首先，我们创建dev分支，然后切换到dev分支：

admin@zebin MINGW64 /d/BlueTestTry/bluetooth (master)

$ git checkout -b dev

Switched to a new branch 'dev'

admin@zebin MINGW64 /d/BlueTestTry/bluetooth (dev)

$ git branch

\* dev

master

和远程仓库建立连接。

admin@zebin MINGW64 /d/BlueTestTry/bluetooth (dev)

$ git push

fatal: The current branch dev has no upstream branch.

To push the current branch and set the remote as upstream, use

git push --set-upstream origin dev

admin@zebin MINGW64 /d/BlueTestTry/bluetooth (dev)

$ git push --set-upstream origin dev

Total 0 (delta 0), reused 0 (delta 0)

remote:

remote: Create a pull request for 'dev' on GitHub by visiting:

remote: https://github.com/ZebinGao/bluetooth/pull/new/dev

remote:

To github.com:ZebinGao/bluetooth.git

\* [new branch] dev -> dev

Branch 'dev' set up to track remote branch 'dev' from 'origin'.

然后，我们就可以在dev分支上正常提交，比如对readme.txt做个修改，加上一行，然后提交。现在，dev分支的工作完成，我们就可以切换回master分支

admin@zebin MINGW64 /d/BlueTestTry/bluetooth (dev)

$ git checkout master

Switched to branch 'master'

Your branch is up to date with 'origin/master'.

admin@zebin MINGW64 /d/BlueTestTry/bluetooth (master)

$

git merge命令用于合并指定分支到当前分支。合并后，再查看readme.txt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，我们后面会讲其他方式的合并。

合并完成后，就可以放心地删除dev分支了：

admin@zebin MINGW64 /d/BlueTestTry/bluetooth (master)

$ git merge dev

Updating 2ee2e28..86b33dd

Fast-forward

readme.txt | 3 ++-

1 file changed, 2 insertions(+), 1 deletion(-)

admin@zebin MINGW64 /d/BlueTestTry/bluetooth (master)

$ git branch -d dev

warning: not deleting branch 'dev' that is not yet merged to

'refs/remotes/origin/dev', even though it is merged to HEAD.

error: The branch 'dev' is not fully merged.

If you are sure you want to delete it, run 'git branch -D dev'.

admin@zebin MINGW64 /d/BlueTestTry/bluetooth (master)

$ git branch -D dev

Deleted branch dev (was 86b33dd).