

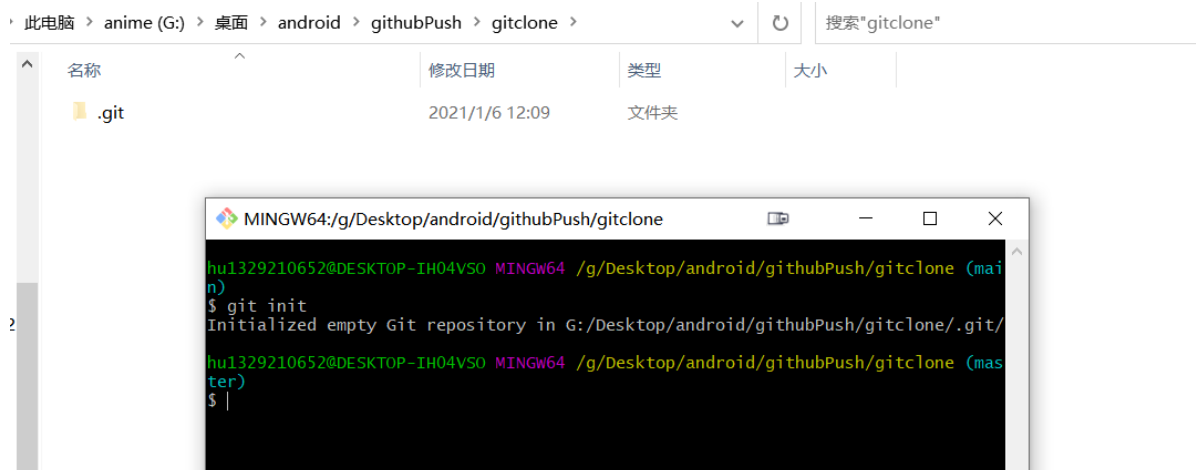
github上传说明

在本次实验中我们使用github的组来进行代码的合并和迭代

首先是，我已经邀请了所有人进入了我的github组，组的名称为Face Android2021.

首先是把项目克隆下来，新建一个文件夹，

```
git init
```



```
git clone https://github.com/ArcFaceAndroid2021/ArcSoft_ArcFace_Android.git
```



如果clone的时候速度太慢，因为大家都是有梯子的，所以这里介绍一下，git bash的代理设置方法

ss 或者 ssr

```
1 git config --global http.proxy "http://127.0.0.1:1080"
2 git config --global https.proxy "https://127.0.0.1:1080"
```

clash

```
1 git config --global http.proxy "http://127.0.0.1:7890"
2 git config --global https.proxy "https://127.0.0.1:7890"
```

```

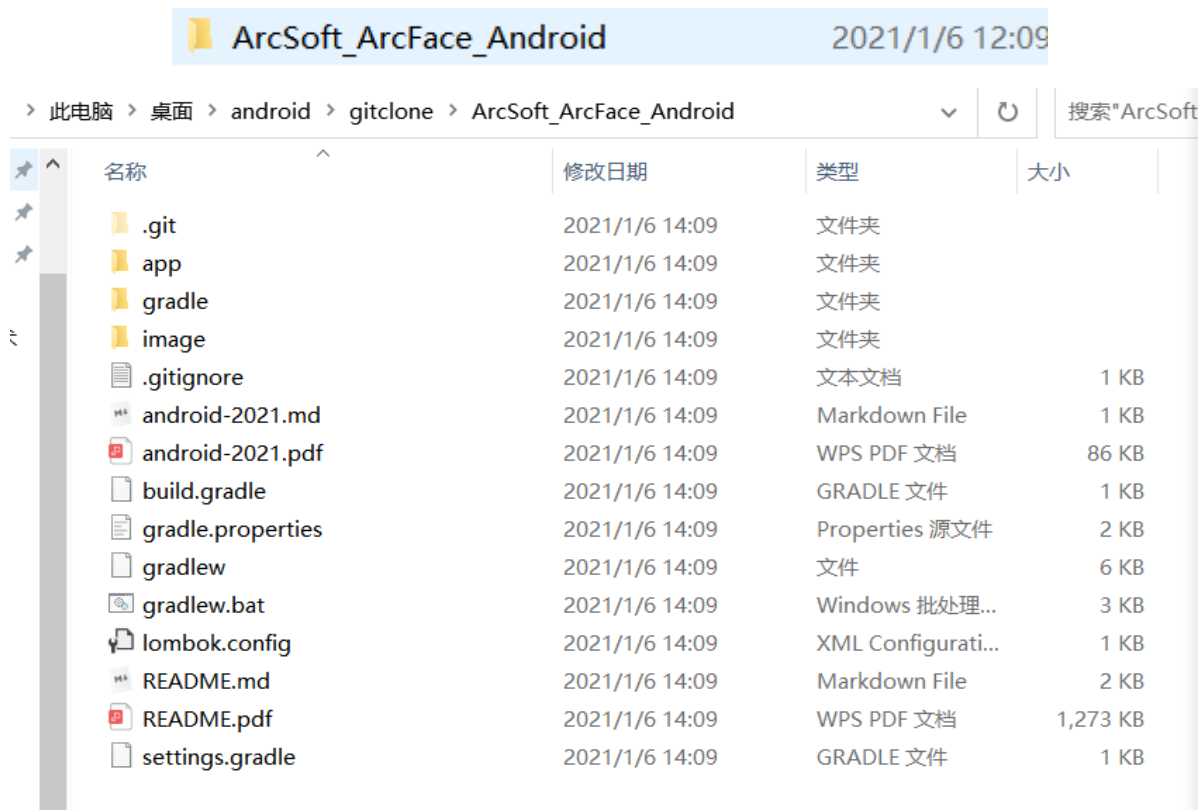
hu1329210652@DESKTOP-IH04VSO MINGW64 /g/Desktop/android/gitclone (master)
$ git config --global http.proxy "http://127.0.0.1:7890"

hu1329210652@DESKTOP-IH04VSO MINGW64 /g/Desktop/android/gitclone (master)
$ git config --global https.proxy "http://127.0.0.1:7890"

hu1329210652@DESKTOP-IH04VSO MINGW64 /g/Desktop/android/gitclone (master)
$ git clone https://github.com/ArcFaceAndroid2021/ArcSoft_ArcFace_Android.git
Cloning into 'ArcSoft_ArcFace_Android'...
remote: Enumerating objects: 194, done.
remote: Counting objects: 100% (194/194), done.
remote: Compressing objects: 100% (157/157), done.
Receiving objects: 46% (91/194), 20.78 MiB | 2.29 MiB/s

