# github学习笔记

# 第一章 欢迎来到 GitHub 的世界

1.社会化编程social coding

快速了解别人的代码，多人协作编写代码。

1. github可以有力展示一个程序员的编程能力。

# Git的导入

1. Fork 就是将 GitHub 的某个特定仓库复制到自己的账户下。

# 第3章 使用github的前期准备

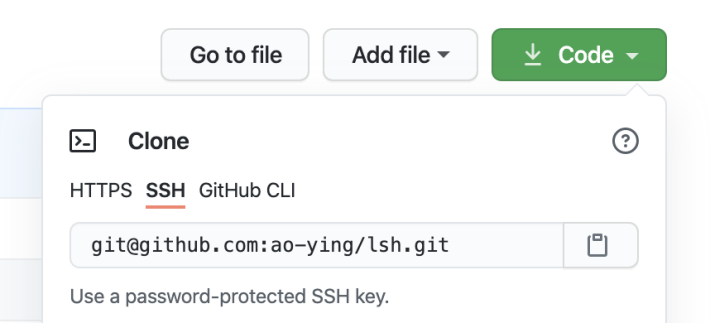
1.创建一个新的仓库

点击右上角的加号，按照提示填写信息即可。

1. git clone git@github.com:hirocastest/Hello-World.git

将网站上的仓库clone到本地。

之后就可以在本地编写文件了。

注意，克隆时选取的地址是ssh地址，如下图所示：  


1. git status（非必需）

进入clone后的仓库的目录后，查看当前文件的状态。

4.git add hello\_world.php

通过git add命令将hello\_world.php文件加入到暂存区

如果修改了多个文件，则使用git add .即可

5.git commit -m "Add hello world script by php"

通过git commit命令提交文件hello\_world.php或同时提交多个文件，并添加更新文件的注释。

也可以使用git commit之后在第一行添加注释。

6.git log（非必需）

查看git的日志，可以查到刚刚更新文件的记录。

1. git push

将文件更新到网页上的仓库中。

此时，在网页上查看即可发现文件hello\_world.php已经被添加到了相关网站中。

显示无法连接错误时的解决方法（不一定有效）:这个错误可能是第2.步地址错误引起的。

git config --global --unset http.proxy

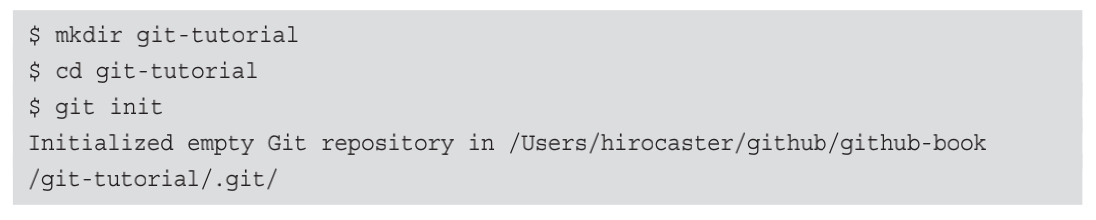
git config --global --unset https.proxy

8.

