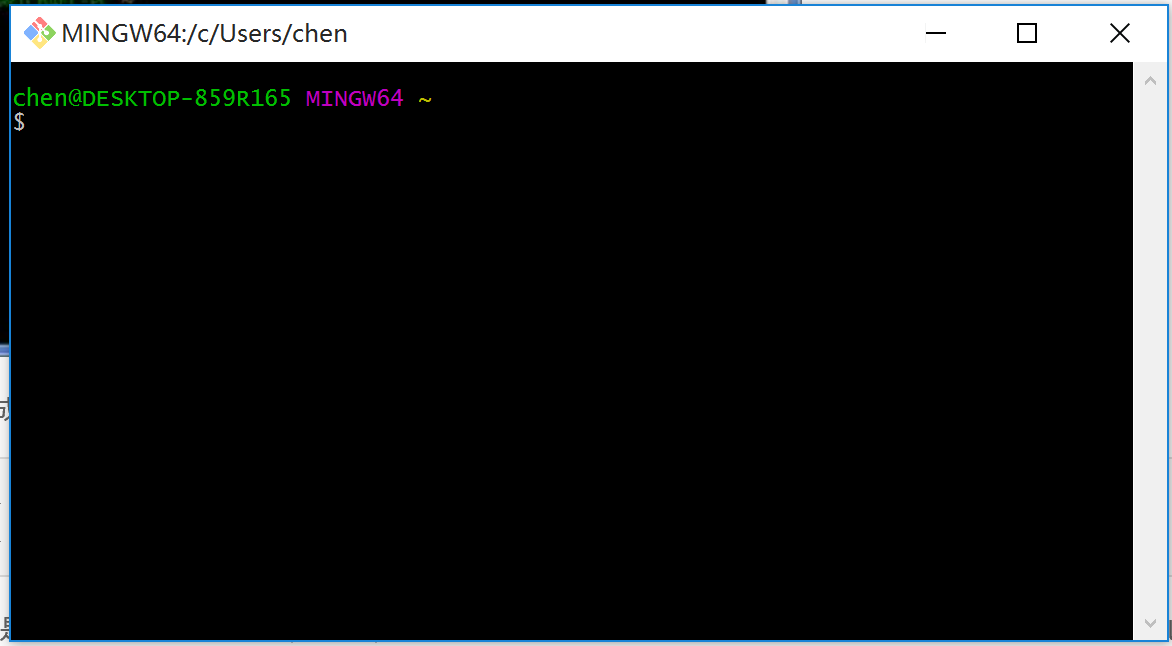
**GIT使用教程**

**GIT安装**

Windows下要使用很多Linux/Unix的工具时，需要Cygwin这样的模拟环境，Git也一样。Cygwin的安装和配置都比较复杂，就不建议你折腾了。不过，有高人已经把模拟环境和Git都打包好了，名叫msysgit，只需要下载一个单独的exe安装程序，其他什么也不用装，绝对好用。

msysgit是Windows版的Git，从<http://msysgit.github.io/>下载，然后按默认选项安装即可。

安装完成后，在开始菜单里找到Git --> Git Bash，蹦出一个类似命令行窗口的东西，就说明Git安装成功！

****

安装完成后，还需要最后一步设置，在命令行输入：

$ git config --global user.name "Your Name"

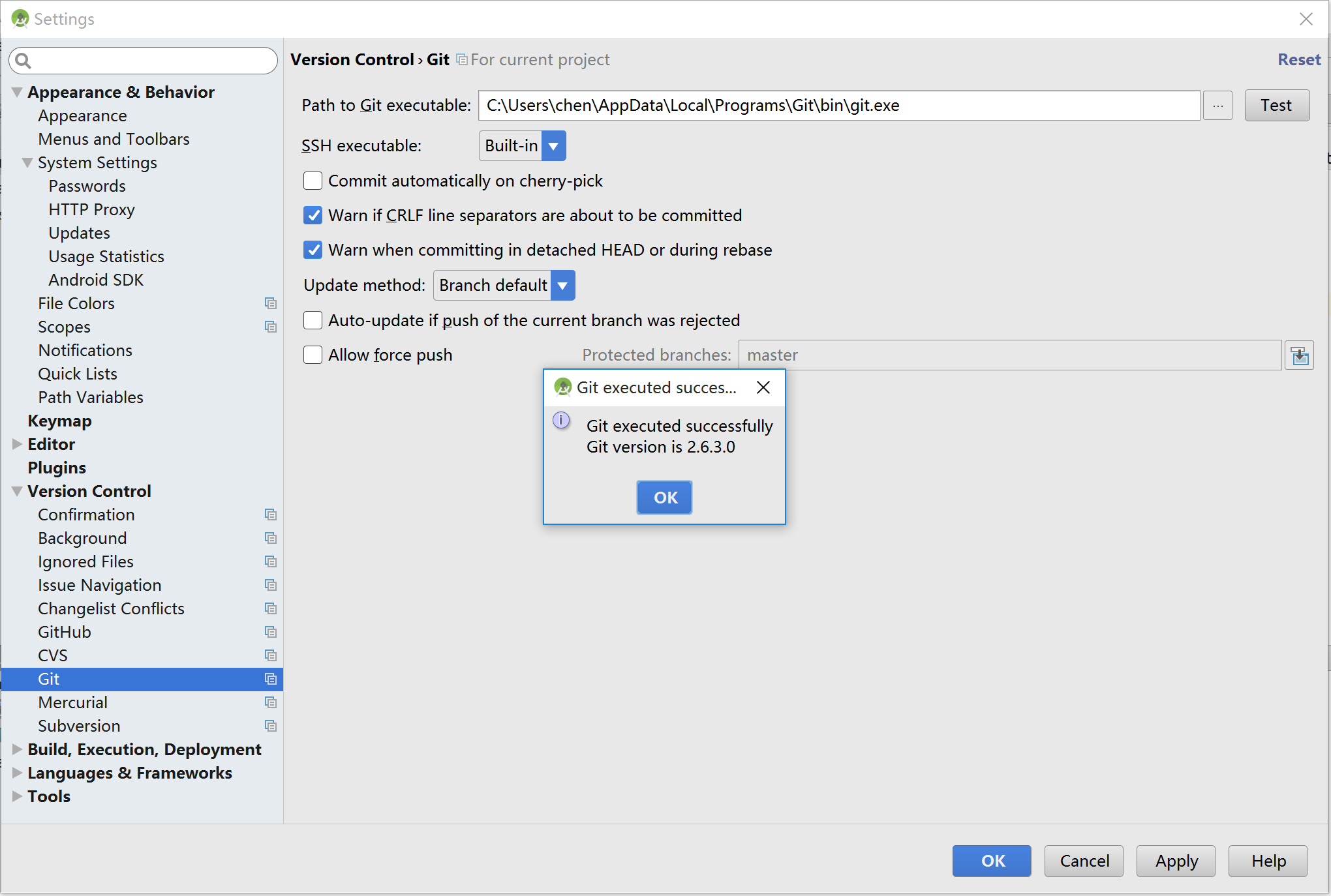
$ git config --global user.email "email@example.com"

