

参考资料

码云官网底部:



深圳市奥思网络科技有限公司版权所有

关于我们

使用条款

Git 大全

Git 命令学习

Git 大全

// 从工具安装下载, 插件、服务搭建, 命令行

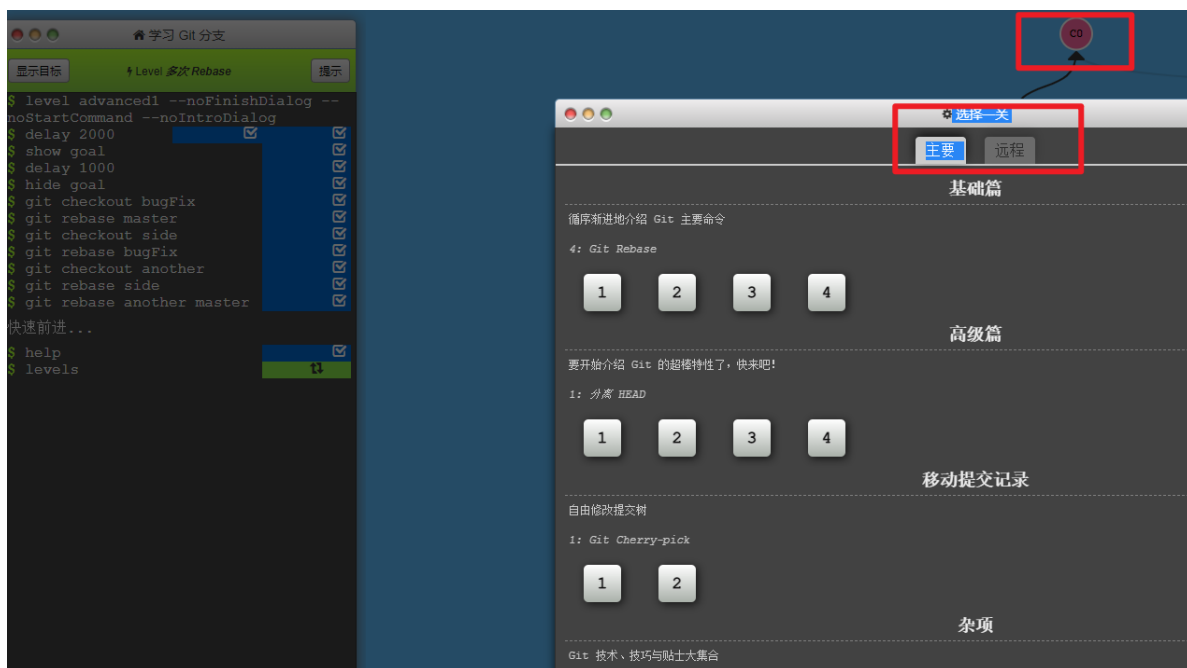
工具下载经验: 国外工具在国内镜像的下载

<http://npm.taobao.org/mirrors/git-for-windows/>

搜索: taobao 镜像 git // 直接访问<http://npm.taobao.org/mirrors>会提示404

Git 命令学习

动画演示, 循序渐进学习命令, 所见即所得



狂神说 B站视频

Git环境配置

`git config -l` 查看全局和系统配置

```
git config -l
```

```
$ git config -l
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=openssl
http.sslcainfo=D:/Environment/Git/mingw64/ssl/certs/ca-bundle.crt
core.autocrlf=true
core.fscache=true
core.symlinks=false
core.editor="C:\\Program Files\\Notepad++\\notepad++.exe" -multiInst -
notabbar -nosession -noPlugin
credential.helper=manager
user.name=kuangshen
user.email=24736743@qq.com
```

`git config --system -l`

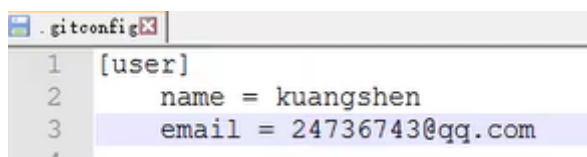
查看系统配置，系统默认配置

`git config --global -l`

查看用户配置

```
$ git config --global --list
user.name=kuangshen
user.email=24736743@qq.com
```

