**一、安装：**

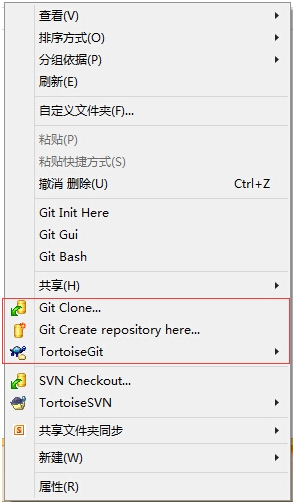
1. 根据安装环境，选择TortoiceGit，32位或者64位(不习惯英文的朋友也可以下个语言包)。



1. 先安装Git，安装好后，右键，会发现菜单中多了几项关于GIT的选项



1. 再安装tortoisegit，安装好以后，右键，会发现菜单中多了几项关于tortoisegit的选项。



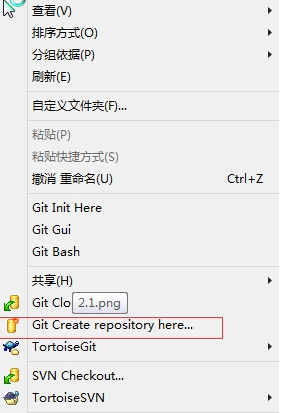
到此，安装完成。

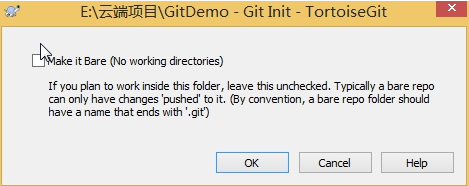
**二、使用**

简介：git属于分布式版本控制器，其实每个人的电脑都是一个完整的版本库；比如两个人在局域网写一个项目，那么这两个人的电脑里面都是一个完整的项目，当有新的代码出现的时候，直接push给对方即可。当然人多的时候，又不在局域网里面，不可能push给每一个人，git的解决方案是用一台电脑当做集中的版本库，大家把新代码都push到上面去，要得时候在pull到自己电脑上。国外比较著名的就是github了,国内的oschina也广为人知。

创建一个版本库

创建一个文件夹GieDemo，然后 右键-Git Create repository here……





此处不用勾选Make is Bare



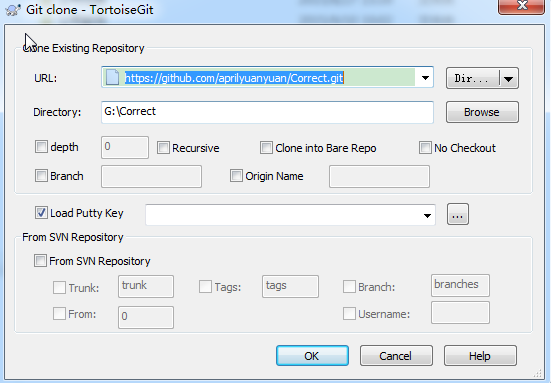
确定后，在GitDemo文件夹下回多一个.git文件夹【电脑要设置为显示隐藏文件夹】

