

Android Studio 上如何使用 Git

1. 下载 Git Gui 工具 (选择对应电脑系统的版本, 实验电脑为 windows 系统)

地址: <http://git-scm.com/downloads>



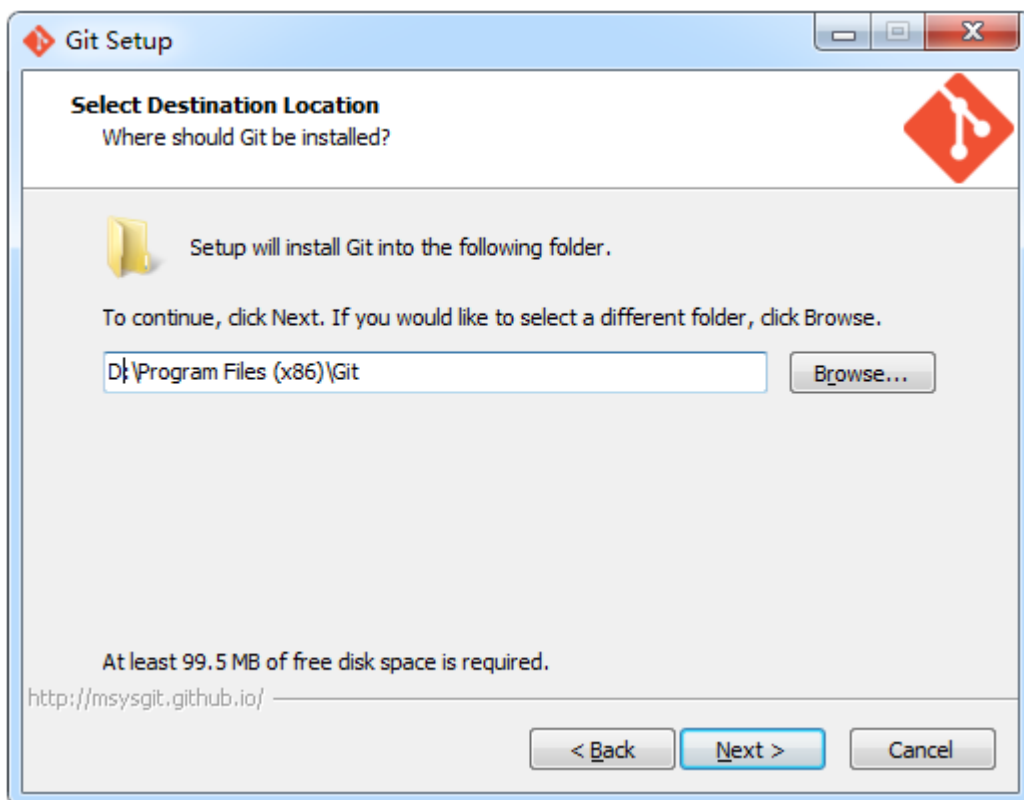