1. 默认保存在.gitconfig文件中，c盘 用户目录下文件保存



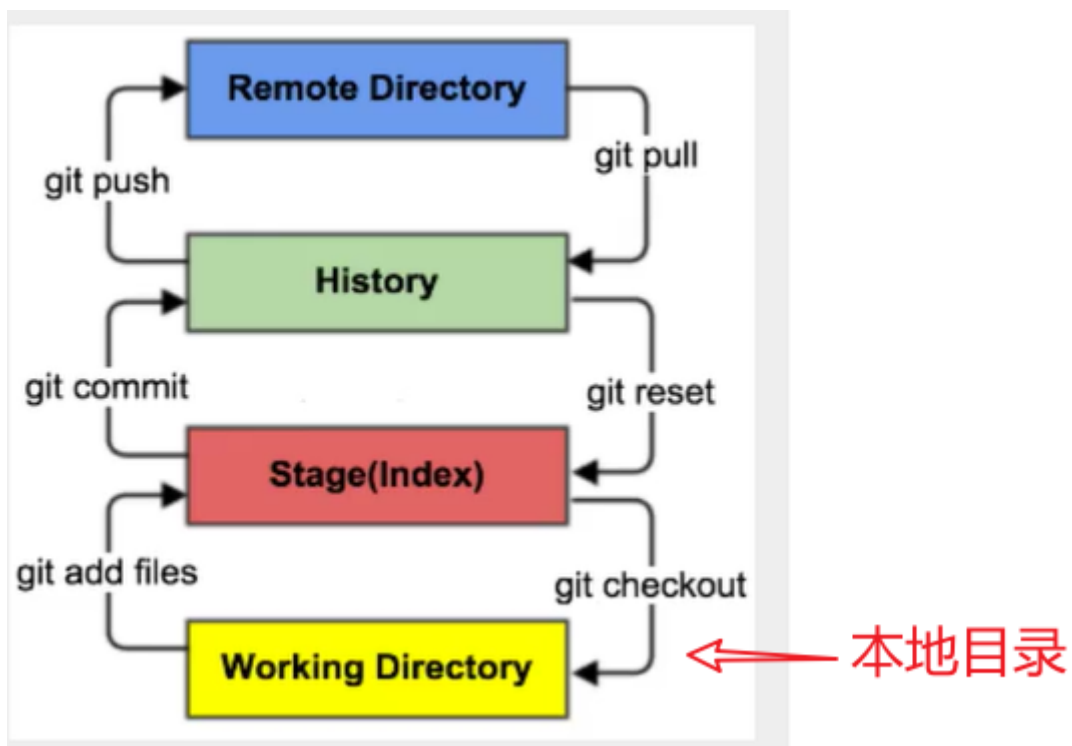
```
.gitconfig
1 [user]
2   name = kuangshen
3   email = 24736743@qq.com
4
```

