

封装一些GitHub常用命令

我们在日常的开发过程中，肯定会经常要用到一些代码版本控制工具，其中较为常用的如GitHub，当然GitHub的命令已经比较精简了，不过依照我们每个人的个人习惯不同还是可以进行一些简单的封装的。

封装一些适用于某个项目的命令

比如说，我最近一直在维护一个开源的[Android笔记的项目](#)，这样我每天可能都会有很多次的提交，每次提交可能输入的都是那么几个命令：

```
cd /Users/mac/WorkSpace/git_android_notes
git pull
git add .
git commit -m "description"
git push origin master
```

虽然命令不是非常复杂，但是每次都需要手动输入，还是很麻烦的，所以如果我们能将其封装成一句命令就非常nice了，例如：

```
push description XXX XXX
```

其实做这样一个封装是非常简单的，但是可以帮我们省很多事情。如果您对Shell的基本命令还不是很了解，请参考[Shell脚本入门](#)

我们看一下，shell脚本的代码：

```
cd /Users/mac/WorkSpace/git_android_notes

echo "begin it ..."
```



```
git pull
git add .

git commit -m "$*"

echo $*

git push origin master

echo "finish it ..."
```

只要通过这样简单的封装，我们就可以实现我们，一行命令上传脚本的想法啦～

封装一些具有普适性的代码

- 进入到工作空间目录

封装前: `cd /Users/mac/WorkSpace` 封装后: `. me`

shell脚本代码:

```
cd ~/WorkSpace
```

- 拉取远程仓库代码

封装前: `git pull` 或 `git pull origin XXX` 封装后: `pull` 或 `pull XXX`

shell脚本代码:

```
if [ "$1" = "" ]
then
    git branch --set-upstream-to=origin/master master
    git pull
else
    git pull origin $1
fi
```

- 提交代码到远程仓库

封装前:

```
git pull
git add .
git commit -m "description"
git push origin master
```

封装后: `push master "description"`

shell脚本代码:

```
git pull
git add .
temp=$1
shift
git commit -m "$*"
git push origin $temp
```

当然，这里面只介绍了几种简单的封装，大家可以按照自己的需求，进行一些封装～