因为Git是分布式版本控制系统，所以，每个机器都必须自报家门：你的名字和Email地址。

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。

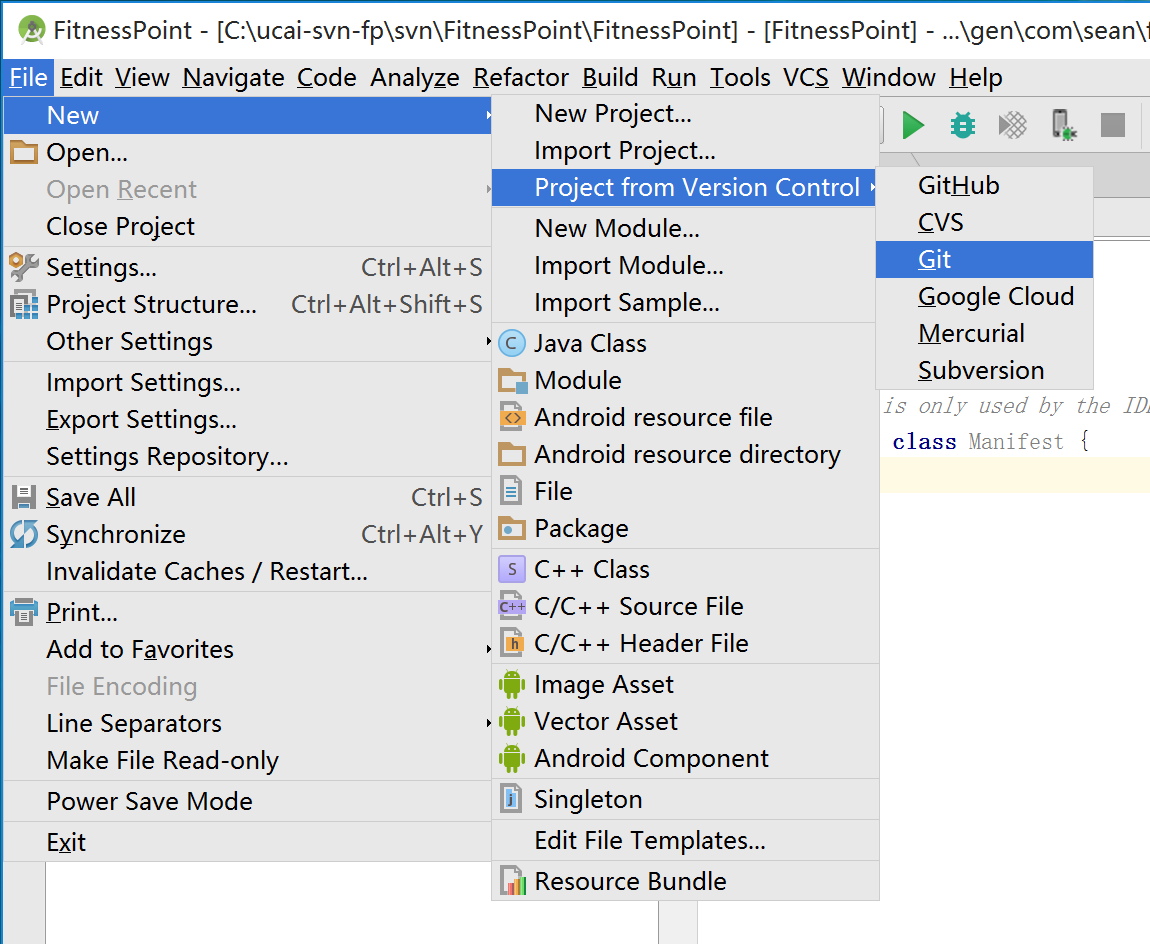
**在Android Studio中配置Git**

在Android Studio中，找到菜单项File --> Settings --> Version Control --> Git，配置Git程序目录，点击右侧的【Test】按钮来测试配置是否成功。

