

## 拉取指定branch代码

```
git clone -b <branchName> https://xxx.git
```

## 查看所有的分支

```
git branch -a
```

## 查看远程分支

```
git branch -r
```

## 切换分支

```
git checkout <branchName>
```

## 创建本地分支

```
git branch <branchName>
```

## 删除本地分支

```
git branch -d <branchName>
```

## 创建远程分支

```
git branch <branchName>  
git push origin <branchName>
```

## 删除远程分支

```
git push origin --delete <branchName>
```

## 提交代码

```
git status                # 查看本地代码状态

git add .                 # 添加修改代码到缓存

git commit -am "一些信息" # 提交

git checkout -b [branchName] # 切换分支

git push origin [branchName] # push进去了!
```

如果branch已经存在，需要事先git branch -d [branch]一下，再切换分支

## 代码回滚

```
git reset --hard commit_id
```

多人协作的工作模式通常是这样：

1. 首先，可以试图用 `git push origin <branch-name>` 推送自己的修改；
2. 如果推送失败，则因为远程分支比你的本地更新，需要先用 `git pull` 试图合并；
3. 如果合并有冲突，则解决冲突，并在本地提交；
4. 没有冲突或者解决掉冲突后，再用 `git push origin <branch-name>` 推送就能成功！
5. 如果 `git pull` 提示 `no tracking information`，则说明本地分支和远程分支的链接关系没有创建，用命令 `git branch --set-upstream-to <branch-name> origin/<branch-name>`。

## 合并commit

```
git rebase -i HEAD~2
```

将第二行pick改为s，保存退出

```
pick b4d576b add b.php
s 90bc004 add c.php
s 45edfda add d.php
```

弹出新的界面修改注释

```
# This is a combination of 3 commits.
# The first commit's message is:
add b.php

# This is the 2nd commit message:
add c.php
3
# This is the 3rd commit message:
add d.php

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
# Date: Fri Jan 5 11:03:02 2018 +0800:
#
# interactive rebase in progress; onto 36224db
# Last commands done (3 commands done):
# mmit s 90bc004 add c.php mmit的注释 (缩写:r)
# mmit s 45edfda add d.php mmit的注释 (缩写:e)
# No commands remaining.
# You are currently editing a commit while rebasing branch 'ssss' on '36224db'.
#
# Changes to be committed:
#   new file:   b.php
#   new file:   c.php
#   new file:   d.php
```

这是合并后的提交注释信息

根据自己需求，修改注释

界面：

es-jianshu.io/upload\_images/2147642-03d48ee767efb307.png?imageMagick=to-

View2/AM/1240)

列出的是我们本次rebase操作包含的所有提交，下面注释部分是git为我们提供的命令说明。每一个

指令类型

commit (缩写:p)

commit和上一个commit合并 (缩写:s)

commit, 但我不要保留该提交的注释信息 (缩写:f)

commit (缩写:d)

将commit内容编辑如下:

```
git push -f origin <branchName>
```