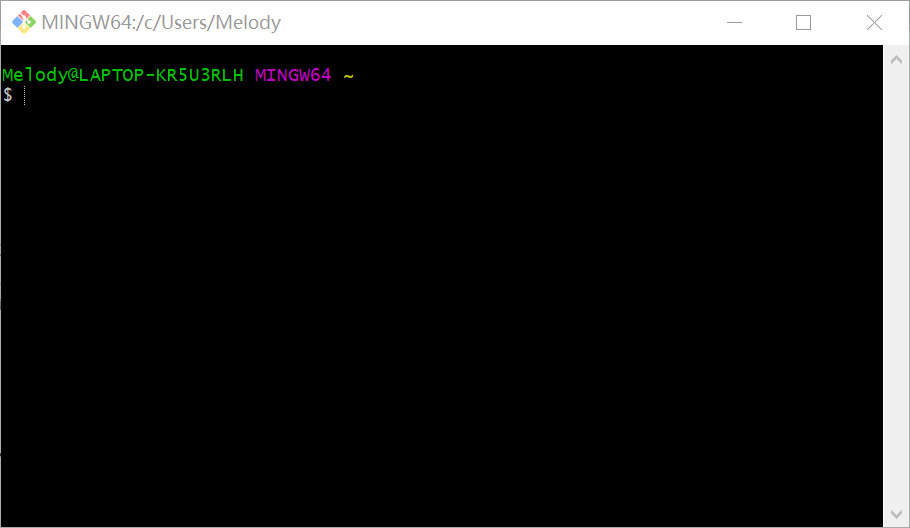
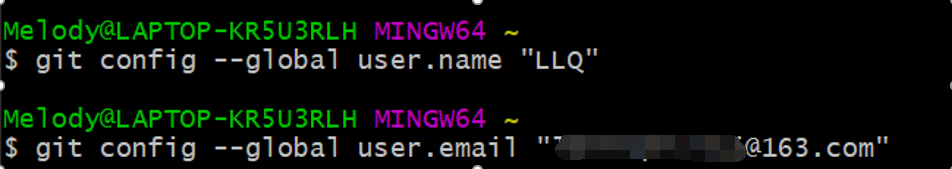
实验四、基于GIT的版本管理

一、安装

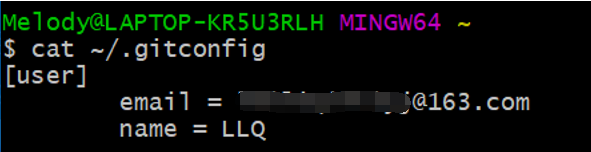
* 1. 在Windows上使用Git，可以从Git官网直接<https://git-scm.com/download/>，然后选择安装路径，按默认选项安装即可。
  2. 安装完成后，在开始菜单里找到“Git”->“Git Bash”，跳出一个类似命令行窗口，说明Git安装成功。



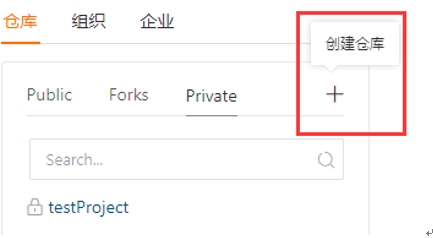
* 1. 安装完成后，需要设置你的名字和email，这些就是你在提交commit时的签名。使用git config命令进行配置。注意git config命令的--global参数，用了这个参数，表示你这台机器上所有的Git仓库都会使用这个配置，当然也可以对某个仓库指定不同的用户名和Email地址。