```

进入clone后的文件夹



在此路径打开 `bash`

如果你做了更改

请先 `git add` 和 `git commit -m ""`

```

MINGW64:/g/Desktop/android/gitclone/ArcSoft_ArcFace_Android
hu1329210652@DESKTOP-IH04VSO MINGW64 /g/Desktop/android/gitclone/ArcSoft_ArcFace
_Android (main)
$ git add .

hu1329210652@DESKTOP-IH04VSO MINGW64 /g/Desktop/android/gitclone/ArcSoft_ArcFace
_Android (main)
$ git commit -m "second commit"
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

```

使用 `git branch + your name of github` 创建自己的分支

```

hu1329210652@DESKTOP-IH04VSO MINGW64 /g/Desktop/android/gitclone/ArcSoft_ArcFace
_Android (main)
$ git branch sssuibian123

```

使用 `git branch` 查看目前已有的分支, 从该图中可以看出, 一共有两个分支, 目前在分支 `main` 上, 也就是我们目前所作的更改还是在 `main` 上

```
hu1329210652@DESKTOP-IH04VSO MINGW64 /g/Desktop/android/gitclone/ArcSoft_ArcFace_Android (main)
$ git branch
* main
  sssuibian123
```

使用 `git checkout + your name of github` 切换分支, 此时所作的修改在你自己的分支上

```
hu1329210652@DESKTOP-IH04VSO MINGW64 /g/Desktop/android/gitclone/ArcSoft_ArcFace_Android (main)
$ git checkout sssuibian123
Switched to branch 'sssuibian123'
Your branch is up to date with 'sssuibian123/sssuibian123'.
```

```
hu1329210652@DESKTOP-IH04VSO MINGW64 /g/Desktop/android/gitclone/ArcSoft_ArcFace_Android (sssuibian123)
$ git branch
  main
* sssuibian123
```

通过 `git remote` 设置远程链接 `git remote add +you name of github+`

`https://github.com/ArcFaceAndroid2021/ArcSoft_ArcFace_Android.git`

```
hu1329210652@DESKTOP-IH04VSO MINGW64 /g/Desktop/android/gitclone/ArcSoft_ArcFace_Android (main)
$ git remote add sssuibian123 https://github.com/ArcFaceAndroid2021/ArcSoft_ArcFace_Android.git
```

`git remote add` 后面的字符可以自由设置, 但是为了便于记忆建议改成自己的github名字, 这个就是给后面的url起个名字, 可以通过 `git remote` 查看目前已经存在的所有远程连接

```
hu1329210652@DESKTOP-IH04VSO MINGW64 /g/Desktop/android/gitclone/ArcSoft_ArcFace_Android (main)
$ git remote
origin
sssuibian123
```

`git push -u +远程连接名字+分支名字` 把某个分支上传到某个远程连接

```
hu1329210652@DESKTOP-IH04VSO MINGW64 /g/Desktop/android/gitclone/ArcSoft_ArcFace_Android (main)
$ git push -u sssuibian123 sssuibian123
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'sssuibian123' on GitHub by visiting:
remote:   https://github.com/ArcFaceAndroid2021/ArcSoft_ArcFace_Android/pull/new/sssuibian123
remote:
To https://github.com/ArcFaceAndroid2021/ArcSoft_ArcFace_Android.git
 * [new branch]      sssuibian123 -> sssuibian123
Branch 'sssuibian123' set up to track remote branch 'sssuibian123' from 'sssuibian123'.
```

最后大概相应的github页面查看效果

https://github.com/ArcFaceAndroid2021/ArcSoft_ArcFace_Android/tree/main

main ▾

3 branches

0 tags

Switch branches/tags

Find or create a branch...

Branches

Tags

✓ main

default

AowerDmax

ssssuibian123

[View all branches](#)

first commit

first commit

first commit

first commit

first commit

1、查找历史版本

使用 `git log` 命令查看所有的历史版本，获取你git的某个历史版本的id

假设查到历史版本的id是 `fae6966548e3ae76cfa7f38a461c438cf75ba965`。

2、恢复到历史版本

```
1 $ git reset --hard fae6966548e3ae76cfa7f38a461c438cf75ba965
```

3、把修改推到远程服务器

```
1 $ git push -f -u origin master
```