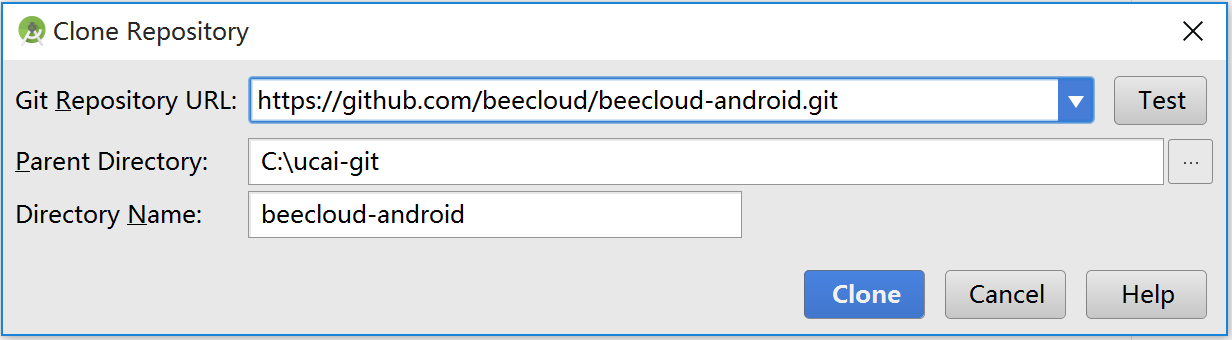
****

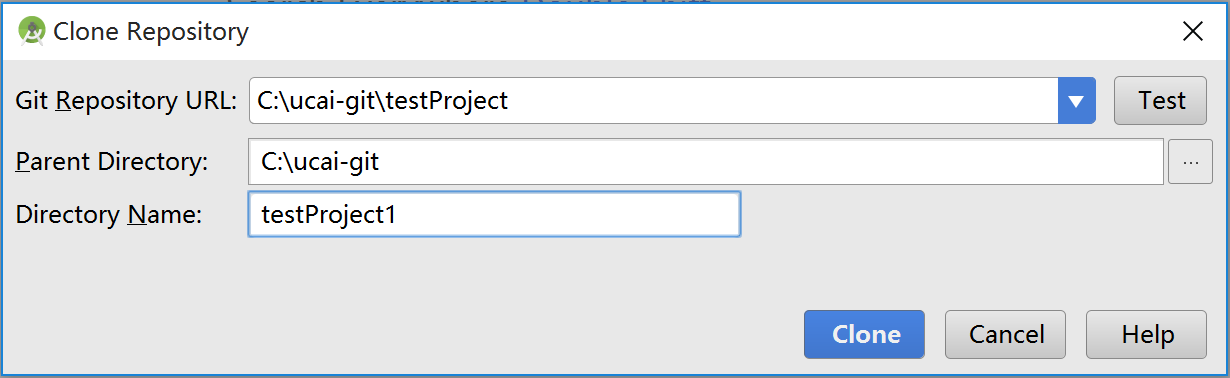
**在Android Studio中使用Git导入项目**

在Android Studio中，找到菜单项File --> New --> Project from Version Control --> Git，如图：