# 通过实际操作学习Git

1. 创建一个仓库并准备提交涉及的基本操作

git init——初始化仓库

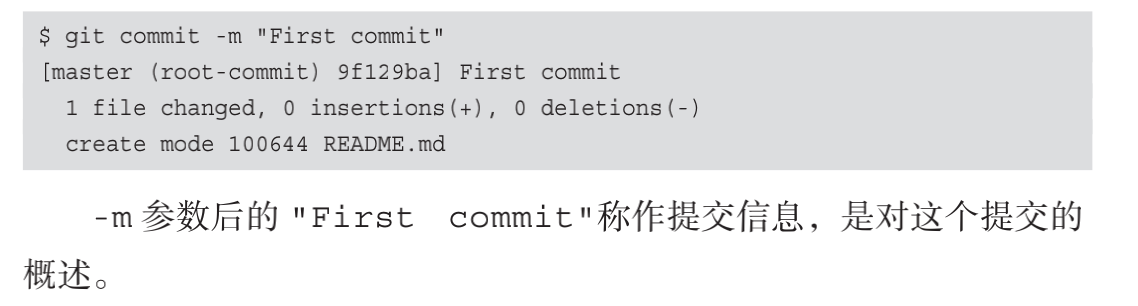


git status——查看仓库的状态

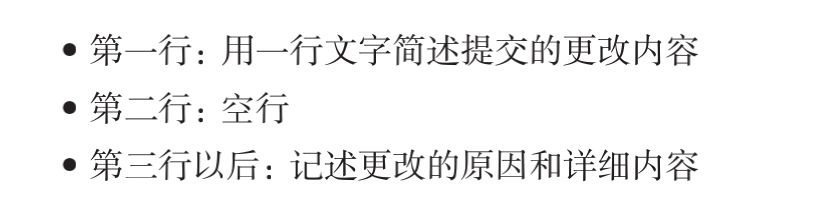
touch README.md——新建readme文件

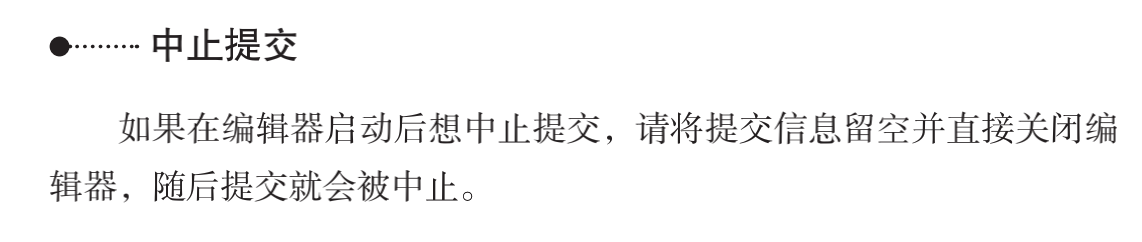
git add——向暂存区中添加文件。暂存区是提交之前的一个临时区域。

git commit——保存仓库的历史记录。git commit命令可以将当前暂存区中的文件实际保存到仓库的历史记录中。



如果要添加多行说明，则输入git commit，之后按照下面的格式进行编写：





2.git log相关

git log——查看提交日志

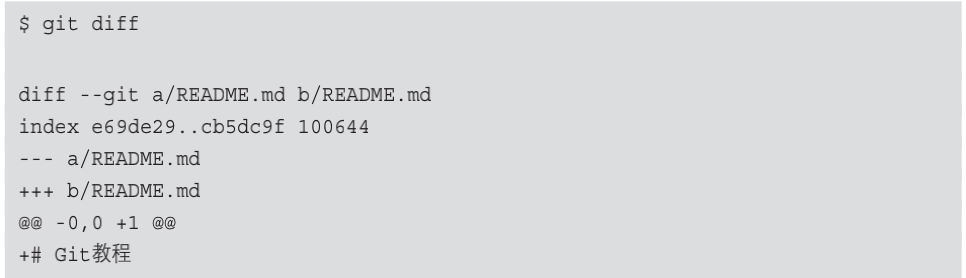
git log --pretty=short 只想让程序显示第一行简述信息

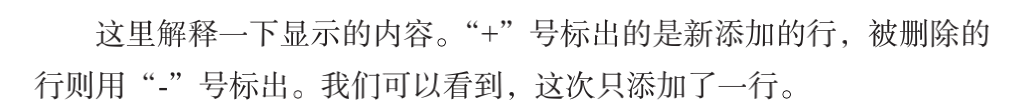
git log目录名/文件名 只显示指定目录、文件的日志

git log -p 显示文件的改动

各种操作可以联用，eg：git log -p README.md

1. git diff——查看更改前后的差别



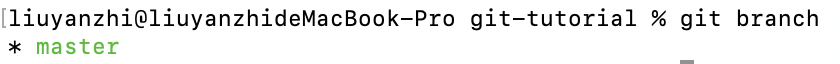


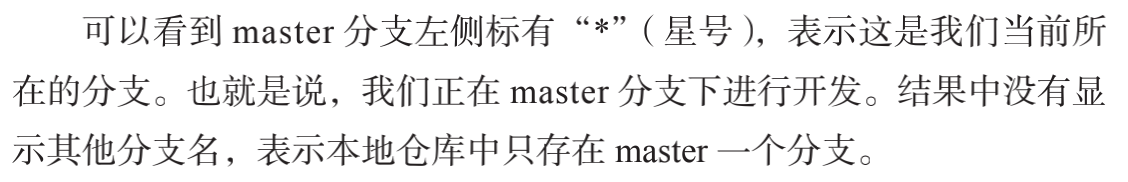
4.查看本次提交与上次提交之间的差别

不妨养成这样一个好习惯：在执行 git commit命令之前先执行git diff HEAD命令，查看本次提交与上次提交之间有什么差别，等确认完毕后再进行提交。这里的 HEAD 是指向当前分支中最新一次提交的指针。

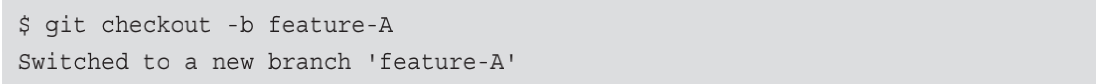
1. 分支基本操作

git branch——显示分支一览表。将分支名列表显示，同时可以确认当前所在分支。

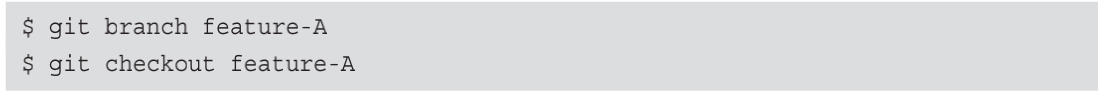




git checkout -b——创建一个分支，并切换到新创建的分支



等价于



之后对文件进行的修改，add，commit等操作都将会记录在feature-A这个分支中

git checkout master 切换回master分支

这时会发现，刚刚对feature-A中文件的修改消失了。

git checkout -切换回上一个分支

1. 合并分支

将feature-A分支的内容合并到master分支。

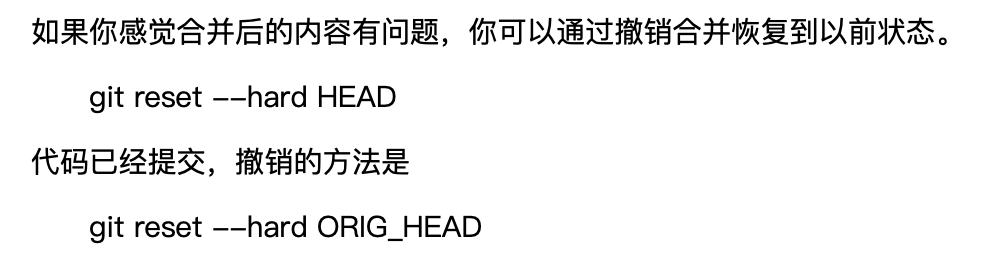
git checkout master 先切换到master分支

git add README.md

git commit -m "Add feature A"

git merge --no-ff feature-A 之后进行合并

撤销合并分支的方法：不太行



出现问题之后强行继续合并：git merge feature-A

关闭编辑器的方法: esc+wq

git log --graph——以图表形式查看分支

