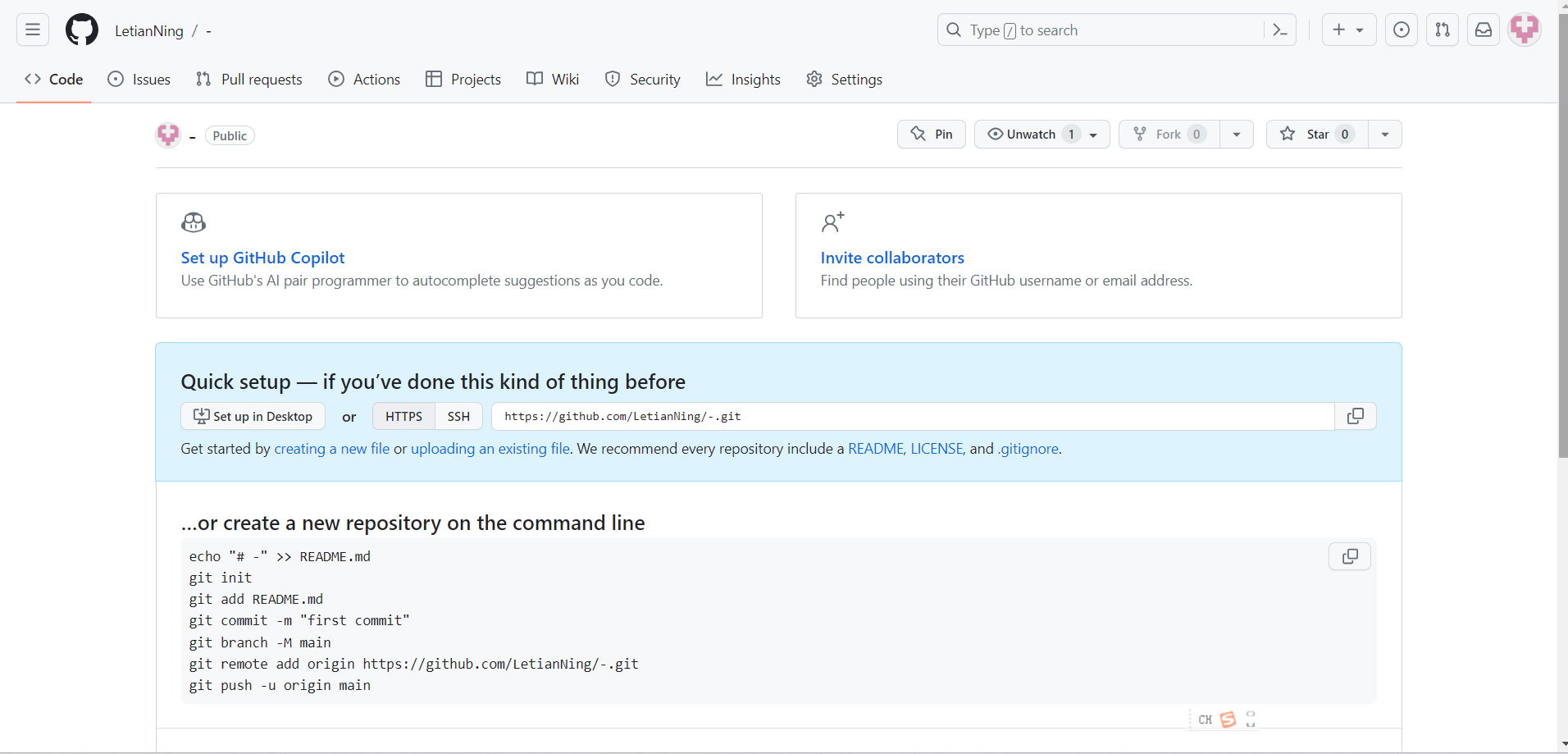
对于一个文件b.txt而言，一般情况下，可以直接在文件目录中把文件删了，或者使用如上rm命令：rm b.txt ，如果我想彻底从版本库中删掉了此文件的话，可以再执行commit命令 提交掉。在没有commit之前，如果我想在版本库中恢复此文件可以使用如下命令 git checkout -- b.txt

**添加远程仓库：**

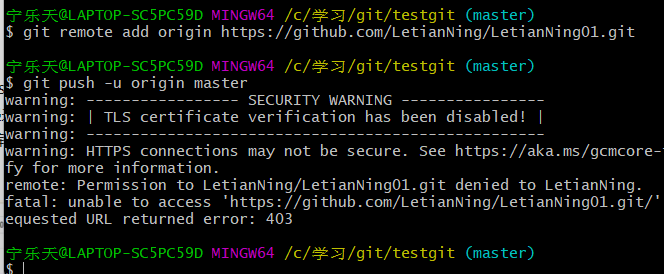
登录github上，然后在右上角找到“create a new repo”创建一个新的仓库。填入仓库名，其他默认即可创建成功



