

Git

版本控制发展

文件管理

单机版本控制

集中式版本控制

SVN

分布式版本控制

Git

基本操作

cd 进入被管理的文件夹

初始化

git init

操作

git status 检测状态

红色的是新添加未被管理的文件

git add ____ (文件名/文件夹)

绿色是正在管理的文件

git commit -m "备注"

最后的提交

版本库操作

git log

查看当前状态的版本

git reset --hard 版本号

回到老版本

git reflog

查看所有的版本以及跳转记录

工作区操作

git checkout 文件名

使本次修改失效

暂存区操作

git reset head 文件名

绿→红

分支 branch

命令操作

git branch

获取当前分支

git branch dev

创建新的分支dev

git checkout dev

切换到dev

后续都是正常操作

git merge bug

将bug分支合并到当前分支

有时候会自动合并有冲突,需要手动解决

工作流

dev 开发

公测版本

master 正式版本,最稳定

正式版本

不许添加乱乱的代码

GitHub

推送 push

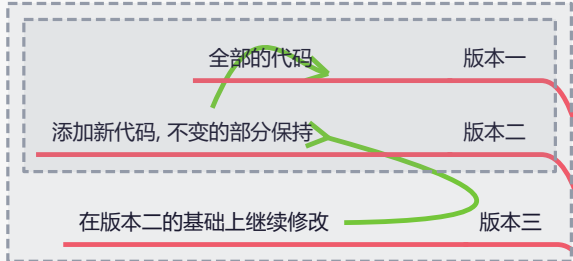
创建仓库

链接仓库

克隆(第一次)

出现错误

更新 pull



以前的方法只能暂停到某一个版本,而不能保留中间状态.
branch 可以保留中间状态.

.....

git remote add origin https://github.com/wangsong008/mytestres.git
git branch -M master
git push -u origin master

git clone https://github.com/wangsong008/mytestres.git

git config --global --unset http.proxy
git config --global --unset https.proxy

git pull origin master