此文件夹，尽量不要乱修改或删除。

1. 拉取
2. 首先得到希望拉取程序的目标地址，右键，选择Git Clone……



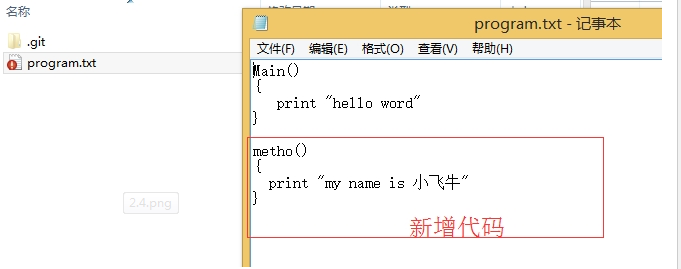
2）粘贴目标地址到URL上，点击ok



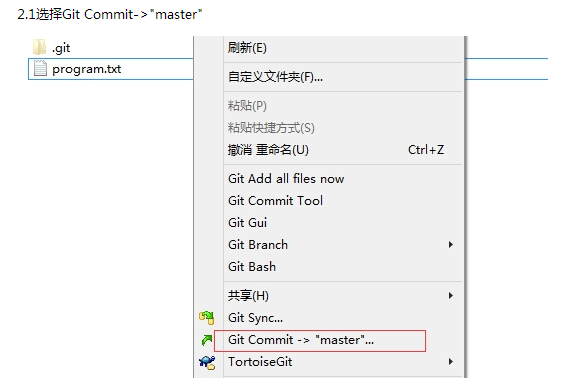
则可以将目标代码拉取下来。如下图



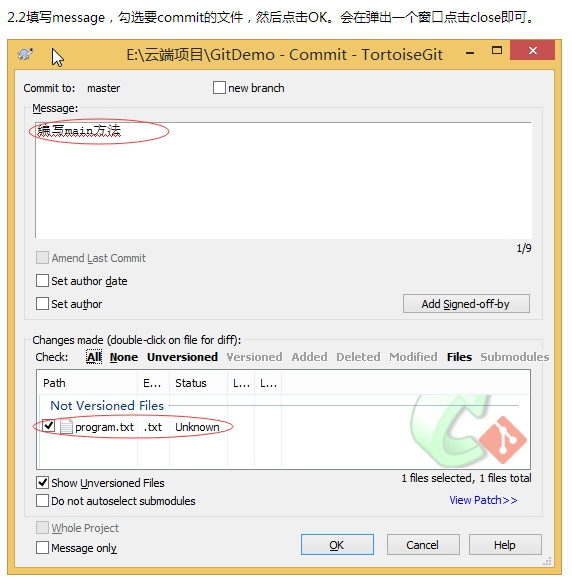
1. 更改代码时，可以直接在拉取下来的文件中更改，如更改LICENSE文件中的内容后，文件上会出现红色的感叹号。



