1. 配置

$ git config --global user.name "Your Name"

$ git config --global user.email "email@example.com"

1. 创建版本库

git init

1. 文件添加至版本库

添加到暂存区：git add filename

提交到版本库：git commit –

m “chane log”

1. 日志

记录历史版本：git log （以一行显示log，加参数）–-preety=oneline

记录版本管理历史：git reflog

1. 版本切换

上一个版本是HEAD^，上上一个版本是HEAD^^，当然往上100个版本HEAD~100。

git reset --hard HEAD^

git reset --hard commit\_id (git reflog查看commit\_id)

1. 查看工作区、暂存区和版本库间的状态

git status

git diff HEAD -- filename命令可以查看工作区和版本库里面最新版本的区别

1. 撤销修改回到最近一次git commit或git add时的状态

git checkout -- filename

1. 删除

git rm filename

1. 创建ssh key

ssh-keygen -t rsa -C [youremail@example.com](mailto:youremail@example.com)

id\_rsa是私钥，不能泄露出去，id\_rsa.pub是公钥，可以放心地告诉任何人，存入Github的SSHKEY中。

1. 远程github

把已有的git仓库推送至GitHub：

远程关联：git remote add origin [git@github.com:michaelliao/learngit.git](mailto:git@github.com:michaelliao/learngit.git)

推送：master 至 origin:

git push -u origin master

克隆：

git clone git@github.com:michaelliao/gitskills.git

1. 分支管理

创建分支

git branch name

查看分支

git branch

切换分支

git checkout name

删除分支

git branch –d name

git branch –D name 不合并时强制删除

合并分支(快速合并name至当前分支）

git merge name

创建并切换分支

git checkout -b name

合并分支冲突需手动解决，重新commit，查看分支合并图：

git log --graph --pretty=oneline --abbrev-commit

当前工作现场临时“储藏”起来

git stash

临时储藏列表

git stash list

恢复并删除储藏

git stash pop

恢复git stash apply 删除 git stash drop

多人协作：

查看远程库的信息：git remote -v

推送分支：git push origin dev

建立与远程仓库对应分支：git checkout -b dev origin/dev

远程仓库与本地合并：git pull origin dev

1. 标签

查看：git tag <tagname>

创建标签：git tag –a tagname –m “…”

删除本地：git tag -d v0.1

删除远程：git push origin :refs/tags/v0.9

推送：git push origin v1.0 git push origin –tags（全部推送）

1. 忽略文件

文件：.ignore

强制增加忽略文件：git add -f <name >

https://github.com/github/gitignore