# 实验五

## 实验目的

1、 实现分支的创建、合并。

2、 学习解决分支冲突。

3、 了解分支管理策略。

4、 掌握与远程仓库的分支交互。

## 实验内容

完成分支的创建、合并，解决分支冲突和远程仓库的分支交互。

## 实验步骤

在版本回退中， 大家应该已经了解了每次提交， Git 都把它们串成一条时间线，这条时间线就是一个 分支。

在之前的实验中， 我们的操作只有一条时间线， 在 Git 里， 这个分支叫主分支，即 master 分支。HEAD 严格来说不是指向提交，而是指向 master，master 才是指向提交的，所以，HEAD 指向的就是当前分支。下面我们一起来实践一下如何创建和合并分支。

## 实验过程

1 、实现分支的创建、合并

1. 进入本地仓库 testgit 目录中，然后 git checkout –b 分支名，切换到 dev 分支上。

\*\* git checkout 命令加上 –b 参数表示创建并切换，相当于 2 条命令：

git branch dev

git checkout dev

（2） 用 git branch 命令查看自己当前目录下所有的分支，此时可查看到自己创

建的分支，当前分支前面会添加一个“\*” ：

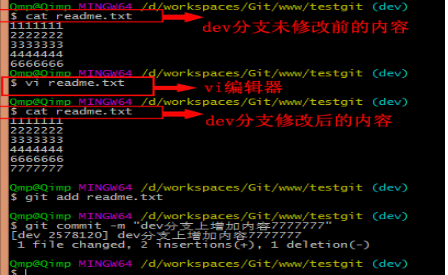
1. 用 cat 来查看下 readme.txt 内容，接着添加内容7777777

（4） 用 git checkout 切换到主分支 master上，继续查看 readme.txt

（5）git merge命令用于合并指定分支到当前分支上合并后， 再查看readme.txt

内容，和 dev 分支最新提交的是完全一样的。

1. 删除 dev 分支了



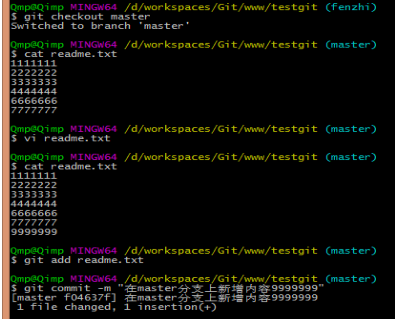
2 、解决分支冲突

1. 新建一个新分支readme.txt 添加一行内容 8888888，然后提交
2. 接着，我们现在切换到 master 分支上来，也在最后一行添加内容

（3）现在我们需要在 master 分支上来合并 fenzhi1，Git 用 <<<<<<< ， ======= ， >>>>>>>标记出不同分支的内容，其中 <<<HEAD是指主分支修改的内容，i >>>>>fenzhi 是指 fenzhi 上修改的内容。

（4）修改冲突部分内容，重新提交（add commit）,最后可用 git log 查看

分支合并情况。



3 、 分支管理策略

1. 创建一个 dev 分支。

2. 修改 readme.txt 内容。

3. 添加到暂存区。

4. 切换回主分支(master)。

5. 合并 dev 分支，使用命令 git merge --no-ff -m “注释” dev。

6. 查看历史记录。

4 、将新分支推送到 b github 以及删除 gitb hub 上的分支推送主分支：

git init

git add a.txt

git commit -m "first commit"

git remote add origin https://github.com/chenxhjeo/learngit.git

git push -u origin master

在本地新建一个分支： git branch dev

切换到你的新分支: git checkout dev

将新分支发布在 github 上： git push origin dev

在本地删除一个分支： git branch -d dev

在 github 远程端删除一个分支： git push origin :dev (分支名前的冒号代表

删除)

5 、 从远程的分支获取最新的版本

Git 中从远程的分支获取最新的版本到本地有这样 2 个命令：

1）git fetch origin master:tmp

git diff tmp

git merge tmp

2）git pull：相当于是从远程获取最新版本并 merge 到本地

git pull origin master

## 实验总结

1 、 总结创建与合并分支命令如下：

查看分支：git branch

创建分支：git branch 分支名

切换分支：git checkout 分支名

创建+ 切换分支：git checkout –b 分支名

合并某分支到当前分支：git merge 分支名

删除分支：git branch –d 分支名

2 、 除上述的 直接 合并方式 外 外，还有 2 2 中合并方式：

压力合并 ：git checkout master

git merge --squash dev

此时，dev 上的所有提交已经合并到当前工作区并暂存，但还没有作

为一个提交，可以像其他提交一样，把这个改动提交到版本库中：

git commit –m “something from dev”

拣选合并 ：比如在 dev 上的某个提交叫：321d76f，把它合并到 master 中：

git checkout master

git cherry-pick 321d76f

要拣选多个提交，可以给 git cherry-pick 命令传递-n 选项，比如：

git cherry-pick –n 321d76f