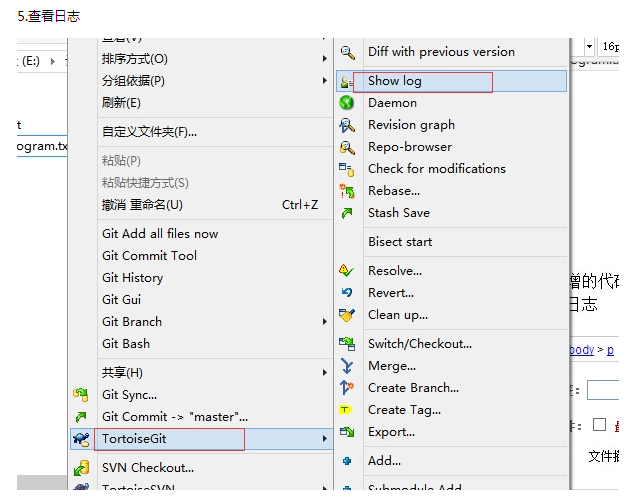
1. 将新增的代码提交到版本库中，步骤如下：

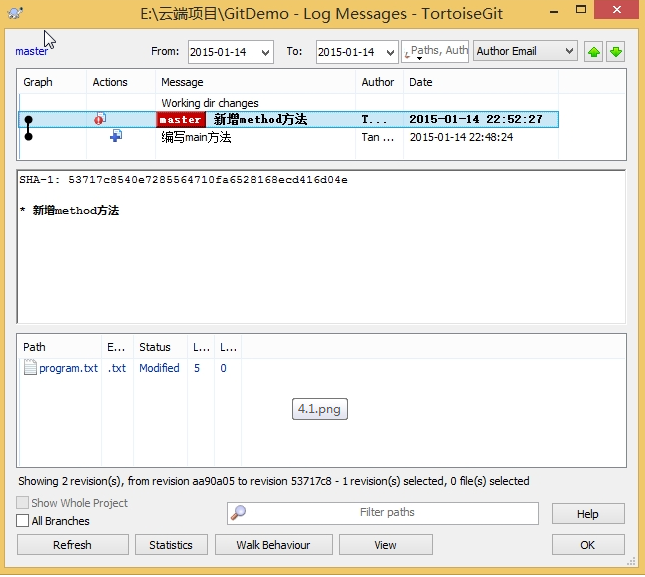


在Message的内容框中可以填写所做的更改内容。



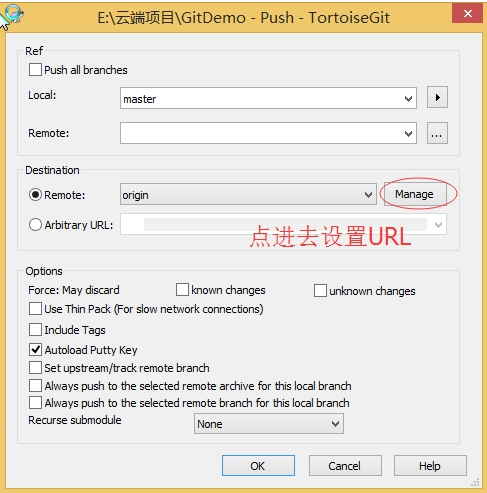
1. 查看日志，在日志中，可以很清楚地看到文件所做的更改。



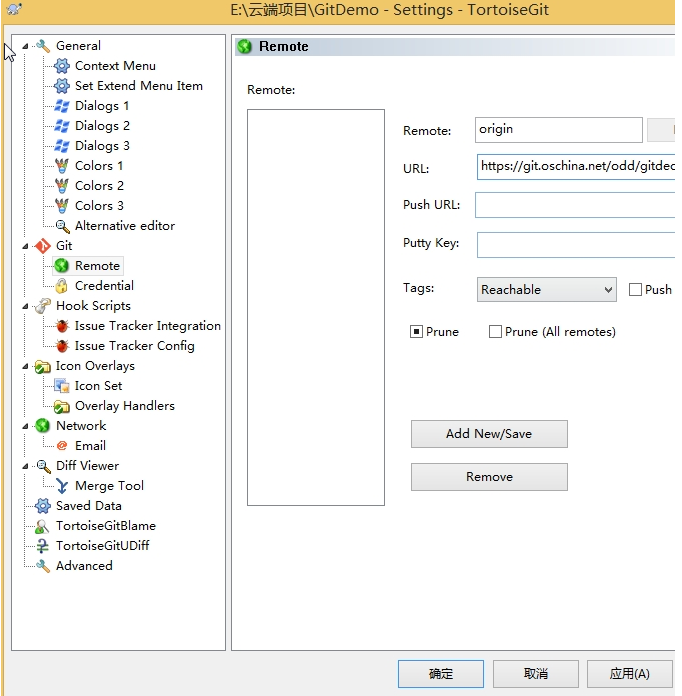


1. 将电脑的代码推送到云端

右键—>tortoiseGit 🡪push



设置好URL，确定。然后开始push，输入用户名，输入密码就OK了，然后等待成功。



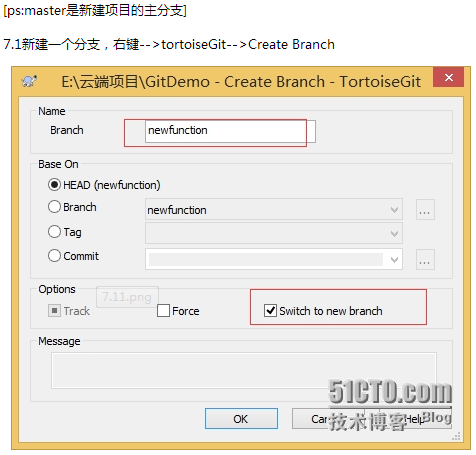
6）成功后，我们去oschina上看看push的项目吧

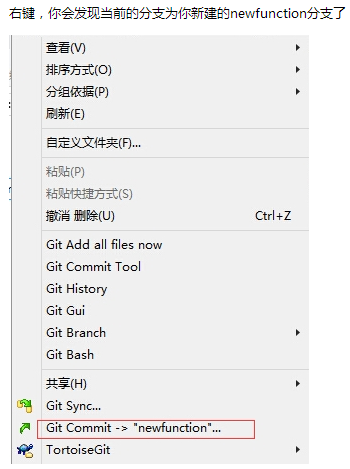


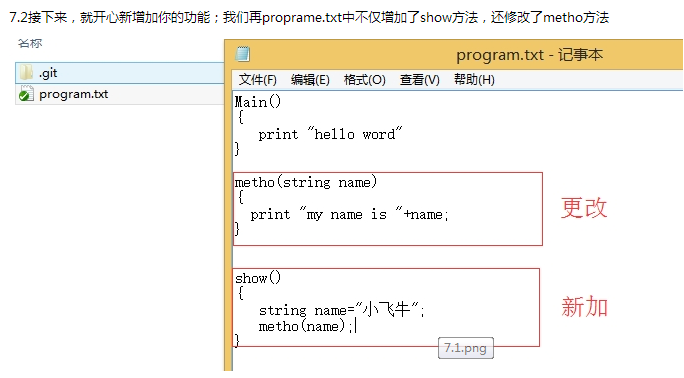
7）git还有个特别好用的功能，分支。

发现问题：你代码写了很多，运行OK；但是突然想加个新功能进去，这个功能你也不知道能否正常运行，而且修改过程中，除了新加代码和文件进去，还会修改以前的代码。要是万一失败，修改回来也是一种很麻烦的事情。这种时候很多人就用备份方式。来看看git是怎么优雅的处理这个问题的。

