

Git 代码下载

1.git 协议 `git clone git://192.168.1.183/库名称`

2.http 协议 `git clone http://192.168.1.183/gitweb/库名称`

3.SSH 协议 `git clone username@192.168.1.183:/home/repo/库名称`

例如: `git clone magnus@192.168.1.183:/home/repo/mozilla-esr24.git`

Git 操作

此处仅列举比较常用的 git 操作

1.配置 git 环境

配置用户名和用户 EMAIL

`git config --global user.name "username"`

`git config --global user.email "username @domain name"`

其他配置请参考 `git config --help`

git 开发人员日常基本操作

(1)将本地代码与服务器同步命令: `git pull origin 分支名称`

(2)修改需要修改的文件, 修改完毕查看当前修改的文件命令为: `git status`

(3)将修改的文件添加到版本树:

a.单个文件 `git add filename`

b.添加工作目录下的所有文件 `git add -A`

(4)提交代码到本地版本库 `git commit -m "commit comment"`,如果注释比较长也可

以用以下命令: `git commit` 回车, 在弹出的界面中输入你的注释, 书写完毕后按 `ctrl+X` 然

后按 `YES` 保存修改即可

如果不小心写错了注释用一下命令修改最近的提交注释 `git commit -amend`

(5)注释写完后就需要将自己的修改打成 `patch` 发给代码提交人员，打 `patch` 的命令为: `git`

`format-patch origin` 将目前的修改打成 `patch`。

如果想只将一个提交打成 `patch` 则命令为: `git format-patch -1 提交号`

(6)将本地代码推送到远程版本库 `git push origin master`

(此操作目前大家没有权限可以忽略)`origin` 是代码的源可以自己添加默认的名称为 `origin`

`master` 是远程版本库的分支名称，可以根据实际情况写上分支名称，默认为 `master`

分支

(1)创建分支 `git branch branchname`

以某个分支为基础创建新的分支 `git branch newbranch oldbranch`

(2)删除分支 `git branch -d branchname` 如果要强制删除则用 `-D`

(3)切换分支

`git checkout branchname`

(4)分支合并

`git merge` 需要合并的分支名称 `-squash` 此参数表示将另一条分支的所有提交合并为当前分支上的一个提交

然后 `git commit -m "commit comment "`

拣选合并 `cherry-pick`

`git checkout master`

`git cherry-pick` 其他分支的提交版本号

(5)分支重命名

`git branch -m oldbranchname newbranchname`