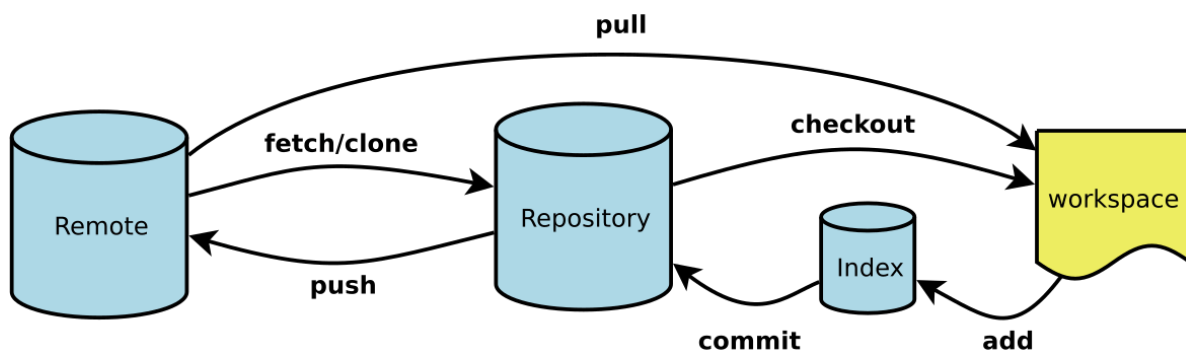


Git常用操作



1. 创建版本库

```
$git init                #初始化本地版本库
$git clone <url>          #克隆远程版本库、默认所有分支
$git clone -b <branch> <url> #克隆远程版本库、单个分支
$git branch -r            #列出远程跟踪分支
$git checkout <branch>    #切换分支
```

2. 修改和提交

```
$git status              #查看状态
$git diff                #查看变更内容
$git add .                #跟踪所有改动后的文件
$git add <file>           #跟踪指定的文件
$git mv <old> <new>       #文件改名
$git rm <file>            #删除文件
$git rm -cached <file>    #停止跟踪文件但是不删除
$git commit -m "commit message" #提交所有更改过的文件，单行注释
$git commit -m '
> msg line1
> msg line2
$ >'                    #提交所有更改文件、多行注释

$git commit --amend       #修改最后一次提交
```

3. 查看提交历史

```
$git log                 #查看提交历史
$git log -p <file>       #查看指定文件提交历史
$git blame <file>        #以列表形势查看制定文件提交历史
```

4. 撤销

```
$git reset --hard HEAD      #撤销工作目录中所有未提交文件的修改内容
$git checkout HEAD <file>   #撤销指定的未提交的文件的修改内容
$git revert <commit>        #撤销指定的提交
```

5.分支与标签

```
$git branch                #显示所有本地分支
$git checkout <branch/tag> #切换到指定分支或者标签
$git branch <new_branch>   #创建新分支
$git branch -d <branch>    #删除本地分支
$git tag                   #列出所有本地标签
$git tag <tagname>         #基于最新提交创建标签
$git tag -d <tagname>     #删除标签
```

6.合并与衍合

```
$git merge <branch>      #合并指定分支到当前分支
$git rebase <branch>     #衍合指定分支到当前分支 （有雷慎用）
```

7.远程操作

```
$git remote -v            #查看远程版本库信息
$git remote show <remote> #查看指定远程版本库信息 <remote>:代表远程主机名
$git remote add <remote> <url> #添加远程版本库
$git fetch <remote>       #从远程库获取代码
$git pull <remote> <branch> #下载代码及快速合并
$git push <remote> <local_branch>:<remote_branch> #上传代码及快速合并
$git push <remote> <branch> #上传代码及快速合并
$git push <remote>:<branch/tag-name> #删除远程分支或标签
$git push --tags           #上传所有标签
$git push --set-upstream <remote> <branch> #推送当前分支并建立与远程上游的跟踪
```

8.用户名存储

```
$git config --global credential.helper store #存储 用户名 密码 避免每次 clone 或者 push 输入
$git config --global credential.helper reset #更换密码 重置
```

9.忽略文件三种情况

1) 新建.gitignore 文件、忽略提交文件

示例 .gitignore，注意.gitignore 也要提交到配置库

```
.DS_Store
node_modules/
dist/
npm-debug.log
build-debugger.js
.idea
*.so
```

2) 已经推送 (push) 过的文件, 想从git远程库中删除, 并在以后的提交中忽略, 但是却还想在本地保留这个文件

```
$git rm --cached Xml/config.xml
```

3) 已经推送 (push) 过的文件, 想在以后的提交时忽略此文件

已经推送 (push) 过的文件, 想在以后的提交时忽略此文件, 即使本地已经修改过, 而且不删除git远程库中相应文件执行命令。后面的 Xml/config.xml 是要忽略的文件的路径。适用于: git远程库上有一个标准配置文件, 然后每个人根据自己的具体情况, 修改一份链接信息自用, 而且不会将该配置文件提交到库

```
$git update-index --assume-unchanged Xml/config.xml
```

10.配置邮箱及用户名

初次commit之前, 需要配置用户邮箱及用户名, 使用以下命令:

```
$git config --global user.email "you@example.com"
$git config --global user.name "Your Name"
```

11.其他命令

\$git config --list	#显示当前的Git配置
\$git config -e [--global]	#编辑Git配置文件
\$git --help	#调出Git的帮助文档
\$git --help <cmd>	#查询单个Git命令的帮助文档