### GIT

1. 概述

git属于分布式（分开放置），每个人都有一套完整的代码，服务器上也有一套完整的代码。本地每个版本都有。

svn属于集中式，全部上传到一台服务器。本地只有一个版本。

1. 安装
2. 右键单击“我的电脑”→ “属性”→ “系统类型”查看本机是32位还是64位系统；
3. 到git官网https://git-scm.com/downloads 下载32位/64位的exe文件；
4. 击右键，按默认设置安装；
5. 安装成功的话，在桌面或其他文件夹击右键会出现“git Bash here”；
6. 打开，开始使用。
7. 路径基本知识
8. 计算机操作系统：
9. win
10. 类unix: ubuntu, redhat, centos, mac OS
11. 路径表示：
12. / 正斜线，在linux中没有盘符的概念，只有一个系统根目录 / ，/Users/itcast/Desktop

git原本就是在linux中运行的程序，后被移植到windows中。

1. \ 反斜线，在window操作系统中，盘符：\ 路径 \ 文件名
2. 关于换行符：在windows中，一个换行相当于一个换行加一个回车，用字符表示，就是“\r\n”；在linux中，一个换行就是一个换行，如果用字符表示，就是“\n”
3. 路径win(C:)\用户\ciro是当前家目录home，用~表示，linux命令中使用cd切换目录change directory.

|  |  |  |  |
| --- | --- | --- | --- |
| cmd命令 | | | |
| copy /b 1.mp3+1.mp3 2.mp3 | | 以二进制合并文件 | |
| mkdir test | 创建文件夹 | cd.. | 回到上一级目录 |
| rmdir test | 删除文件夹 | d: | 切换到d盘 |
|  |  | dir /a | 显示所有文件 |

|  |  |  |  |
| --- | --- | --- | --- |
| Git命令 | | | |
| init | 初始化仓库 |  |  |
| cd~ | 切换到家目录 | ls | 列出该目录下的所有文件 |
| cd ~/Desktop  (tilde) | 切换至桌面 | -a | 列出所有的文件 |
| cd /d | 切换至d盘 | clear | 清除命令 |
| cd .. | 回到上一级 | rmdir xx | 删除空文件夹 |
| / | 根目录 |  |  |
| .或./ | 当前目录 | ctrl + c  close | 按错键了用这个退出 |
| ..或../ | 上一级目录 |  |  |
| touch 文件名 | 创建文件 | rm xx | 删除文件 |
| mkdir 文件夹名  make directory | 创建文件夹 | rm -r  recursion  rm -rf  force | 删除所有文件夹  直接删除无提示 |
| 在linux中所有以**.**开头的文件都是隐藏的文件 | | | |

1. 开始使用
2. git init
3. git config --global,写上global的话，这台电脑上所有git仓库都使用这个配置。
4. 配置用户名git config user.name “testName”

git config --global user.name “testName”

配置邮箱 git config user.email [“test@sina.com”](mailto:\“test@sina.com\”)

查看配置信息git config --list

1. cat ~/.gitconfig 查看注册信息
2. git把项目所在的文件夹分成三个部分：工作区、暂存区、仓库

.git里面的内容分成两个部分：暂存区、仓库

1. 先把代码添加到暂存区（相当于把麦子从田地里转移到打麦场）

命令：git add 文件路径

示例：git add ./reademe.md

1. 把暂存区的文件提交到仓库里（把麦子收到仓库）

命令：git commit -m “This is explaination / message”

示例：git commit -m “添加了一个新功能”

1. -m指定一个字体串作为本次提交的说明，相当于注释。
2. git add . 表示把当前目录下的所有文件都加进去。
3. 合并add与commit命令：git commit -a -m “注释”

这里的**-a**参数表明把所有修改后的文件一起添加到暂存区，只对修改后的文件有效，对于新添加的文件没有作用。

1. git status

如果commit提交的时候忘记-m会自动进入vi编辑器状态，那么需要知道如何退出。vi编辑器有两种状态： 编辑状态、命令状态。

按vi 文件名，进入该文件里进行编辑，编辑完成后，按esc退出编辑状态进入命令状态（按esc确保命令状态），然后按：wq保存退出。如果只按了：q的话就会提示”已修改但尚未保存（可用！强制执行）“，这样就不会保存直接退出。在命令状态下，按i可以进入编辑状态，输入其他内容。