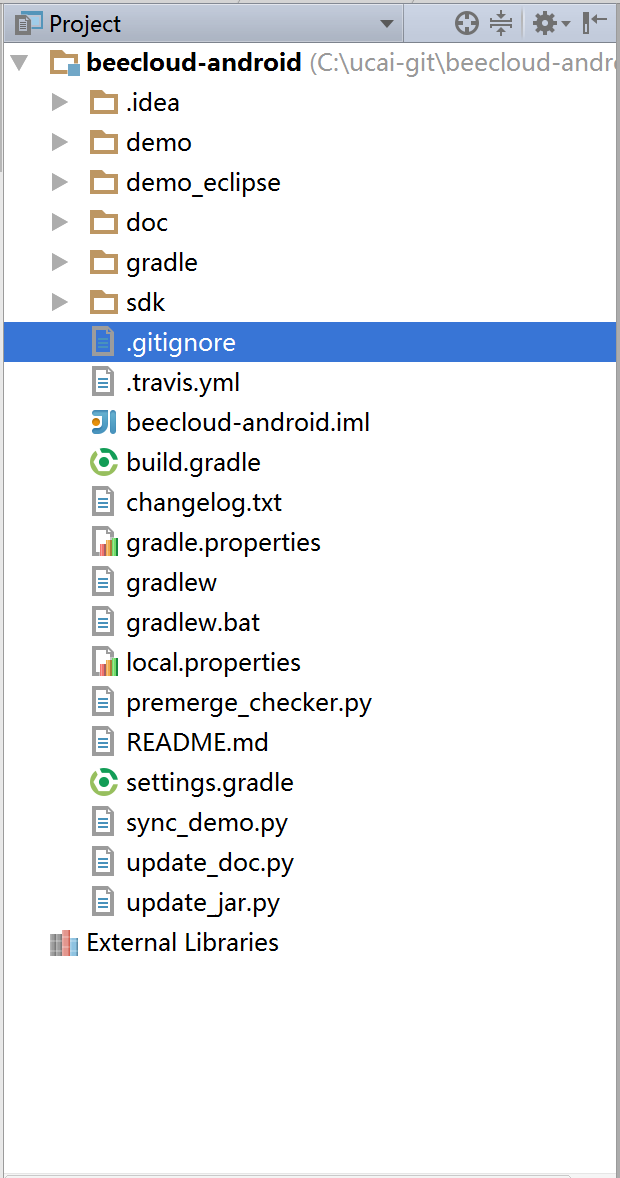
****

填写git仓地址，可以是远程或者网上的，也可以是本地或者局域网中共享的，如图：

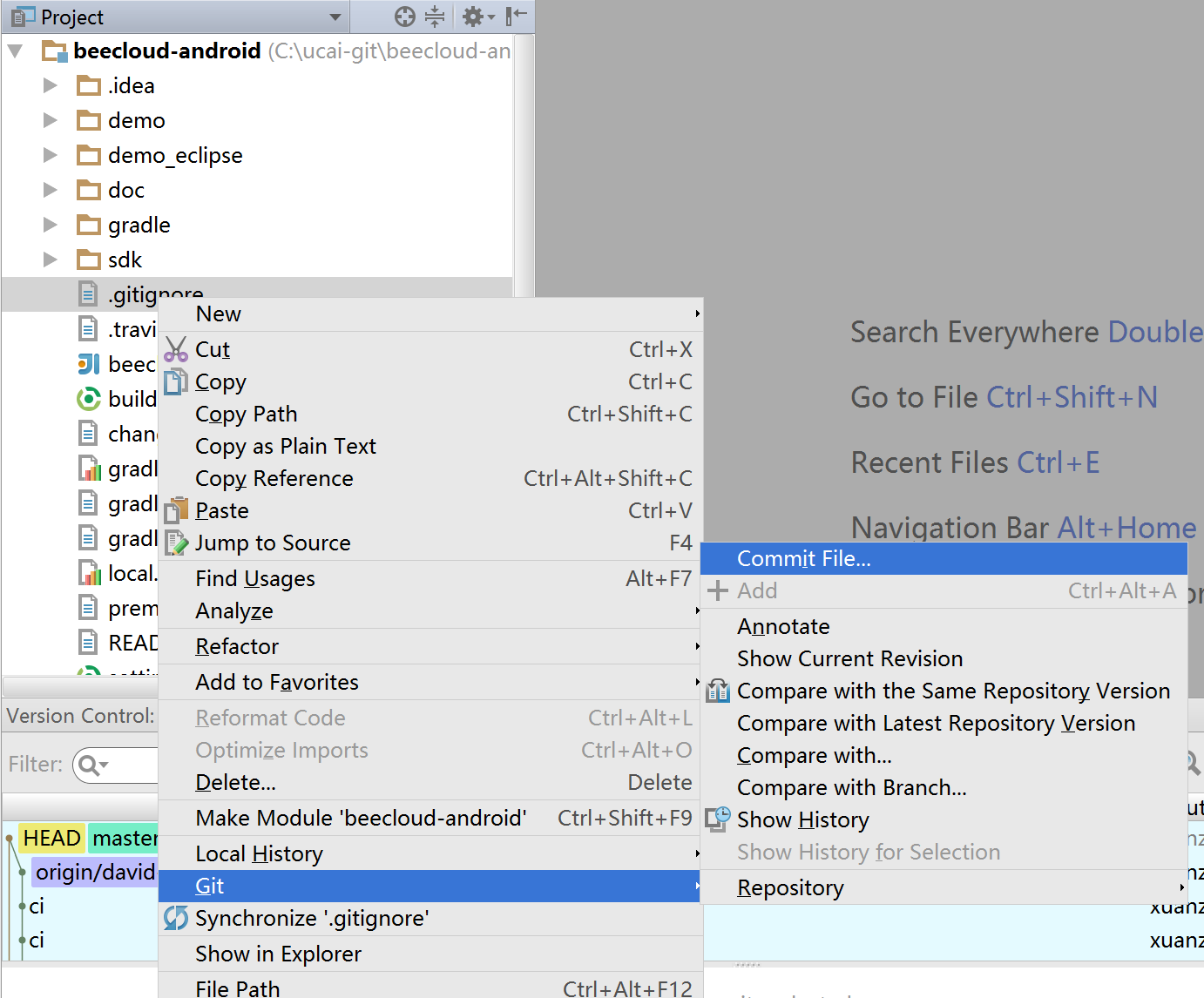




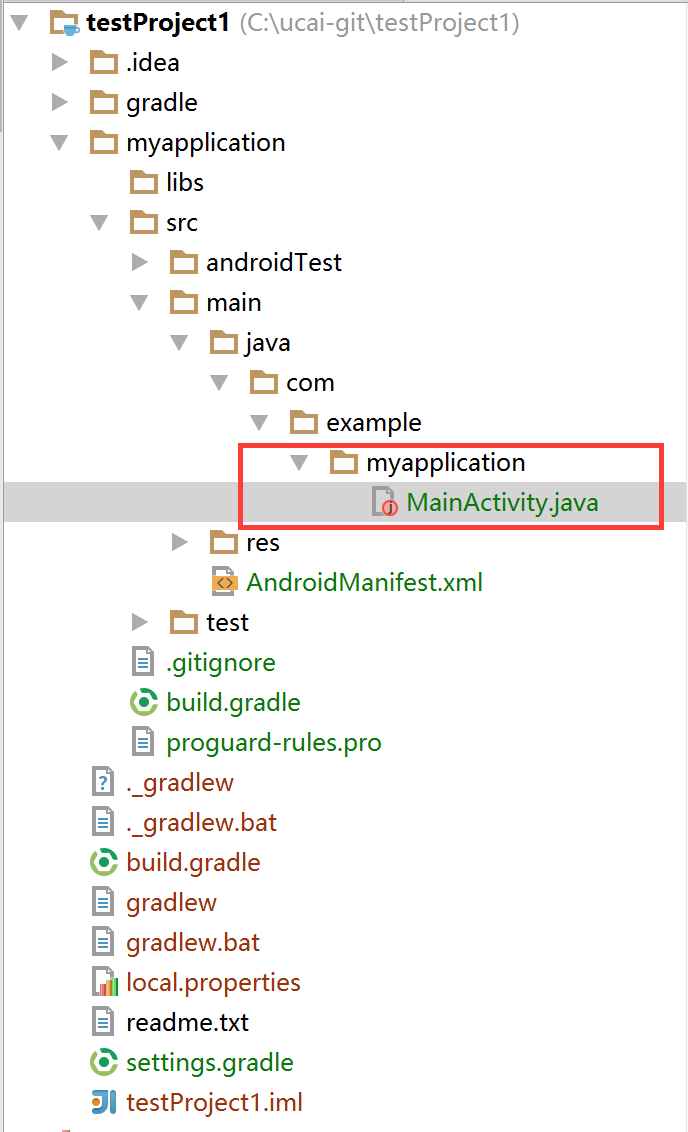
