2018-03-15T08:54 git学习{不跟踪视频和图片这样的二进制文件}

1. 安装完成后使用命令

“git config - - global user.name “Your Name””

“git config - - global user.email “email@example.com””

1. 创建版本库：（repository）这是一个目录，此目录中的所有文件都可以被Git管理起来，每个文件的修改、删除，Git都能跟踪。【pwd用于查看当前目录】【微软的word为二进制文件，无法跟踪】
2. 初始化仓库：“git init”【使用ls –ah查看目录下隐藏的文件】
3. 使用“git add XXX”来讲文件添加到仓库【**add是把文件从工作区添加到暂存区**】
4. 用命令“git commit –m XXX”告诉Git，把文件提交到仓库。“-m” 后面为本次提交说明【commit可以一次提交多个文件，可以多次add之后一起提交】【**commit将文件从暂存区所有内容提交到当前分支**】
5. “git status”查看掌握仓库当前的状态。
6. “git diff”查看difference，查看具体修改了什么。
7. 每当觉得文件修改到一定程度的时候，可以“保存一个快照”，这个快照在Git中被称为“commit
8. “git log - - pretty=oneline” 输出单行的日志
9. 【Git中，HEAD表示当前版本，HEAD^表示上一版本，HEAD^表示上上一个版本】

“git reset”命令退回到上一个版本

“git reset - -hard HEAD”

11、“git reflog”记录你的每一条命令

12、工作区有一个隐藏目录 .git（为Git的版本库），Git的版本库里存了很多东西，其中stage（或者叫index）为暂存区。还有Git为我们自动创建的第一个分支master，以及指向master的一个指针叫HEAD

13、“git diff HEAD - - XXX”查看版本库中的XXX文件的最新版本和工作区中的XXX文件的区别

14、“git checkout **- -** XXX”，把XXX文件在工作区的修改全部撤销。【如果文件还没提交到暂存区，则工作区中的文件与版本库中同步。如果文件提交到了暂存区，则工作区中的文件与暂存区中同步】【git checkout 是用版本库里的版本替换工作区的版本】

15、“git reset HEAD XXX”可以把暂存区的修改回退到工作区！！【HEAD表示最新的版本】

16、“rm XXX”在文件管理器中把XXX文件删除。【使用git checkout 恢复】

17、“git rm XXX”在版本库中删除XXX文件

2018-03-15T09:30 不要使用微软自带的记事本编辑任何文本文件！！！使用notepad++默认编码设置为UTF-8 without BOM

18、“git push”将当前分支master推送到远程【**git push origin master推送到远程库**】

19、克隆一个仓库，“git clone XXX”，XXX为仓库地址

20、Git支持多种协议，包括https和ssh，但通过ssh支持的原生git协议速度最快

21、HEAD指向当前分支，而当前分支指向最新提交的。【当创建新的分支dev时，dev先指向最新提交，HEAD指向dev】

22、分支命令：

●查看分支：git branch

●创建分支：git branch <name>

●切换分支：git checkout <name>

●创建+切换分支：git checkout –b <name>

●合并某分支到当前分支：git merge <name>【通常为快进合并，冲突时不可快进合并】

●删除分支：git branch –d <name>

23、“git log -- graph”查看分支合并图

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

24、“git merge - -no-ff –m “XXX”<branch\_name>”【- -no-f表示禁用Fast forward。普通模式合并，合并后的历史有分支，能看出来曾经做过合并，fast forward看不出来】

25、mater分支通常只用来发布新版本，不在上面干活。都在dev分支上干活，每个人再从dev上分出自己的分支。

26、当手头工作没有完成时，先把工作现场git stash一下，然后去修复bug，修复后，再git stash pop，回到工作现场。【git stash list查看挂起的工作的列表】

27、没添加一个新功能，最好添加一个新的分支。【git branch –D <name>强行删除】

28、查看远程库的信息：git remote

29、git push origin master【推送到master分支】

git push origin dev【推送到dev分支】

master和dev分支需要推送到远程，其他的看自己。

30、远程创建分支：git checkout –b dev origin/dev

31、在本地创建和远程分支对应的分支，使用 git checkout –b branch-name origin/branch-name

32、建立本地分支和远程分支的关联，使用 git branch - -set-upstream branch-name origin/branch-name

33、从远程抓取分支，使用git pull

34、标签类似于commit，但是，commit作为指针可以移动，标签不能移动。

35、标签

●新建一个标签：git tag <name>，默认为HEAD，可以制定一个commit id

●指定标签信息：git tag –a <tagname> -m “XXX”

●用PGP签名标签：git tag –s <tagname> -m “XXX”

●查看所有标签：git tag

36、管理远程标签：

●推送一个本地标签：git push origin <tagname>

●推送全部未推送的本地标签：git push origin - -tags

●删除一个本地标签：git tag –d <tagname>

●删除一个远程标签：git push origin :refs/tags/<tagname>

2018-03-15T11:41 SSH秘钥保证身份验证，如果多个计算机则提交多个SSH秘钥

2018-03-15T13:08 本地远程连接GitHub

“git remote add origin git@github.com:wangshuoqiong/learngit.git

git push -u origin master”

如果输入错误使用“git remote rm origin”删除绑定

2018-03-15T16:07 当配置文件中的信息需要具备顺序含义时，YAML的配置方式比起properties配置文件更有优势。

2018-03-15T16:53 **C/C++输出百分号，应该连续两个%表示**

2018-03-15T17:06 PGP密码保护私钥：xyxyan99