创建新分支

$ git checkout -b dev

Switched to a new branch 'dev'

git checkout命令加上-b参数表示创建并切换，相当于以下两条命令：

$ git branch dev

$ git checkout dev

Switched to branch 'dev'

查看当前分支git branch

$ git branch

\* dev

master

git branch命令会列出所有分支，当前分支前面会标一个\*号。

然后给我的test.docx文件加上一行Creating a new branch is quick

然后提交到版本库

$ git add test.docx

$ git commit -m"branch test"

[dev 3bc93e3] branch test

1 file changed, 0 insertions(+), 0 deletions(-)

切换回master分支：

$ git checkout master

Switched to branch 'master'

再次点开test.docx文件，没有Creating a new branch is quick这一行

因为这是添加在dev分支中的，和master分支没有关系



合并分支

把dev分支的合并到master分支上 git merge

$ git merge dev

Updating c7b842e..3bc93e3

Fast-forward

test.docx | Bin 11701 -> 11787 bytes

1 file changed, 0 insertions(+), 0 deletions(-)

git merge命令用于合并指定分支到当前分支。合并后，再查看test.docx的内容，就可以看到，和dev分支的最新提交是完全一样的。

注意到上面的Fast-forward信息，Git告诉我们，这次合并是“快进模式”，也就是直接把master指向dev的当前提交，所以合并速度非常快。

当然，也不是每次合并都能Fast-forward

合并完成后，就可以放心地删除dev分支了：

删除分支;

$ git branch -d dev

Deleted branch dev (was 3bc93e3).

再次查看分支情况

$ git branch

\* master

只剩下master啦

添加一个新的分支feature1

$ git checkout -b feature1

Switched to a new branch 'feature1'



这种情况下，Git无法执行“快速合并”，只能试图把各自的修改合并起来，但这种合并就可能会有冲突

$ git merge feature1

warning: Cannot merge binary files: test.docx (HEAD vs. feature1)

Auto-merging test.docx

CONFLICT (content): Merge conflict in test.docx

Automatic merge failed; fix conflicts and then commit the result.

果然出现了冲突！

Git告诉我们，test.docx文件存在冲突，必须手动解决冲突后再提交。git status也可以告诉我们冲突的文件：

$ git status

On branch master

You have unmerged paths.

(fix conflicts and run "git commit")

(use "git merge --abort" to abort the merge)

Unmerged paths:

(use "git add <file>..." to mark resolution)

both modified: test.docx

no changes added to commit (use "git add" and/or "git commit -a")

然后就乖乖的把master的改过来吧，然后提交

$ git add test.docx

$ git commit -m"conflict fixed"

[master 88adefa] conflict fixed

现在，master分支和feature1分支变成了下图所示：



然后惯例删除分支feature1

$ git branch -d feature1

Deleted branch feature1 (was b7d4aea).