2. 安装 Git



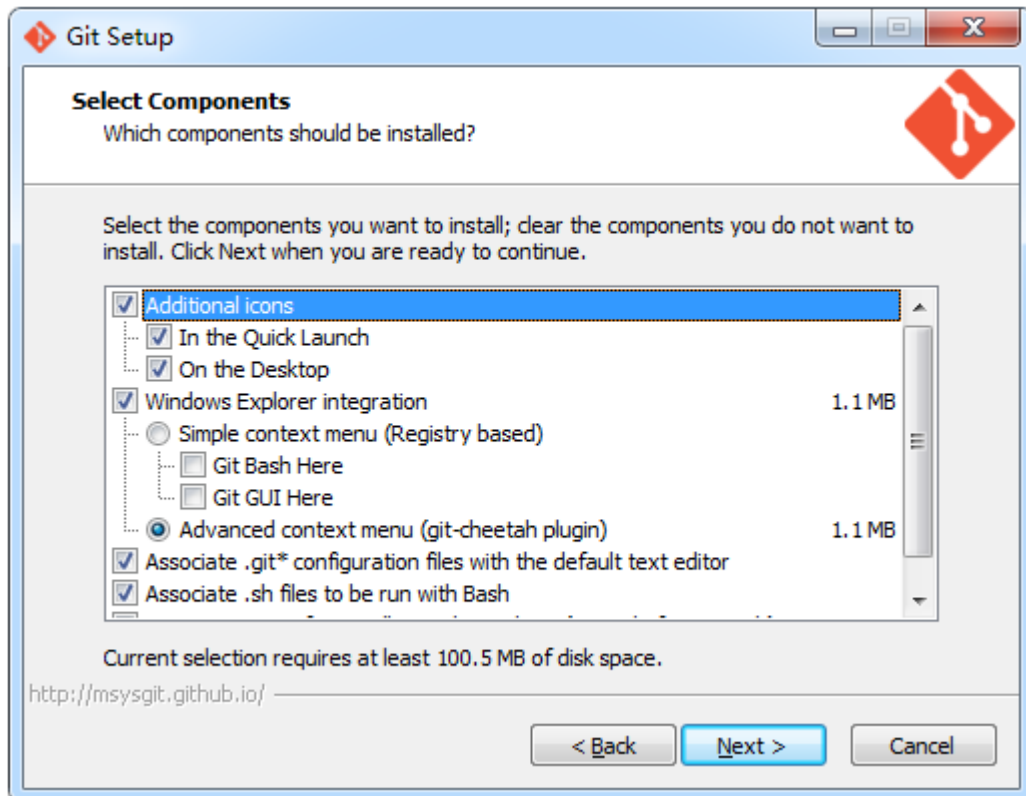
点击 Next,



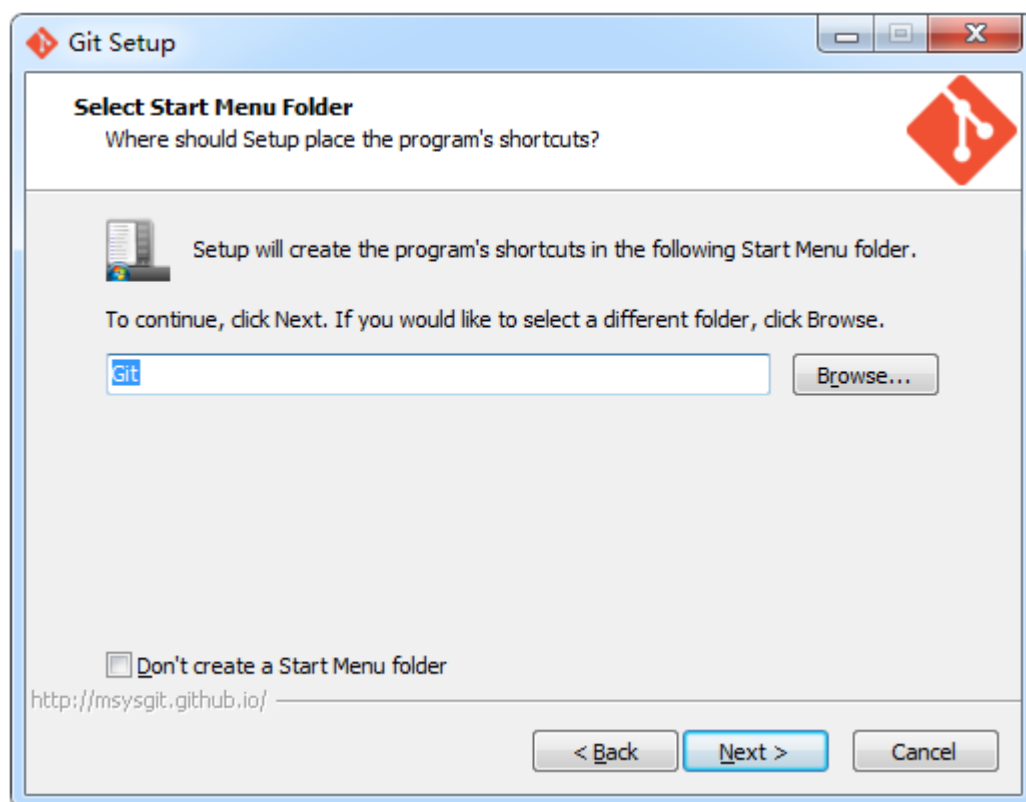
点击 Next, 选择安装地址:



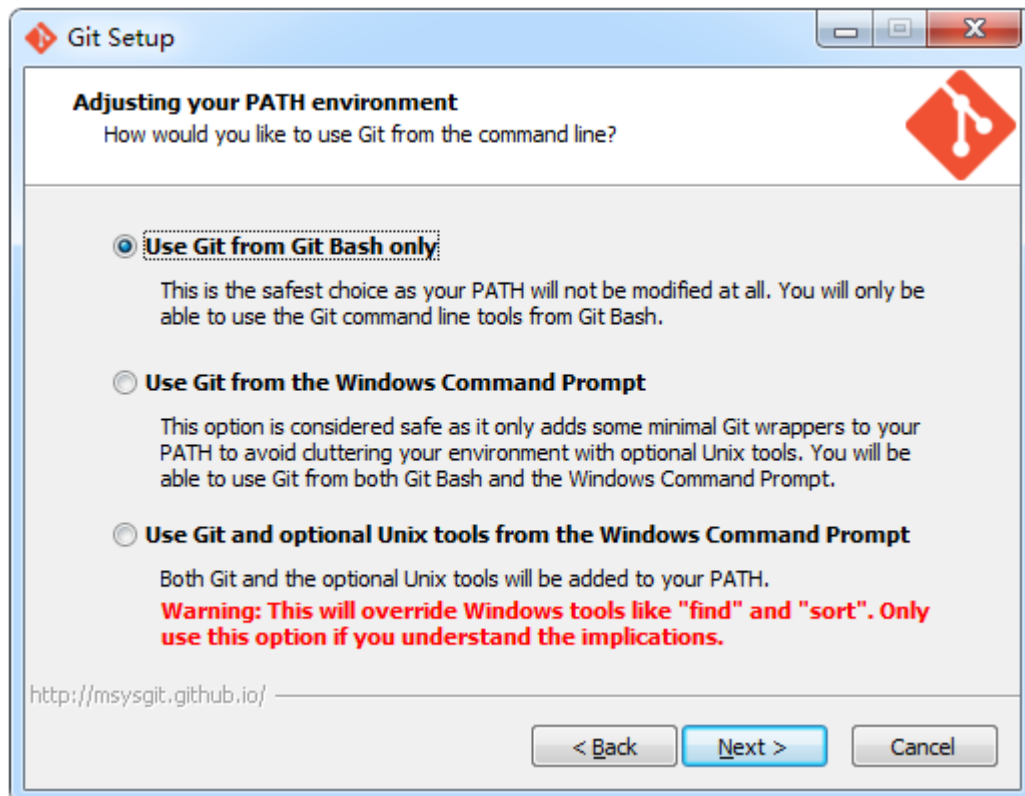
点击 Next



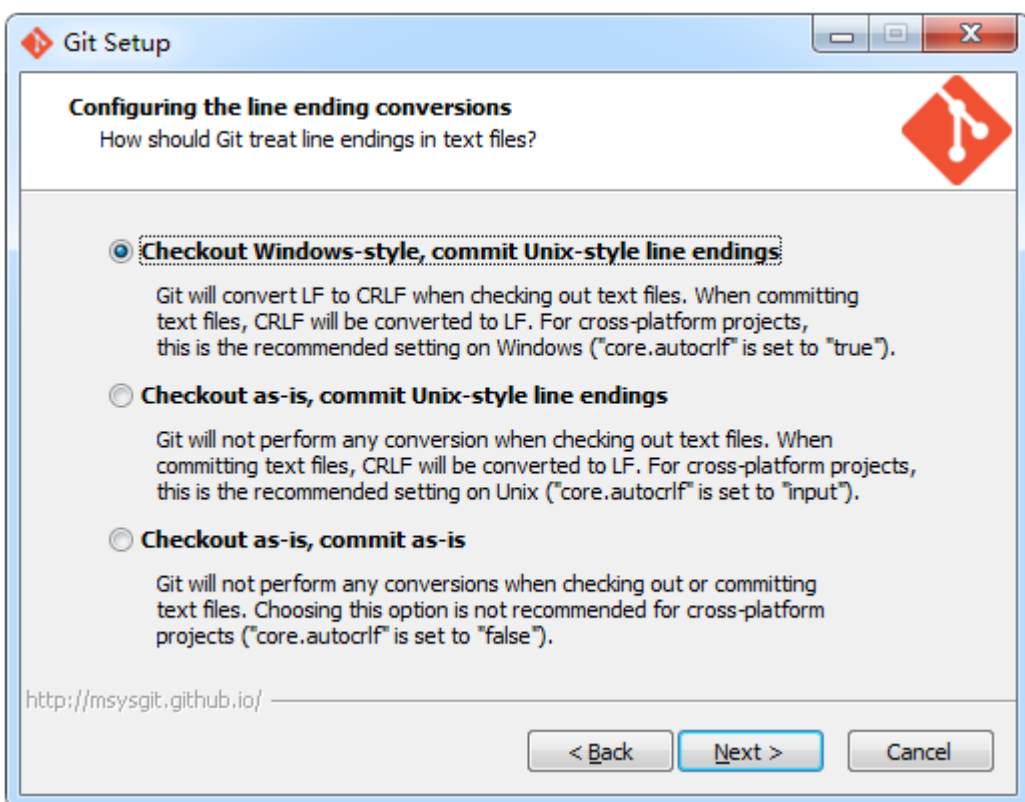
点击 Next



点击 Next



点击 Next



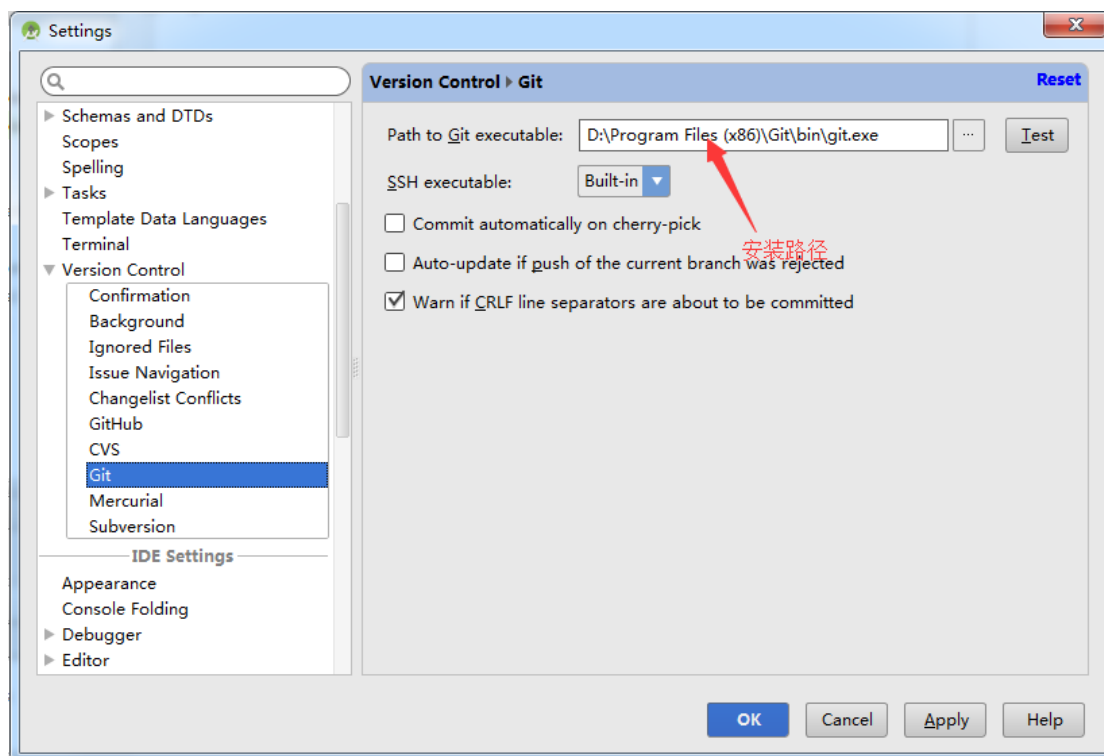
点击 Next



点击 Finish 完成安装。

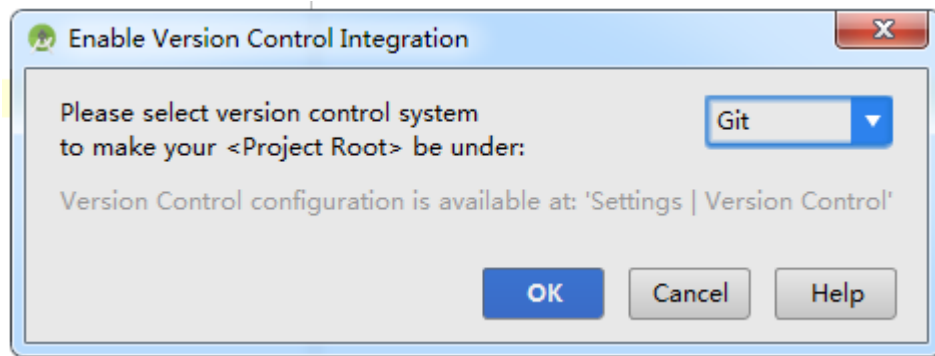
3. 在 Android Studio 上设置 Git 路径

File > Settings > Version Control > Git

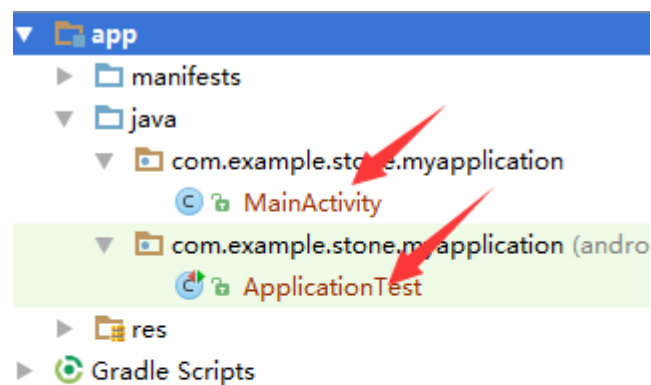


4 开启 git

VCS > Enable Version Control Integration > select "Git"

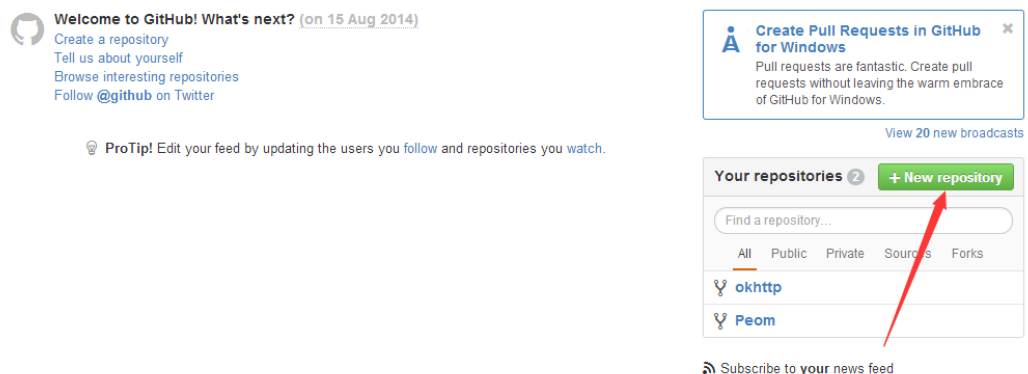


开启 git 之后，文件变成红色。



5. 在 github 上注册账号，创建仓库

- 1) <https://github.com/> 在 github 上注册账号
- 2) 注册登录后，创建仓库



Github 的私人项目是要收费的，创建时请注意类型选择

Owner

Repository name



stone1991

 /

hello_world



Great repository names are short and memorable. Need inspiration? How about **fuzzy-octo-sansa**.

Description (optional)

☒  **Public**

Anyone can see this repository. You choose who can commit.

☐  **Private**

You choose who can see and commit to this repository.