点击【Clone】开始检出项目，如图：



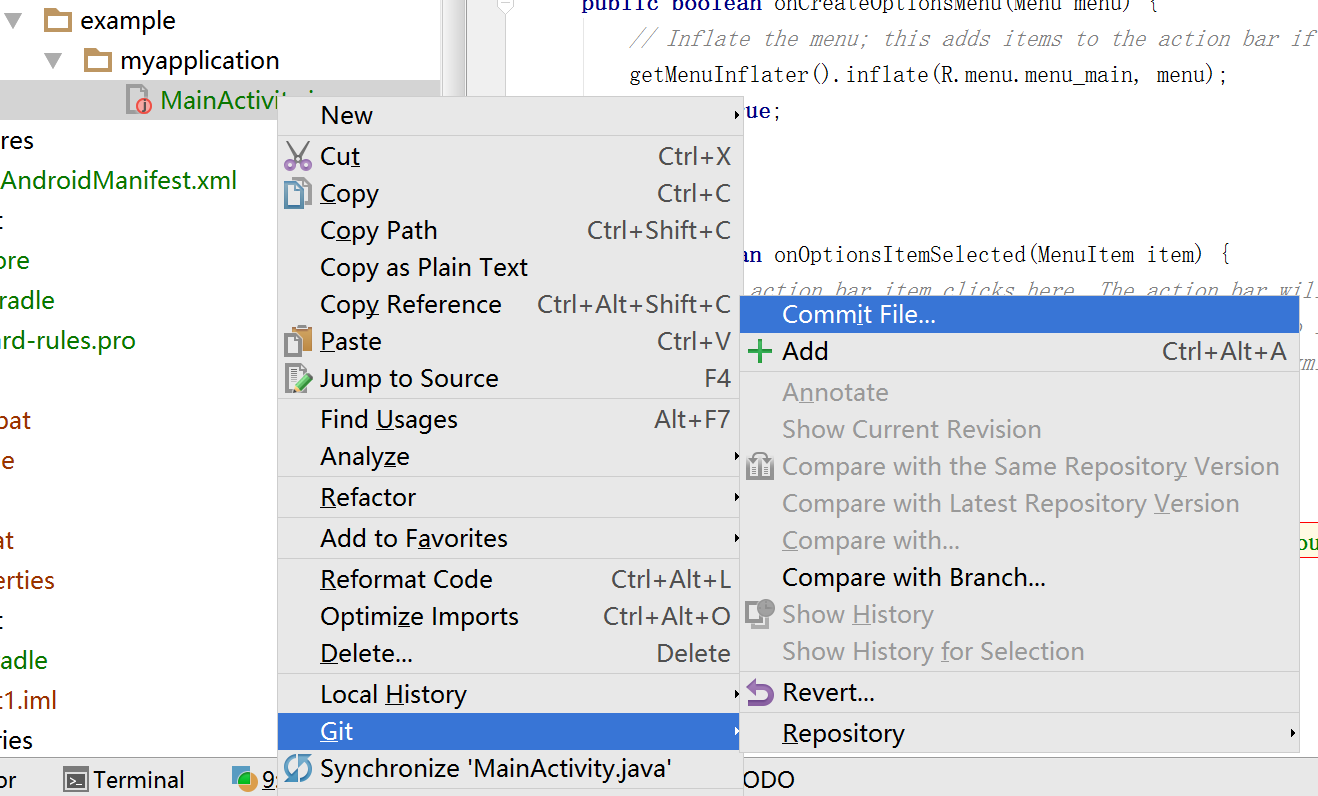
在项目中右键项目或者文件，可以执行add或者commit操作，如图：



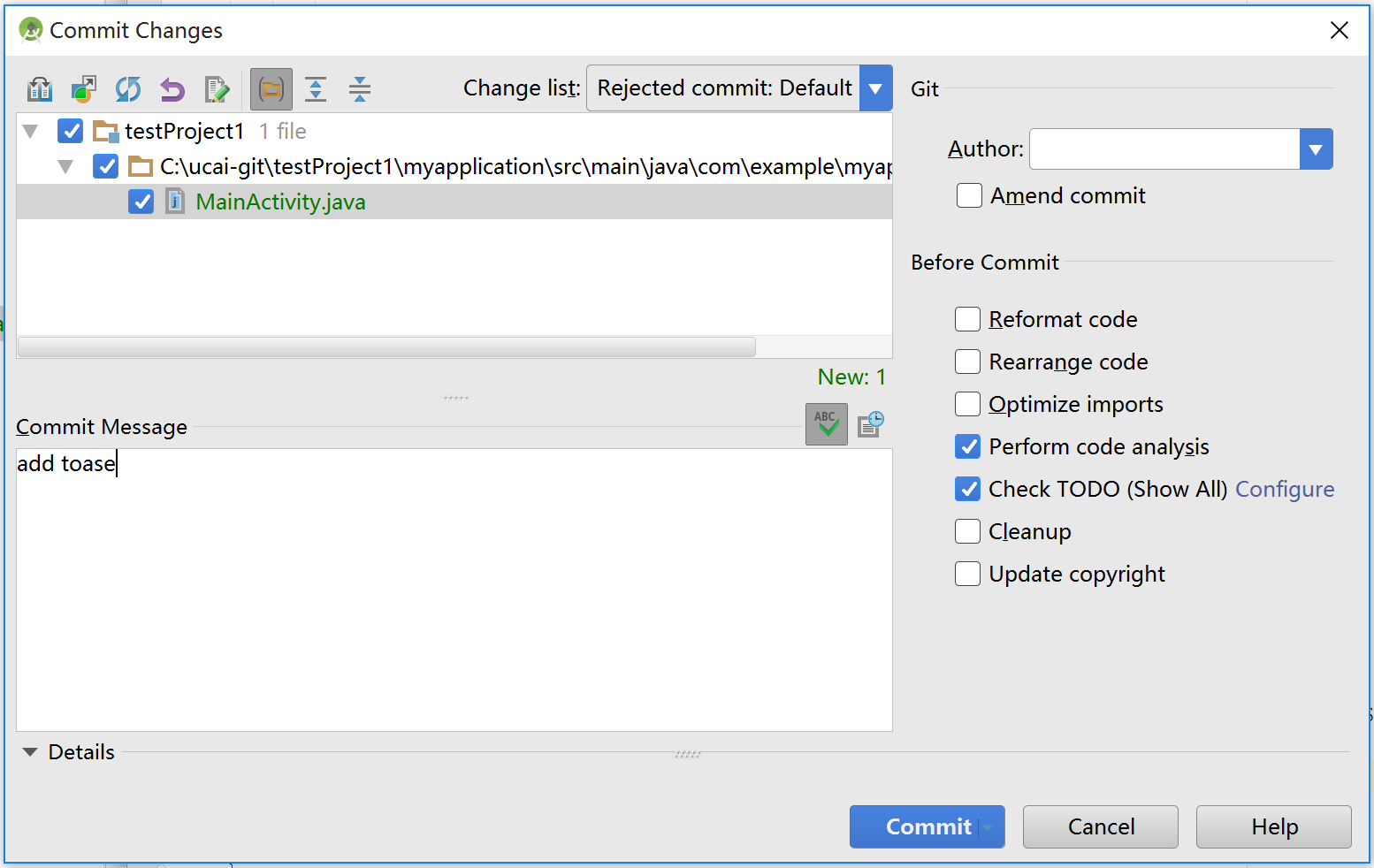
在项目列表，如果有修改文件，会有不同的图标标识出来，如图：