执行上面的命令后，会在目录下建立一个.gitconfig文件，使用vim或cat查看文件：

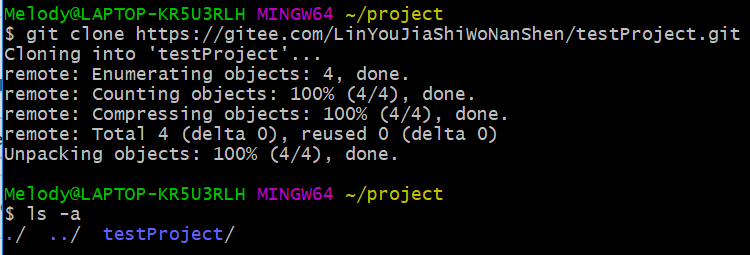


1. 注册Gitee账号<https://gitee.com>，并新建一个远段仓库。这里我新建了一个，名为“testProject”的远端仓库。



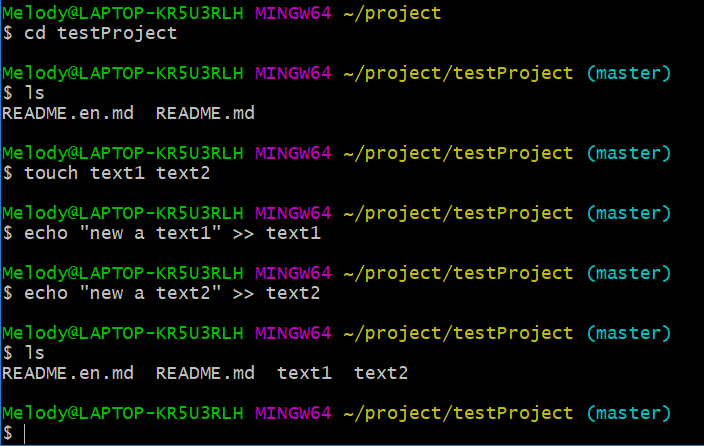
1. Git基本操作

1、 使用git clone [远端git仓库] 将新建的项目克隆到project 目录下。

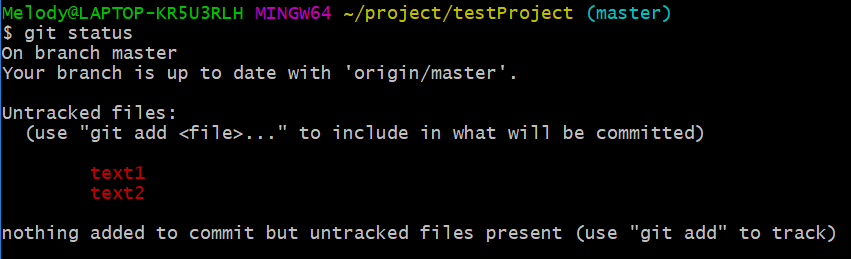


2、 进入刚才textProject目录，新建文件text1和text2。

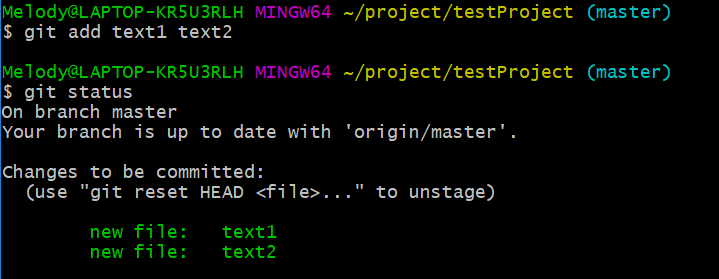
修改文件，使用vim编辑内容，也可以直接echo添加内容。



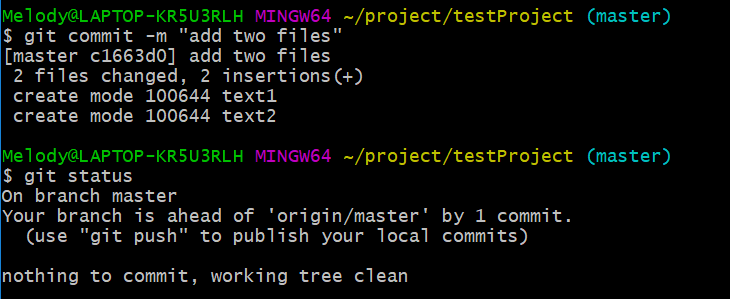
3、 使用git status命令显示有变更的文件，可以看到有2个文件处于untracked状态。