☒ **Initialize this repository with a README**

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None**

Add a license: **None** 

Create repository

 This repository Search

Explore Gist Blog Help

stone1991 + -   

stone1991 / hello_world

 Unwatch 1  Star 0

Quick setup — if you've done this kind of thing before

 Set up in Desktop

 or

HTTPS

SSH

https://github.com/stone1991/hello_world.git



We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo # hello_world >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/stone1991/hello_world.git
git push -u origin master
```



...or push an existing repository from the command line

```
git remote add origin https://github.com/stone1991/hello_world.git
git push -u origin master
```



Code

 Issues 0

 Pull requests 0

 Wiki

 Pulse

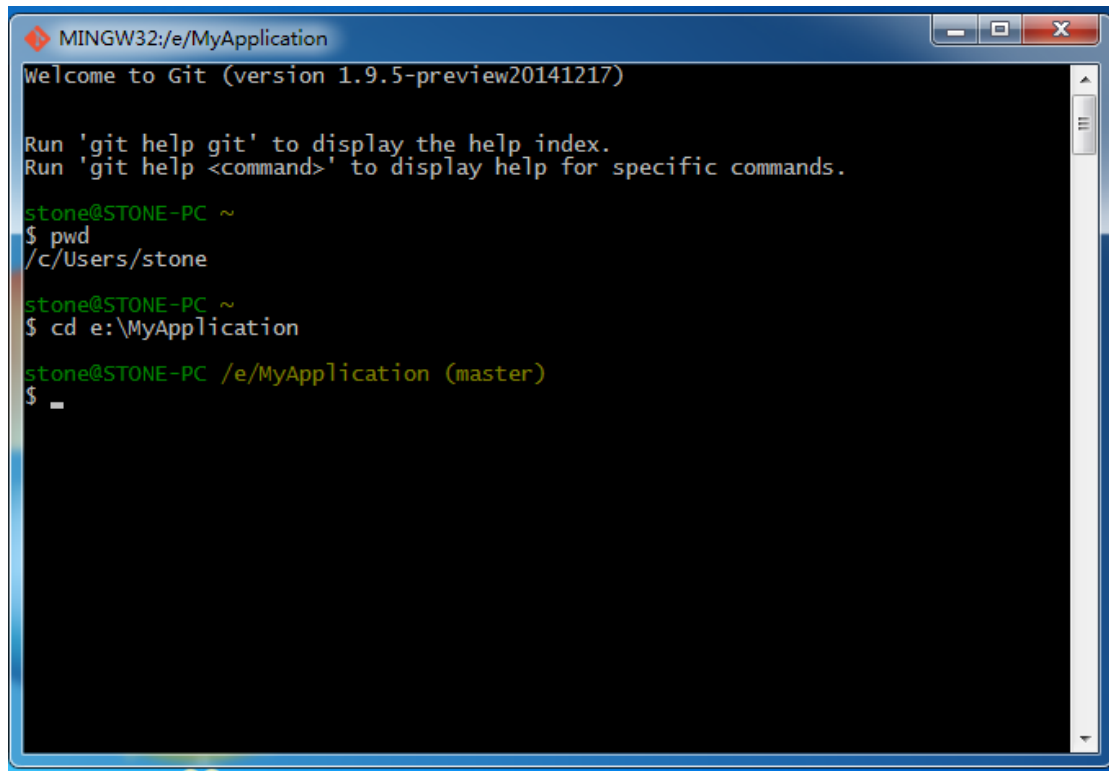
 Graphs

 Settings

远程仓库地址

6. 添加远程库

- 1) 打开下载好的 Git Bash，将目录切换到项目的目录(实验中项目目录：
E:\MyApplication)



```
MINGW32:/e/MyApplication
Welcome to Git (version 1.9.5-preview20141217)

Run 'git help git' to display the help index.
Run 'git help <command>' to display help for specific commands.

stone@STONE-PC ~
$ pwd
/c/Users/stone

stone@STONE-PC ~
$ cd e:\MyApplication

stone@STONE-PC /e/MyApplication (master)
$ _
```

然后输入 git 添加 remote 的命令:

git remote add origin https://github.com/stone1991/hello_world.git

```
stone@STONE-PC /e/MyApplication (master)
$ git remote add origin https://github.com/stone1991/helloworld.git
fatal: remote origin already exists.

stone@STONE-PC /e/MyApplication (master)
$ git remote rm origin

stone@STONE-PC /e/MyApplication (master)
$ git remote add origin https://github.com/stone1991/hello_world.git

stone@STONE-PC /e/MyApplication (master)
$
```

Ps: https://github.com/stone1991/hello_world.git 是 github 创建仓库时生成的地址

-----关于可能出现的错误-----

执行

```
$ git remote add origin git@github.com:defnngj/hello_world.git
```

错误提示: fatal: remote origin already exists.

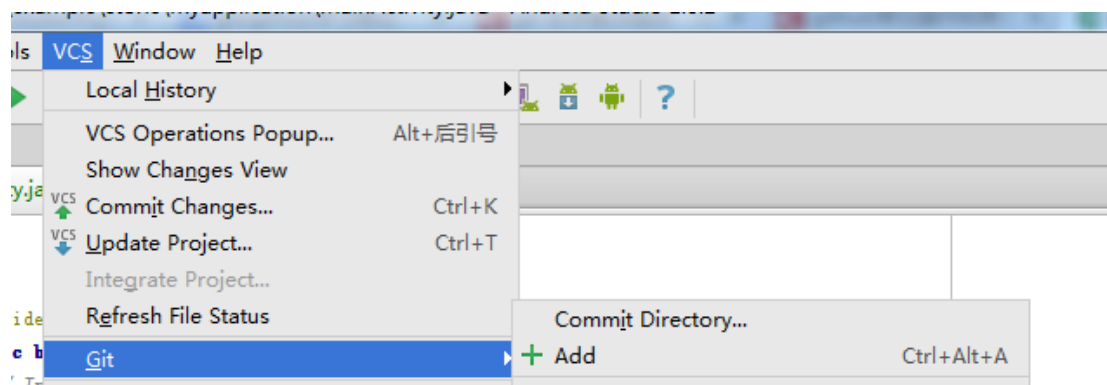
解决办法:

```
$ git remote rm origin
```

然后在执行: \$ git remote add origin git@github.com:defnngj/hello_world.git 就不会报错误了。

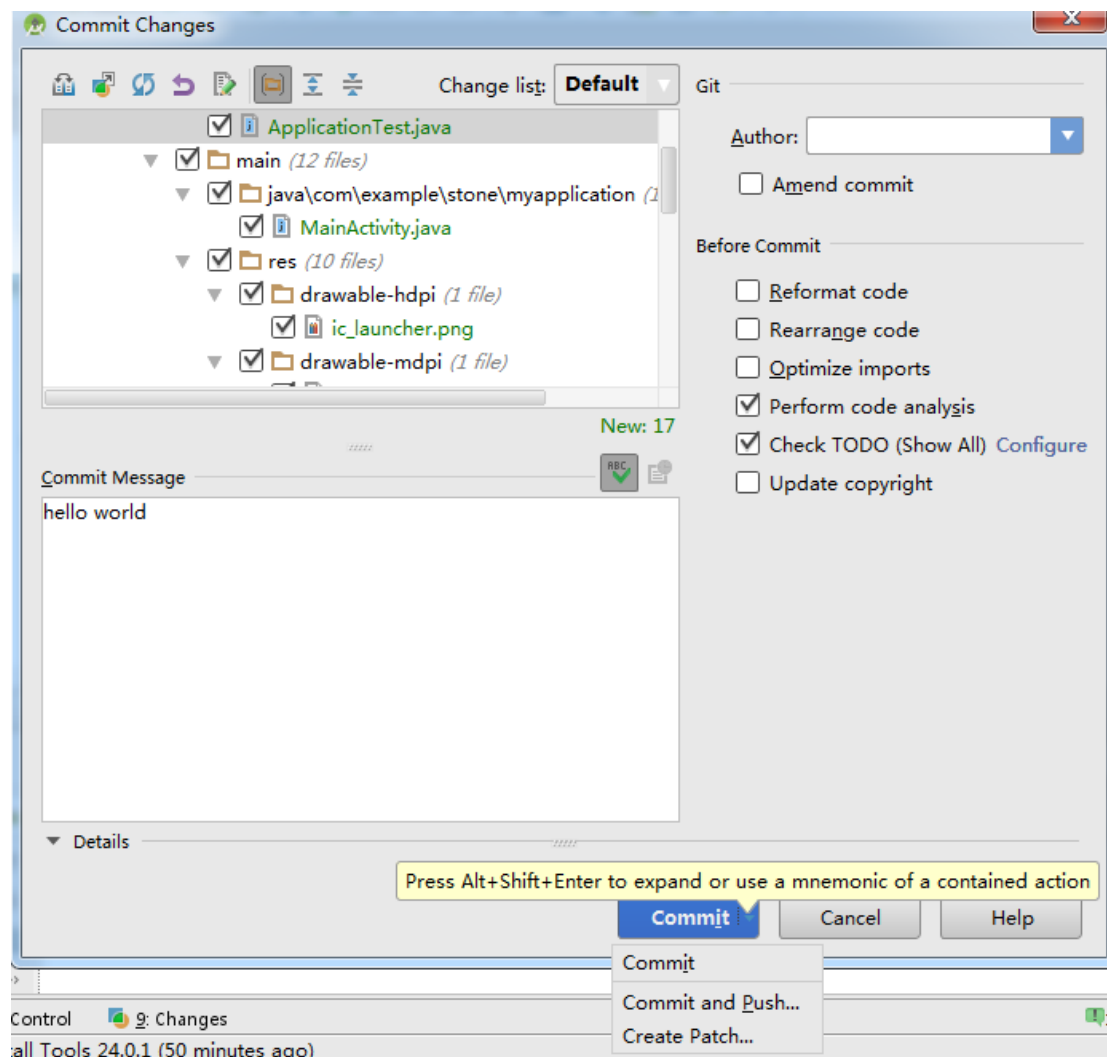
7. 添加项目到 git 版本控制

VCS > Git > Add