2. 设置用户配置

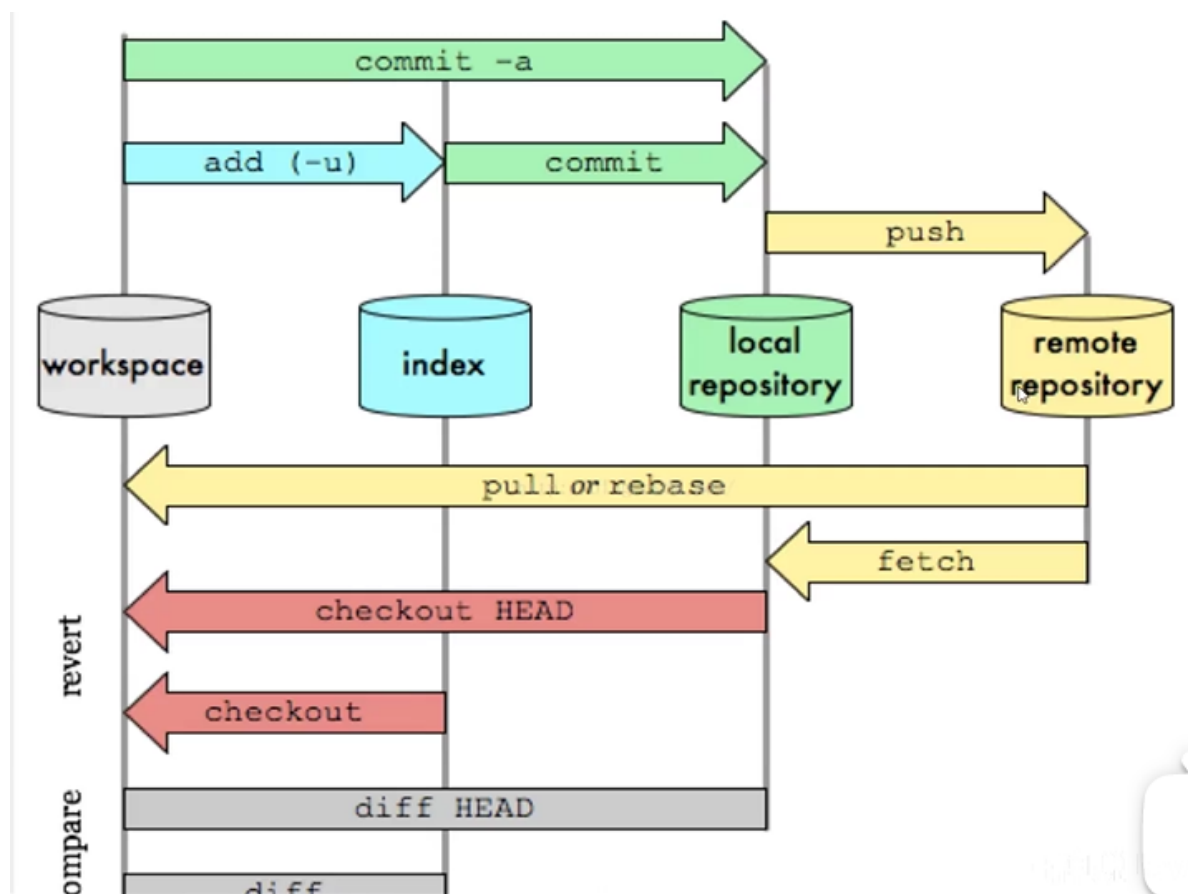
```
git config --global user.name "kuangshen" #名称
git config --global user.email 24736743@qq.com #邮箱
```

Git理论

Git工作区域：工作目录-》暂存区Stage-》本地仓库（Local Repository）->远端git仓库（Remote Directory）



Git命令操作



1. 本地仓库搭建

1. git init: 目录下新建仓库
2. git clone 路径

2. 文件状态

文件有4种状态

- **Untracked:** 未跟踪, 此文件在文件夹中, 但并没有加入到git库, 不参与版本控制. 通过 `git add` 状态变为 **Staged**.
- **Unmodify:** 文件已经入库, 未修改, 即版本库中的文件快照内容与文件夹中完全一致. 这种类型的文件有两种去处, 如果它被修改, 而变为 **Modified**. 如果使用 `git rm` 移出版本库, 则成为 **Untracked** 文件
- **Modified:** 文件已修改, 仅仅是修改, 并没有进行其他的操作. 这个文件也有两个去处, 通过 `git add` 可进入暂存 **staged** 状态, 使用 `git checkout` 则丢弃修改过, 返回到 **unmodify** 状态, 这个 `git checkout` 即从库中取出文件, 覆盖当前修改!
- **Staged:** 暂存状态. 执行 `git commit` 则将修改同步到库中, 这时库中的文件和本地文件又变为一致, 文件为 **Unmodify** 状态. 执行 `git reset HEAD filename` 取消暂存, 文件状态为 **Modified**

命令

```
# 查看所有文件状态
git status
# 上传文件到暂存区
git add .
# 提交暂存区文件到个人本地仓库, m代表message
git commit -m
```

3. 文件.gitignore

配置忽略不提交暂存区文件

在主目录下建立".gitignore"文件, 此文件有如下规则:

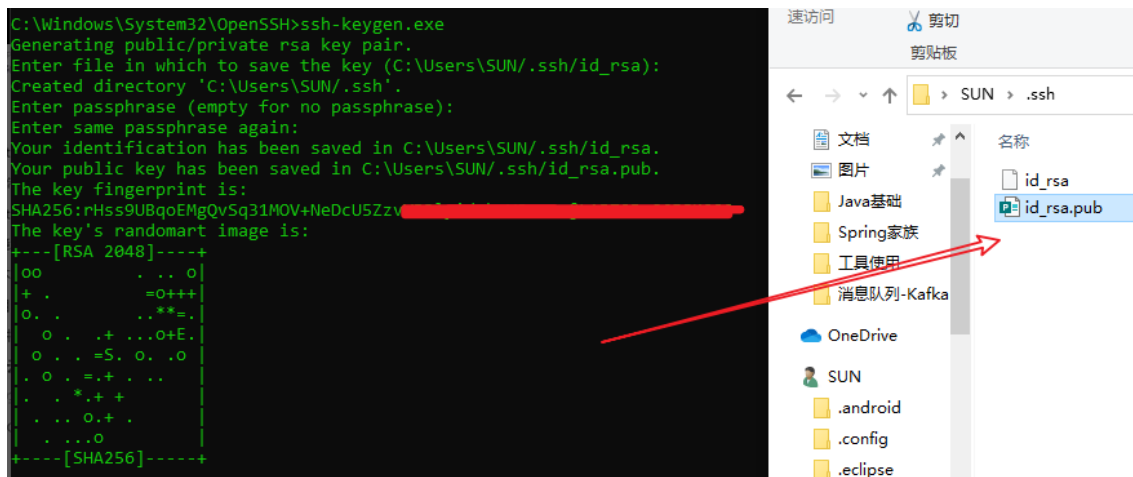
1. 忽略文件中的空行或以井号(#)开始的行将会被忽略。
2. 可以使用Linux通配符。例如 * 星号(*)代表任意多个字符, 问号(?)代表一个字符, 方括号([abc])代表可选字符范围, 大括号({string1,string2,...})代表可选的字符串等。
3. 如果名称的最前面有一个感叹号(!), 表示例外规则, 将不被忽略。
4. 如果名称的最前面是一个路径分隔符(/), 表示要忽略的文件在此目录下, 而子目录中的文件不忽略。
5. 如果名称的最后面是一个路径分隔符(/), 表示要忽略的是此目录下该名称的子目录, 而非文件(默认文件或目录都忽略)。

```
#为注释
*.txt      #忽略所有 .txt结尾的文件
!lib.txt   #但lib.txt除外
/temp      #仅忽略项目根目录下的TODO文件, 不包括其它目录temp
build/     #忽略build/目录下的所有文件
doc/*.txt  #会忽略 doc/notes.txt 但不包括 doc/server/arch.txt
```

使用码云

1. 设置本机绑定SSH公钥, 实现免密登录

```
# 进入 c:\users\登录名\.ssh目录
ssh-keygen
```



2. 将公钥信息添加码云账号，SSH配置，将pub文件所有内容放入
3. 使用码云界面创建个人仓库：仓库名称，开源许可证信息
4. git clone克隆到本地
- 5.

Git 分支操作

1. 常用命令

```
# 列出所有本地分支
git branch
# 列出所有远程分支
git branch -r
# 新建一个分支，但仍停留在当前分支
git branch [branch-name]
# 新建一个分支，并切换该分支
git checkout -b [branch]
# 合并指定分支到当前分支
git merge [branch]
# 删除分支
git branch -d [branch-name]

# 删除远程分支
git push origin --delete [branch-name]
git branch -dr [remote/branch]
```

2. 冲突解决

1. 拉取远端仓库（参考remote仓库操作）
2. 根据提示解决冲突，提交本地仓库
3. 提交远端仓库

Git 远程remote仓库操作

查看所有远程仓库

```
# 查看所有远程仓库信息
git remote -v
# 添加远端仓库
git remote add [shortname] [url]
# 删除远程仓库
git remote rm name

# 拉取远端仓库代码
git pull [远端仓库shortname]
```

样例:

```
$ git remote -v
origin https://github.com/tianqixin/runoob-git-test (fetch)
origin https://github.com/tianqixin/runoob-git-test (push)

添加远端仓库
$ git remote add origin git@github.com:tianqixin/runoob-git-test.git
```