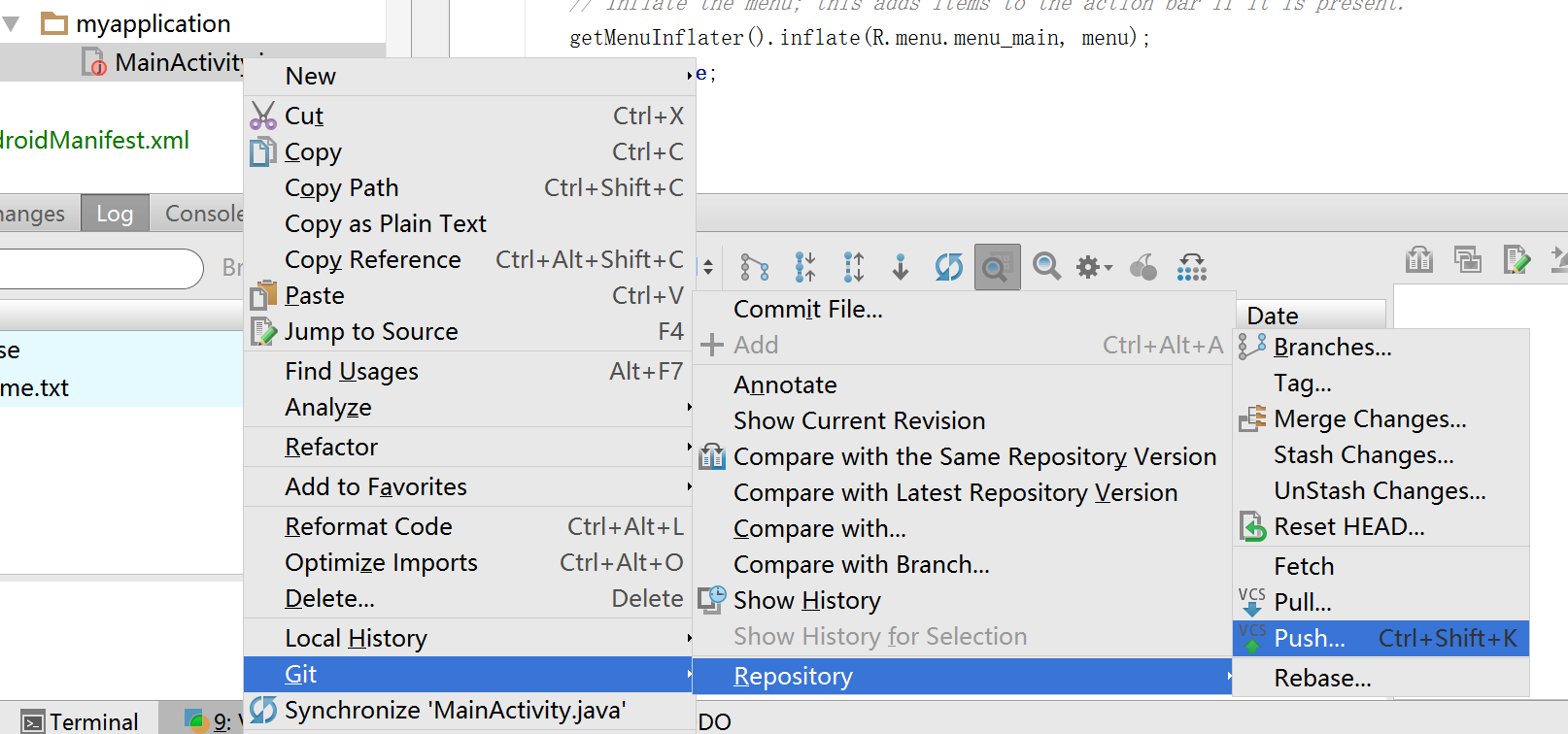
右键点击修改的文件，菜单项Git --> Commit File来执行提交操作，如图：



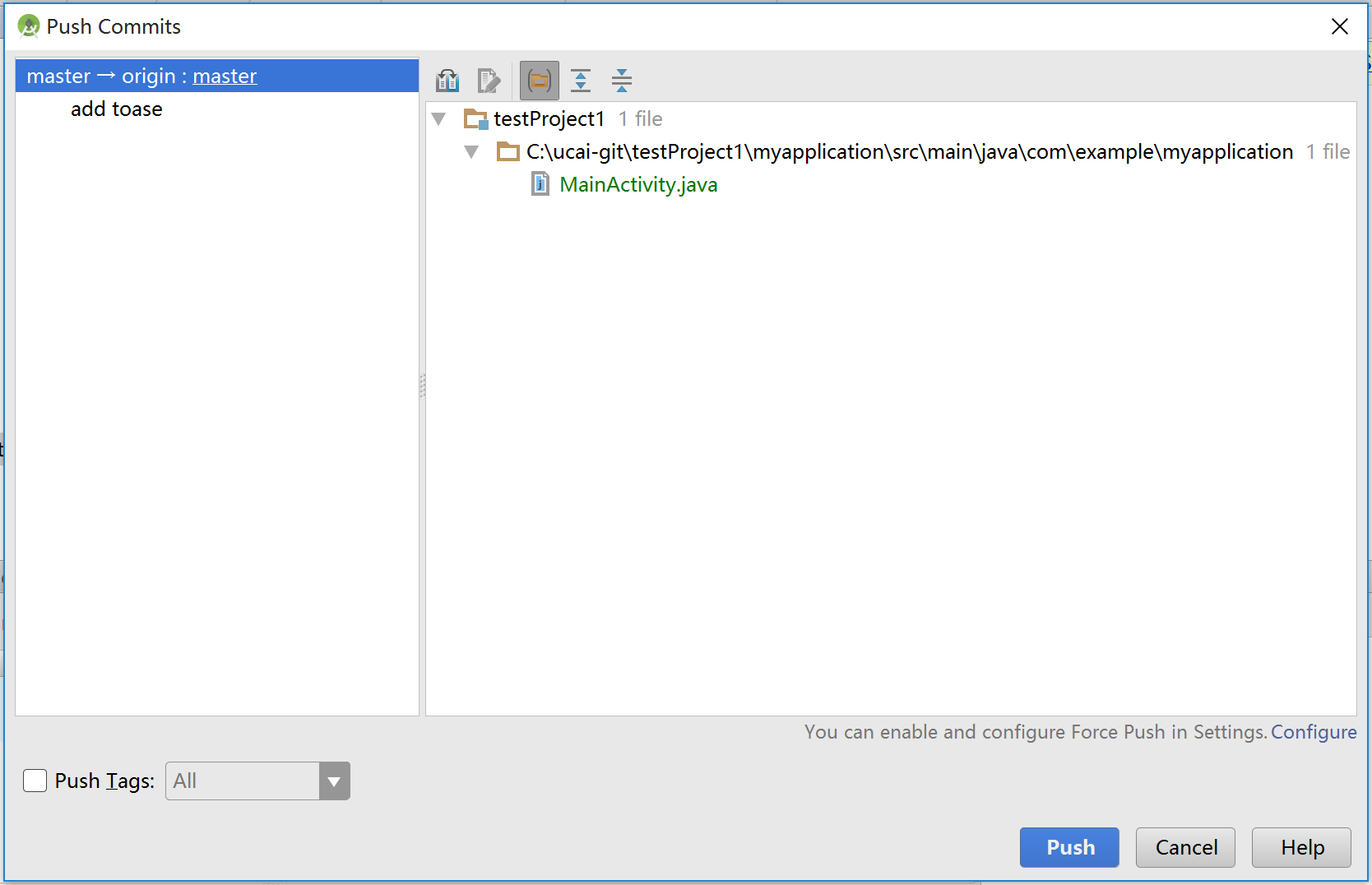
在提交界面，可以添加提交的备注说明，确认无误后，点击【Commit】提交，如图：