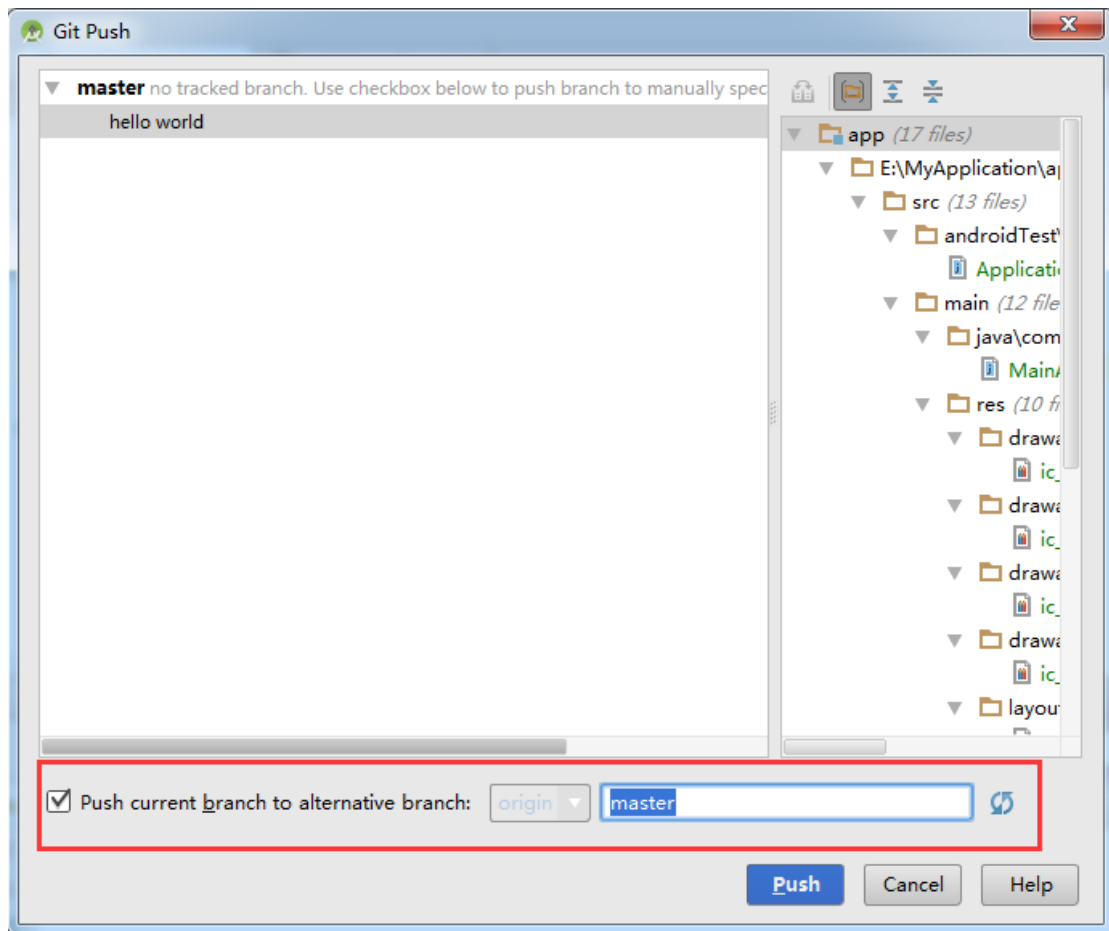
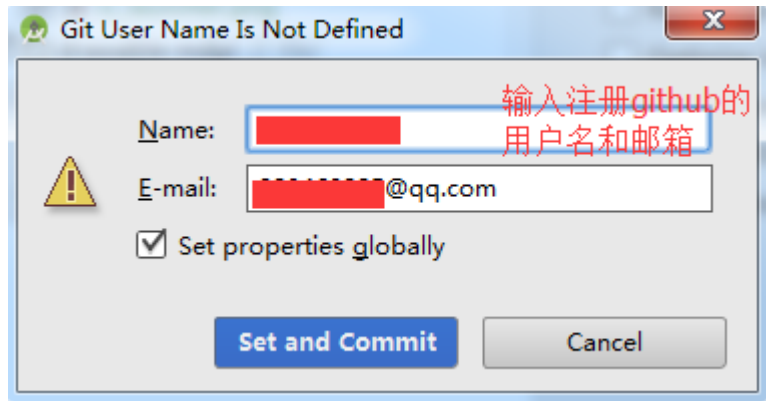


8. commit 到 git 服务器

VCS > Commit Changes...



若选择 commit and push 将提交到服务器上，也可以通过 VCS > Git > Push...提交到服务器。