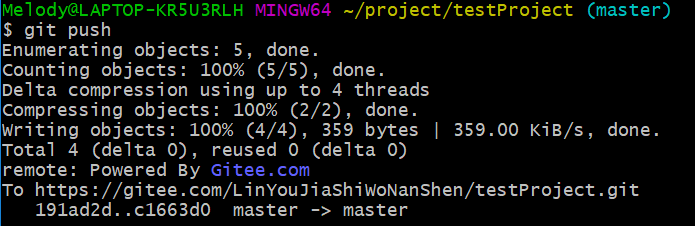
4、 使用git add命令将处于untracked状态的2个文件加入到缓存区（Index）。再次执行git status会发现新的变化。

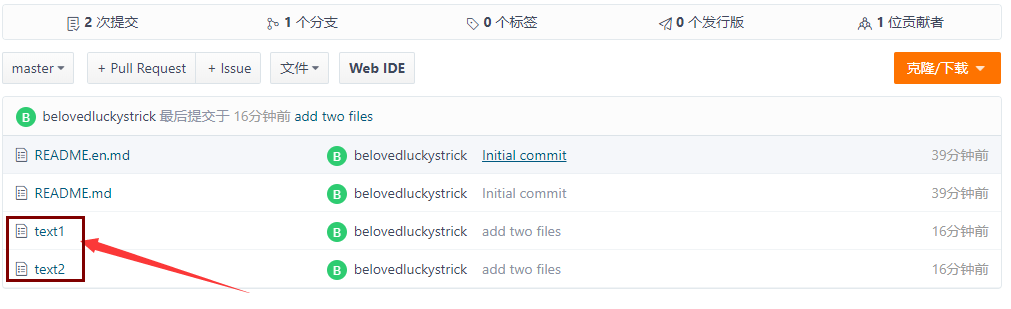


5、 现在已经为commit到本地仓库做好了准备。（可以使用git diff--cached命令查看缓存区和上一次commit差异，看哪些文件被修改了。）使用 git commit -m “本次提交内容的摘要” 命令将缓存区文件提交到本地仓库。再次执行git status会发现新的变化。



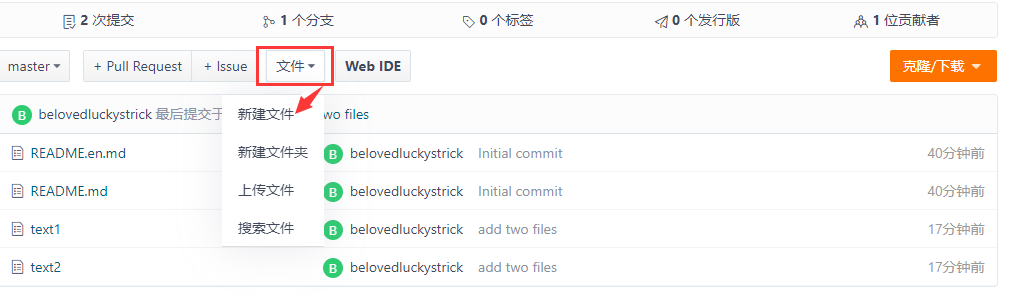
6、 使用git push命令将本地仓库的文件更新到远端仓库中。可以看到gitee的项目已新增text1和text2文件。