按vi，进入一个新的空白文件里面，输入完成后，按：wq 新的文件名，就可以把内容存到这个新文件中。

输入vi 3.js即可创建临时文件。

|  |  |  |  |
| --- | --- | --- | --- |
| q quit | w write | i insert | pwd present working directory |

1. ignore

不想上传的文件名称要放在.gitignore里，同时把.gitignore也放在.gitignore里面，这样.gitignore就不会报红色。

创建.gitignore文件的三个方法：

1. 高级记事本：在sublime中建
2. 右键→新建→文件名为.gitignore.
3. 用touch命令新建.gitignore
4. 多数不提交，但有几个要提交，要用！否定。比如先把./src/全部放到ignore里面，然后把里面需要提交的文件路径前面加！。
5. 对比工作区和暂存区的差异：git diff。如果暂存区没有文件，就会将工作区的代码与最近一次提交的代码（仓库中）对比：git diff --cached.

对比之前某两次提交的文件的差异：git diff [版本号1][版本号2][比较的文件路径]，示例：git diff 3d512da56 34df45fd3 readme.md

1. 查看日志

命令：git log，可以查看每一次提交的日志

命令：git log --oneline，表示使用简洁的形式输出提交日志--pretty

命令：git log --graph，表示以图的形式显示日志

1. 功能回退：git reset --hard Head 回到上一次提交。--hard表示重置并覆盖工作区内容。

git reset --hard Head~1 回到上上一次提交，2表示上上上次提交。

git reset --hard SHA1值（版本号） 回到任意一个版本。

git reflog(reference) 可以查看之前所有版本切换的操作记录，可以通过这个命令得到的版本号回退到指定的版本。

1. 分支

为什么使用分支？互不影响。分支有feature/bug-fix/develop

|  |  |
| --- | --- |
| 创建新的分支 | git branch [branch name] |
| 查看当前所有分支 | git branch |
| 切换分支 | git checkout [branch name]  切换分支后可以在切换后的分支中进行正常的操作 |
| 合并分支 | git merge [branch name]  git会将指定的分支合并到当前分支 |
| 删除指定分支 | git branch -d[branch name]  -d参数表示要执行删除操作 |

如果git不能自动合并分支，就会有冲突，我们要**手动**解决冲突，然后再次提交。

### GITHUB

1. git是版本管理工具，github是一个提供git服务器功能的网站。国内提供同类服务的是git.oschina.net 开源中国（码云）.gitlab个人服务器

上传代码到git服务器用push，下载用down.

命令：git push [远程服务器地址] [远程服务器的分支]

示例：git push <https://github.com/huoqishi/test.git> master

上传时可以使用一些简化的命令，将远程服务器地址写成变量的形式

命令：git remote add [变量名] [远程服务器地址] 仅本仓库可用

示例：git remote add origin <https://github.com/huoqi/test.git>

这样以后就可以直接使用origin来代替git push后面的一串地址了，

例如git push origin master.

git remote 查看存储的变量。还可以进一步简化，在push时加上-u参数，

git push -u origin master就会默认建立本地当前分支与远程指定分支的关联，以后在该分支提交到服务器相应的分支，下一次push时就不需要输入分支名了git push origin.

1. git生成公钥和私钥

ssh-keygen -t rsa在用户家目录的.ssh 文件夹内生成公钥私钥。git服务器设置公钥，并设置上传代码地址为ssh方式地址。公钥和私钥只和计算机有关，即使是同一个用户，在不同的电脑上也不行。不同的计算机、不同的项目都需要不同的公钥。key generate 注册机

1. 服务器与本地代码冲突fix

在每次要push文件到服务器前，先从服务器pull代码到本地，pull会自动将服务器的文件与本地文件进行合并，合并成功后再执行push文件到服务器。若pull产生冲突，则需要在本地修改文件后，再次pull文件到本地，确保没有冲突后，再进行push操作。push，pull 只能针对仓库

用pull的方法下载文件的缺点是：push时需要输入用户名密码。

git clone 网址 也可以下载代码，并且如果从仓库clone，push时后面什么都不用写。

1. 利用github搭建blog
2. 创建一个项目
3. 创建gh-pages分支
4. 将项目上传至该分支
5. 利用 [用户名].github.io/[]项目名]/[页面] 访问

git branch gh-pages 创建博客分支，然后把项目上传到分支当中，就可以访问了，访问的url形式：[github用户名].github.io/[仓库的名字]/[具体的页面]

### **tortoiseGit** git图形化工具