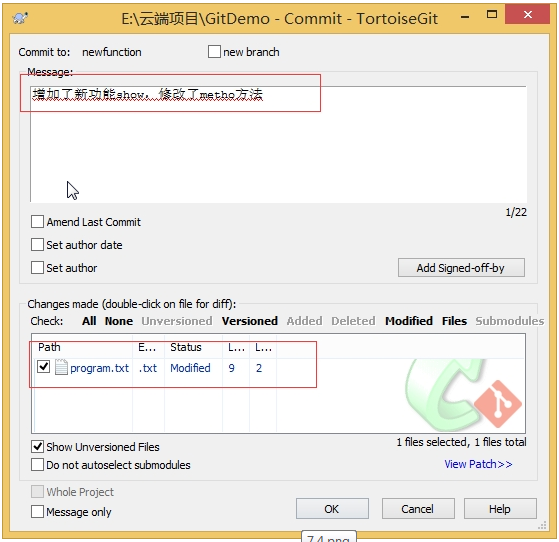
git的处理方式：当你想加一个新功能进去的时候，你可以新建一个分支，例如名字叫newfunction，然后在分支中把新功能加上去，如果OK，将代码合并到master分支上，如果新功能失败，切换回master分支上来，在newfunction写的代码，又全看不到了。



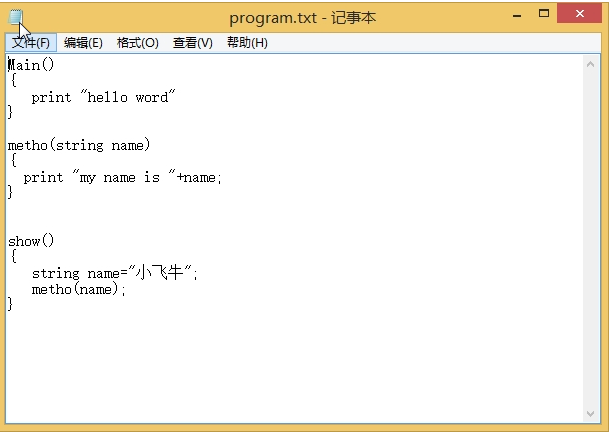




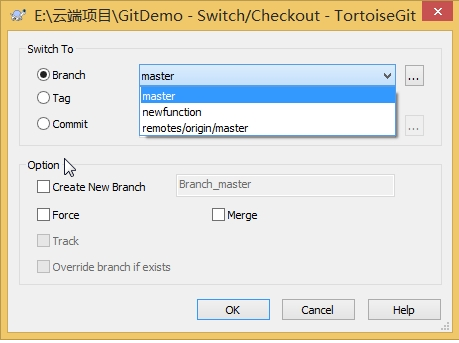
7.3提交我们的代码，此时我们发现这个新功能不行，存在很多漏洞等等问题。我们想回到之前的代码。怎么办呢？



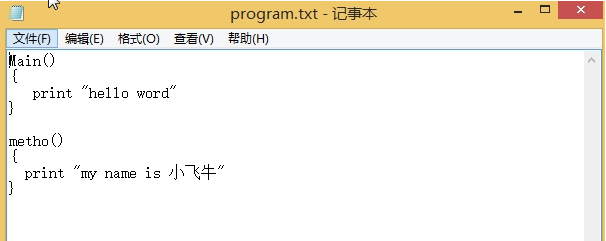
7.4为了回到之前的项目，我们直接切回master分支。先来看看现在的文件内容，新加的和修改的都在。



然后我们切换到master分支，右键-->tortoiseGit-->Switch/Checkout，选择master即可，然后确定。



再来看看program中的代码是否回到了从前呢？



无论是修改，还是添加，都没影响到复原。这就是分支的作用。这个功能是不是让你省去了很多工作量呢。