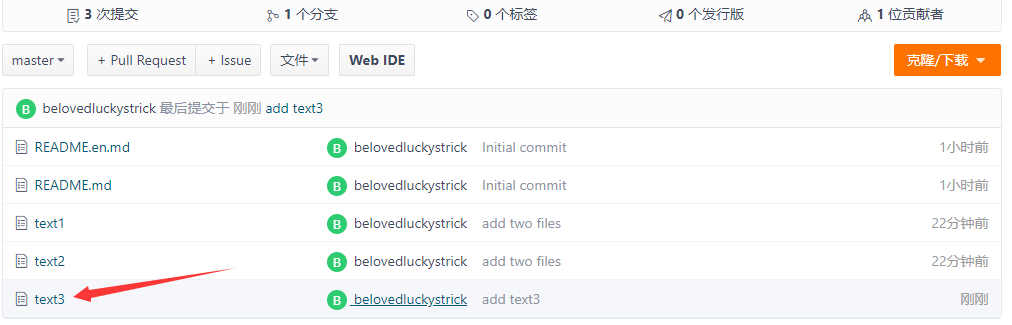


7、 一般一个项目会有多个成员进行操作，当其他成员新增代码文件或者改动代码文件并push到远端仓库后，远端仓库的进程会比本地仓库快。这是需要把远端新增文件git pull到本地。

这里，我们手动在gitee的testProject项目中新建一个文件text3 来模拟其他成员push到远端的代码文件。text3的文件内容可以任意添加。

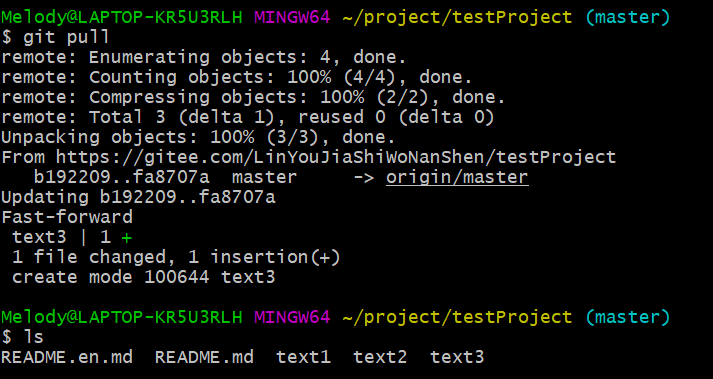






这个时候远端仓库进程比本地仓库快。

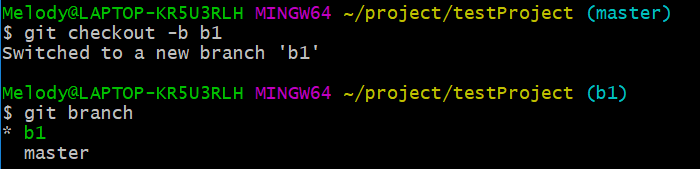
使用git pull 命令（或者先git fetch 再 git checkout）同步远端仓库的文件到本地。



8、 分支与合并。

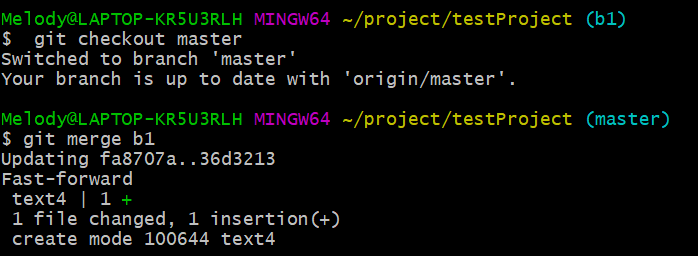
Git的分支可以让你在主线（master 分支）之外进行代码提交，同时又不会影响代码库主线。分支的作用体现在多人协作开发中，比如一个团队开发软件，你负责独立的一个功能需要一个月的时间来完成，你就可以创造一个分支，只有把该功能的代码提交到这个分支，而其它成员仍然可以继续使用主线开发，你每天的提交不会对他们造成任何影响。当你完成功能后，测试通过在把你的功能分支合并到主线上。

* 1. 步骤1-7都是对主线（master分支）进行操作的。这里，使用git checkout -b [分支名] 命令新建一个分支并跳转到这个分支目录下。使用git branch 查看所有本地分支。



* 1. 在b1分支下，创建并写入文件text4，并重复上述操作，git add—>git commit，将text4 commit到本地仓库。

使用git checkout master切换到主线上，即，使合并分支操作的目标分支是master分支，使用git merge [源分支] 命令将分支b1合并到master分支上。



* 1. 再次 git push到远端。查看远程仓库gitee已更新。