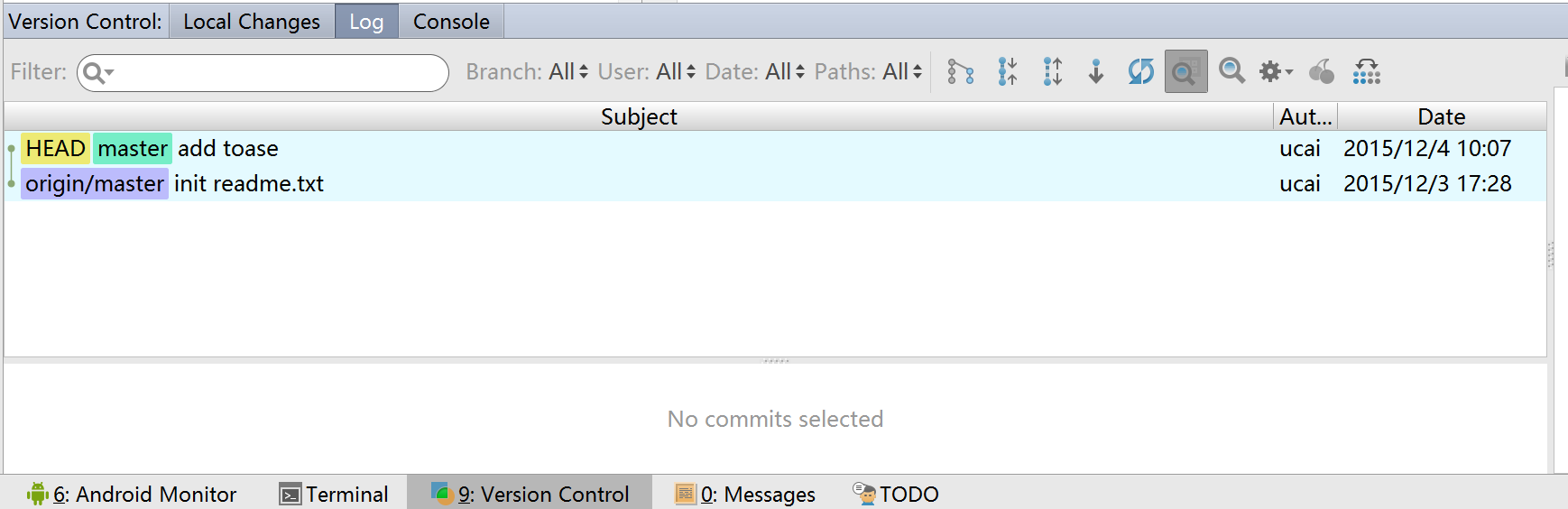
Commit完成后，再次点击修改的文件，执行push操作，如图：



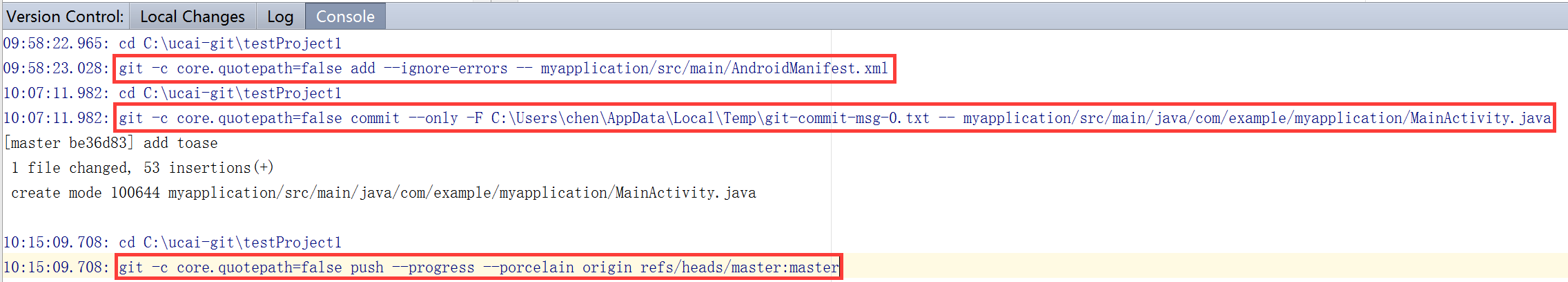
push界面会讲已经提交的修改信息显示，点击【Push】，如图：



push成功后，可以在Version Control中的Log栏查看分支信息，如图：



以上操作都是图形界面的交互，其实本质是Git命令行执行，可以在Version Control中的Console中查看，如图：



关键命令及参数如下：

git add myapplication/src/main/AndroidManifest.xml

git commit myapplication/src/main/… …/MainActivity.java

git push origin refs/heads/master:master

**Git基础命令**

我在版本库testgit目录下新建一个记事本文件 readme.txt 内容如下：11111111

第一步：使用命令 git add readme.txt添加到暂存区里面去。如图：



如果和上面一样，没有任何提示，说明已经添加成功了。

第二步：用命令 git commit告诉Git，把文件提交到仓库。如图：



现在我们已经提交了一个readme.txt文件了，我们下面可以通过命令git status来查看是否还有文件未提交，如图：



说明没有任何文件未提交，但是我现在继续来改下readme.txt内容，比如我在下面添加一行2222222222内容，继续使用git status来查看下结果，如图：



上面的命令告诉我们 readme.txt文件已被修改，但是未被提交的修改。

接下来我想看下readme.txt文件到底改了什么内容，如何查看呢？可以使用命令： git diff readme.txt， 如图：



如上可以看到，readme.txt文件内容从一行11111111改成二行添加了一行22222222内容。

知道了对readme.txt文件做了什么修改后，我们可以放心的提交到仓库了，提交修改和提交文件是一样的2步(第一步是git add，第二步是：git commit)。

如图：



Git版本回退：

     如上，我们已经学会了修改文件，现在我继续对readme.txt文件进行修改，再增加一行内容为33333333333333.继续执行命令如下：



现在我已经对readme.txt文件做了三次修改了，那么我现在想查看下历史记录，如何查呢？我们现在可以使用命令git log演示如图所示：



  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，最近的一次是增加内容为333333，上一次是添加内容222222，第一次默认是 111111，如果嫌上面显示的信息太多的话，我们可以使用命令git log

–pretty=oneline 演示如图：



  现在我想使用版本回退操作，我想把当前的版本回退到上一个版本，要使用什么命令呢？可以使用如下2种命令，第一种是：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。未回退之前的readme.txt内容如图：



如果想回退到上一个版本的命令如下操作：



再来查看下readme.txt内容，通过命令cat readme.txt查看，如图：



可以看到，内容已经回退到上一个版本了。我们可以继续使用git log来查看下历史记录信息，如图：



我们看到增加333333内容我们没有看到了，但是现在我想回退到最新的版本，如：有333333的内容要如何恢复呢？我们可以通过版本号回退，使用命令方法如下：git reset  –hard版本号，但是现在的问题假如我已经关掉过一次命令行或者333内容的版本号我并不知道呢？要如何知道增加3333内容的版本号呢？可以通过如下命令即可获取到版本号：git reflog演示如图：