```
Version Control Console
fatal: Authentication failed for 'https://github.com/stone1991/hello_world.git/'
23:08:45.665: cd E:\MyApplication
23:08:45.666: git push --progress origin master:master
remote: Invalid username or password.
fatal: Authentication failed for 'https://github.com/stone1991/hello_world.git/'
23:10:28.145: cd E:\MyApplication
23:10:28.145: git log --max-count=5 master --pretty=format:%x01%H%x02ct%x02%an%x02%at%x02%ae%x02%
23:10:31.702: cd E:\MyApplication
23:10:31.703: git log --max-count=5 master --pretty=format:%x01%H%x02ct%x02%an%x02%at%x02%ae%x02%
23:10:31.802: cd E:\MyApplication
23:10:31.803: git push --progress origin master:master
Counting objects: 42, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (22/22), done.
Writing objects: 100% (42/42), 52.90 KiB | 0 bytes/s, done.
Total 42 (delta 0), reused 0 (delta 0)
To https://github.com/stone1991/hello_world.git
 * [new branch]      master -> master

Push successful
master pushed to new branch origin/master
```

stone1991 / hello_world Unwatch 1

Description

Website

Short description of this repository

Website for this repository (optional)

Save or Cancel

1 commit

1 branch

0 releases

1 contributor

branch: master

hello_world / +

hello world

stone1991 authored an hour ago latest commit 261efa92cb

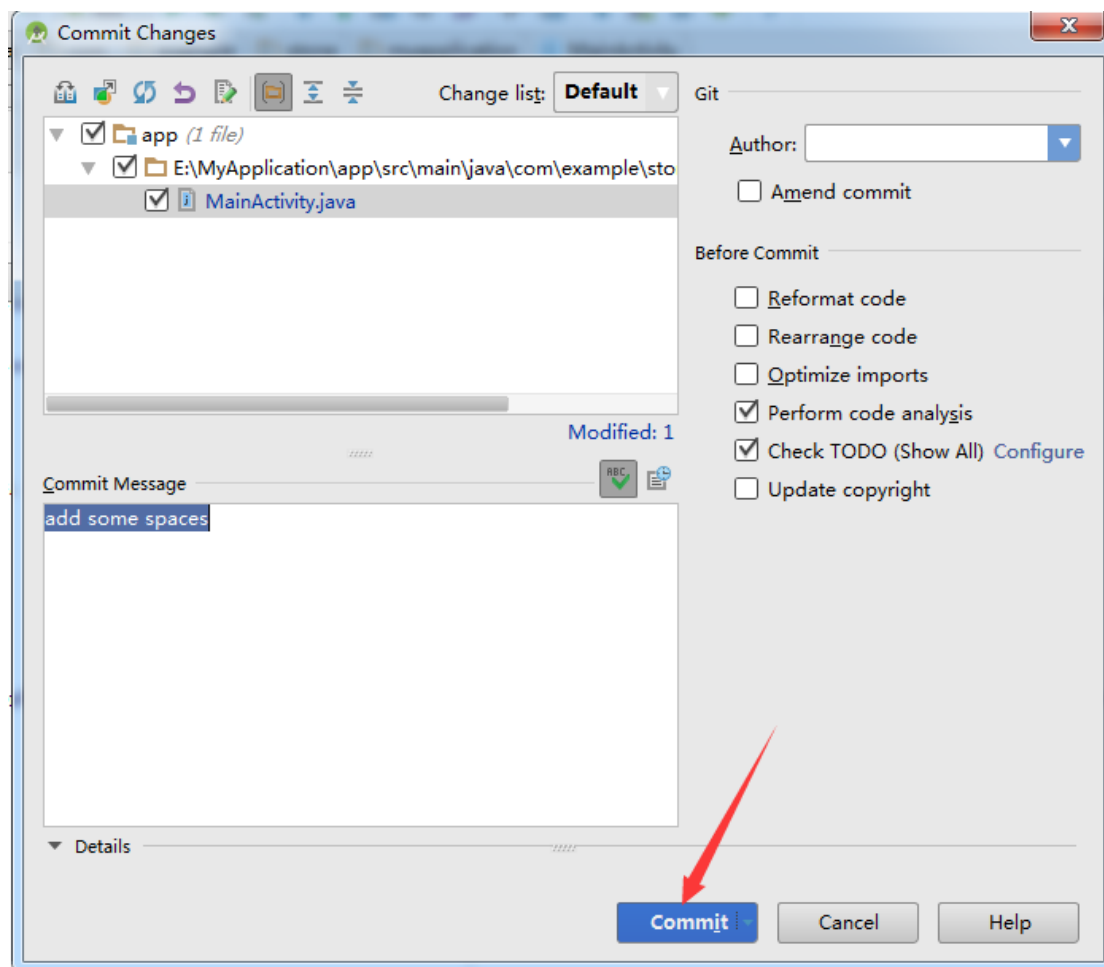
app hello world an hour ago

Help people interested in this repository understand your project by adding a README!

Add a README

9. commit 到 git 本地

修改项目文件小细节后，可以提交到本地



提交到本地

```
[total 42 (delta 0), reused 0 (delta 0)]
To https://github.com/stone1991/hello_world.git
 * [new branch] master -> master
23:16:24.351: cd E:\MyApplication
23:16:24.351: git config core.autocrlf
23:16:24.496: cd E:\MyApplication
23:16:24.496: git commit --only -- C:\Users\stone\AppData\Local\Temp\git-commit-msg-5307711310713356005.txt -- app/src/main/java/com/example/stone/myapplication/MainActivity.java
[master 5671a38] add some spaces
1 file changed, 2 insertions(+)
```

10. 从服务器上获取最新版本

git pull 命令相当于是从远程获取最新版本并 merge 到本地

git fetch 命令相当于是从远程获取最新版本到本地，不会自动 merge

在实际使用中，git fetch 更安全一些，因为在 merge 前，我们可以查看更新情况，然后再决定是否合并。

11. 回滚已经 commit 的代码

git reset --hard <commit ID 号>

12. 创建分支

一般在增加新功能，防止出问题影响到主分支，而且分支还可以保留开发中可能出现的各种 bug 方便修复但不影响主分支的使用。

执行 git branch <branchname> 命令创建新分支

切换分支: 执行 `git checkout <branchname>` 命令切换到新分支

13. 合并分支

如果要将开发中的分支 (develop), 合并到稳定分支 (master)。

首先切换的 master 分支: `git checkout master`。

然后执行合并操作: `git merge develop`。