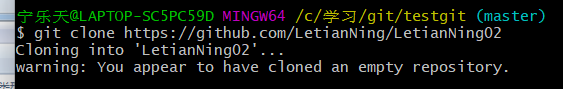
**创建成功的一个新的仓库：**



**本地仓库关联远程仓库：**



**克隆：**

****

**创建与合并分支：**

查看分支：git branch

创建分支：git branch name

切换分支：git checkout name

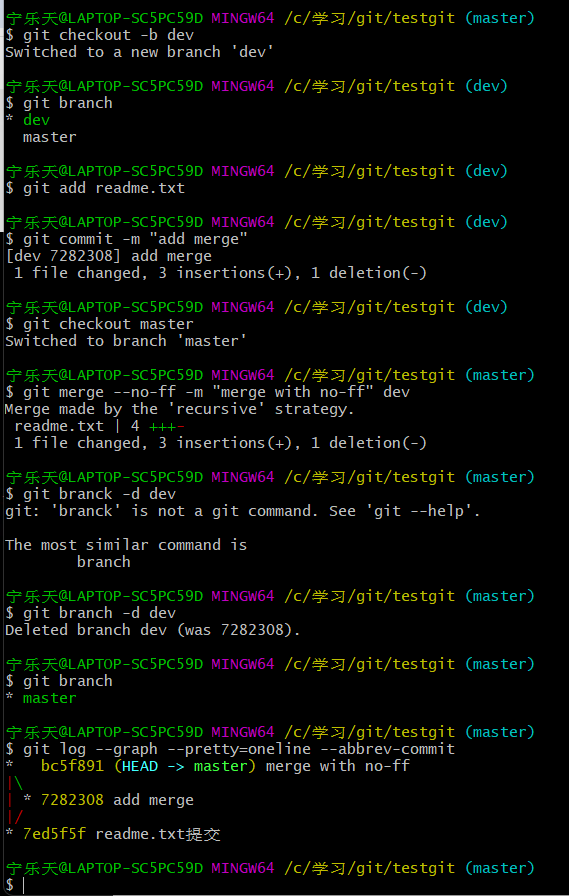
创建+切换分支：git checkout –b name

合并某分支到当前分支：git merge name

删除分支：git branch –d name

**分支管理：**

master主分支应该是非常稳定的，也就是用来发布新版本，一般情况下不允许在上面干活，干活一般情况下在新建的dev分支上干活，干完后，比如上要发布，或者说dev分支代码稳定后可以合并到主分支master上来。

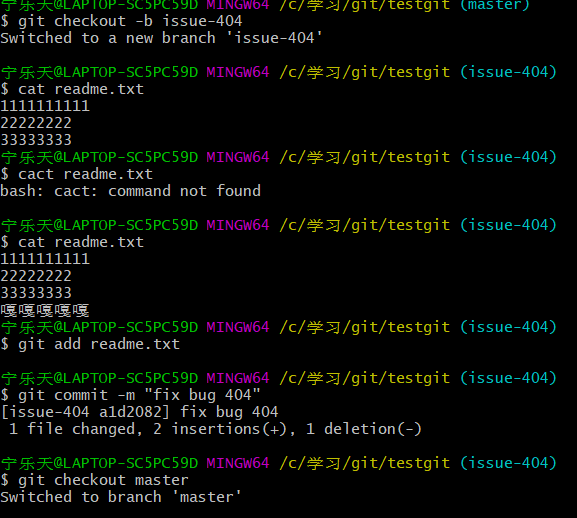


**BUG分支：**

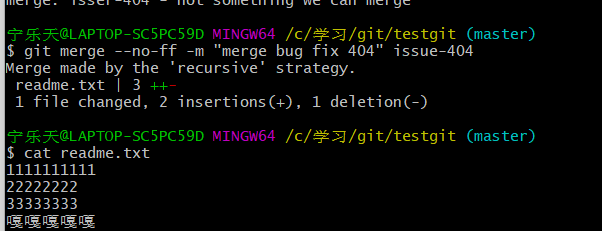
工作现场隐：git stash

工作现场恢复：git stash apply恢复，恢复后，stash内容并不删除，你需要使用命令git stash drop来删除。另一种方式是使用git stash pop,恢复的同时把stash内容也删除了。

新建一个issue-404分支用来解决BUG



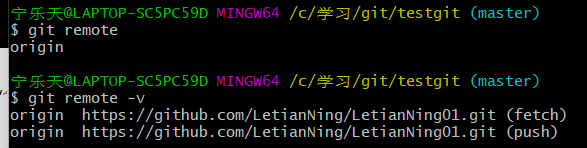
转化为主分支后合并分支issue-404:



**多人协作：**

要查看远程库的信息 使用 git remote

要查看远程库的详细信息 使用 git remote –v



推送分支：

命令：git push origin master，如果需要push到其他分支如dev分支，则使用git push origin dev

在多人协作中，当多个人在同一文件的相同部分进行更改时，就会发生冲突。解决冲突需要合并双方的更改，通常需要手动编辑文件。在解决冲突时，需要仔细审查并确保合并的代码是正确的。Git会标记冲突的部分，你需要手动编辑这些部分来解决冲突。在合并操作时，在将分支的更改合并到主分支之前，应仔细审查和测试代码。确保在合并之前运行适当的测试套件，并验证合并后的代码的正确性。避免将有问题的代码合并到主分支。