通过上面的显示我们可以知道，增加内容3333的版本号是 6fcfc89。我们现在可以命令git reset  –hard 6fcfc89来恢复了。演示如图：



可以看到，目前已经是最新的版本了。

**理解工作区与暂存区的区别**

工作区：就是你在电脑上看到的目录，比如目录下testgit里的文件(.git隐藏目录版本库除外)。或者以后需要再新建的目录文件等等都属于工作区范畴。

版本库(Repository)：工作区有一个隐藏目录.git,这个不属于工作区，这是版本库。其中版本库里面存了很多东西，其中最重要的就是stage(暂存区)，还有Git为我们自动创建了第一个分支master,以及指向master的一个指针HEAD。

我们前面说过使用Git提交文件到版本库有两步：

  第一步：是使用git add把文件添加进去，实际上就是把文件添加到暂存区。

  第二步：使用git commit提交更改，实际上就是把暂存区的所有内容提交到当前分支上。

我们继续使用demo来演示下：

我们在readme.txt再添加一行内容为4444444，接着在目录下新建一个文件为test.txt 内容为test，我们先用命令git status来查看下状态，如下：



现在我们先使用git add命令把2个文件都添加到暂存区中，再使用git status来查看下状态，如下：



接着我们可以使用git commit一次性提交到分支上，如下：



Git撤销修改操作

比如我现在在readme.txt文件里面增加一行内容为555555555555，我们先通过命令cat readme.txt查看如图：



在我未提交之前，我发现添加5555555555555内容有误，所以我得马上恢复以前的版本，现在我可以有如下几种方法可以做修改：

第一：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。

第二：我可以按以前的方法直接恢复到上一个版本。使用 git reset  –hard HEAD^

但是现在我不想使用上面的2种方法，我想直接想使用撤销命令该如何操作呢？首先在做撤销之前，我们可以先用git status查看下当前的状态。如图所示：



可以发现，Git会告诉你，git checkout  -- file可以丢弃工作区的修改，如下命令：

git checkout  --  readme.txt,如图所示：



命令 git checkout --readme.txt 意思就是，把readme.txt文件在工作区做的修改全部撤销，这里有2种情况，如下：

1. readme.txt自动修改后，还没有放到暂存区，使用撤销修改就回到和版本库一模一样的状态。
2. 另外一种是readme.txt已经放入暂存区了，接着又作了修改，撤销修改就回到添加暂存区后的状态。

对于第二种情况，我想我们继续做demo来看下，假如现在我对readme.txt添加一行 内容为6666666666666，我git add增加到暂存区后，接着添加内容7777777，我想通过撤销命令让其回到暂存区后的状态。如图所示：



注意：命令git checkout -- readme.txt 中的--很重要，如果没有--的话，那么命令变成创建分支了。

Git删除文件操作

现在版本库testgit目录添加一个文件b.txt，然后提交。如图：



如上：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，现在目录是这样的，如图：



只要没有commit之前，如果我想在版本库中恢复此文件如何操作呢？

可以使用如下命令git checkout  -- b.txt，如图所示：



再来看看我们testgit目录，添加了3个文件了。如图所示：

