Git总结

git分支：

https://www.jianshu.com/p/b357df6794e3

下载代码

```
git clone http://172.16.1.216/uzaiStudio/uzaiPro.git
```

- master 分支

  Git中默认的主分支，master分支的Head节点所指向的版本始终是可以用于生产环境的正式版本；

- develop 分支

  Git中另一个主分支，其Head节点总是指向下一个待发布版本的最新变化，当develoip分支达到某一稳定点可进行新版发布时，合并到master分支并打上tag标签。

- 辅助分支

1. Feature Branch

   需求分支用于为未来的软件版本开发新的功能需求,只要该需求尚在开发中，该需求分支就会一直存在,需求分支最终会被合并到develop分支中作为下一个待发布版本的功能之一，或者由于该需求无法实现从而被抛弃。

注：需求分支通常仅仅存在于开发者的代码仓库中（本地仓库），并不上传到远程分支。

如何创建需求分支（创建需求分支时，该分支必须从develop分支得到）

从develop分支创建feature_branch 分支

```
git checkout -b feature_branch develop
```

从当前分支切换到feature_branch分支

```
git checkout feature_branch
```

将已完成的需求分支合并到develop分支

切换到develop分支

```
git checkout develop
```

合并分支

```
git merge --no-ff feature_branch
```

删除需求分支

```
git branch -d feature_branch
```

推送

```
git push origin develop
```

2. Release Branch

   发布分支用于辅助新版本（生产环境）发布的准备工作，例如Bug的修复、版本号的修改。

创建分布分支

创建并切换到release-1.2分支

```
git checkout -b release-1.2 develop
```

表示对版本号的修改，或者小bug的修复等

```
vim file
```

提交代码

```
git commit -a -m
```

注： 如果在发布分支进行小型bug的修改，则需要将提交后的代码合并到develop分支中

完成发布分支

合并到develop的操作

```
git checkout develop
git merge --no-ff release-1.2
```

合并到master的操作

```
git checkout master
git merge --no-ff release-1.2
git tag -a "v1.2"
```

删除发布分支

```
git branch -d release-1.2
```

3. Hotfix Branch

   修复分支用于正式版本的紧急修复，在紧急修复完成后 **必须同时合并到master和develop分支**

创建修复分支

```
git checkout -b hotfix-1.2.1 master
vim file #表示修复bug
git commit -a -m "修复bug"
vim file #表示更新版本号
git commit -a -m "版本更新为1.2.1"
```

完成修复分支

```
合并到master分支

git checkout master
git merge --no-ff hotfix-1.2.1
git tag -a 1.2.1 #tag操作

合并到develop分支
git checkout develop
git merge --no-ff hotfix-1.2.1

删除修复分支
git branch -d hotfix-1.2.1


![image](https://upload-images.jianshu.io/upload_images/1226129-
b2018af358d865d5.png?imageMogr2/auto-
orient/strip%7CimageView2/2/w/1000/format/webp)
```

本地提交代码

- 加入git管理

  git init

- 将全部内容添加到Git

  git add .

- 填写提交信息

  git commit -m '提交信息'

- 连接你的guthub仓库

  git remote add origin https://github.com/YanGXiDev/Flutter.git

- 上传项目到Github，可能要求输入Github的账号密码

  git push -u origin master