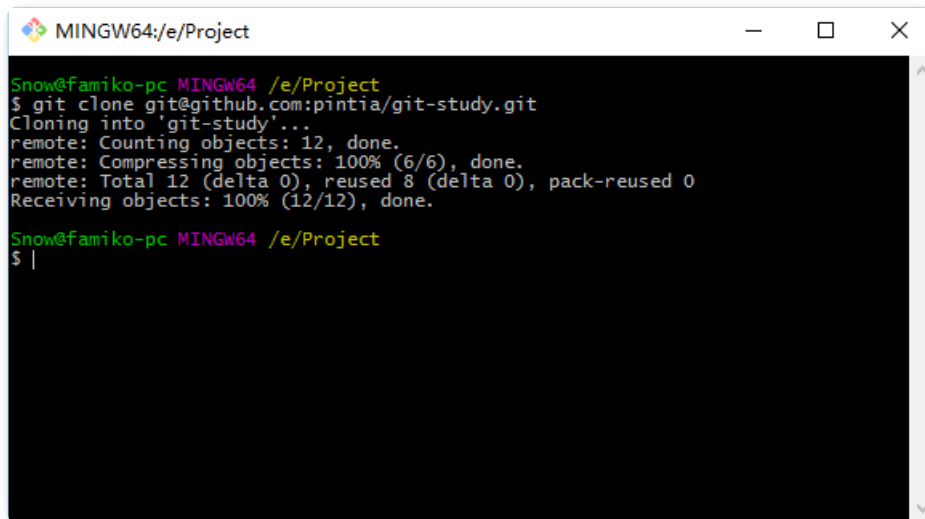


1 从远程仓库克隆项目到本地

```
git clone git@github.com:pintia/git-study.git
```

A terminal window titled 'MINGW64:/e/Project' showing the execution of 'git clone git@github.com:pintia/git-study.git'. The output shows the cloning process: 'Cloning into 'git-study'...', 'remote: Counting objects: 12, done.', 'remote: Compressing objects: 100% (6/6), done.', 'remote: Total 12 (delta 0), reused 8 (delta 0), pack-reused 0', and 'Receiving objects: 100% (12/12), done.' The prompt returns to 'Snow@famiko-pc MINGW64 /e/Project \$' with a cursor on the next line.

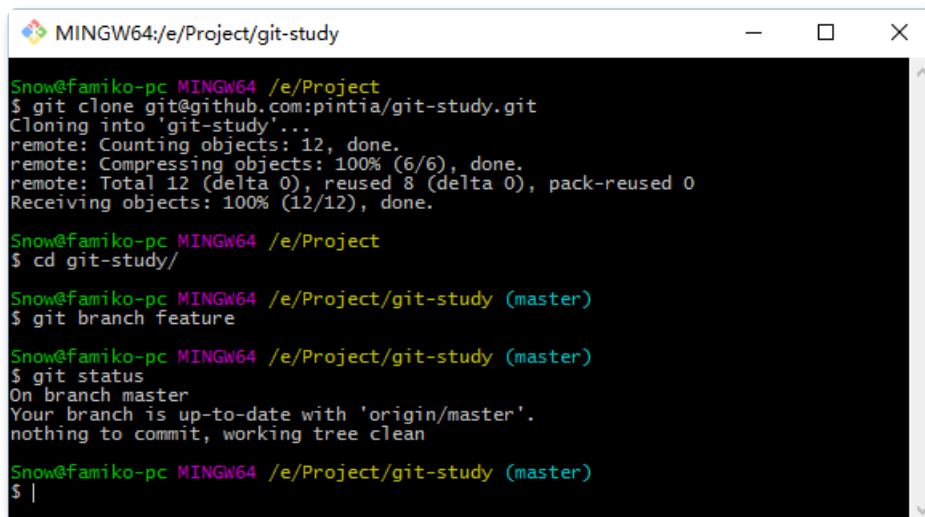
```
MINGW64:/e/Project
Snow@famiko-pc MINGW64 /e/Project
$ git clone git@github.com:pintia/git-study.git
Cloning into 'git-study'...
remote: Counting objects: 12, done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 12 (delta 0), reused 8 (delta 0), pack-reused 0
Receiving objects: 100% (12/12), done.

Snow@famiko-pc MINGW64 /e/Project
$ |
```

1. PNG

2 创建新的分支开发新功能

```
git branch feature
git status
```

A terminal window titled 'MINGW64:/e/Project/git-study' showing the execution of 'git branch feature' and 'git status'. The output of 'git status' shows 'On branch master' and 'Your branch is up-to-date with 'origin/master'. nothing to commit, working tree clean'. The prompt returns to 'Snow@famiko-pc MINGW64 /e/Project/git-study (master) \$' with a cursor on the next line.

```
MINGW64:/e/Project/git-study
Snow@famiko-pc MINGW64 /e/Project
$ git clone git@github.com:pintia/git-study.git
Cloning into 'git-study'...
remote: Counting objects: 12, done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 12 (delta 0), reused 8 (delta 0), pack-reused 0
Receiving objects: 100% (12/12), done.

Snow@famiko-pc MINGW64 /e/Project
$ cd git-study/

Snow@famiko-pc MINGW64 /e/Project/git-study (master)
$ git branch feature

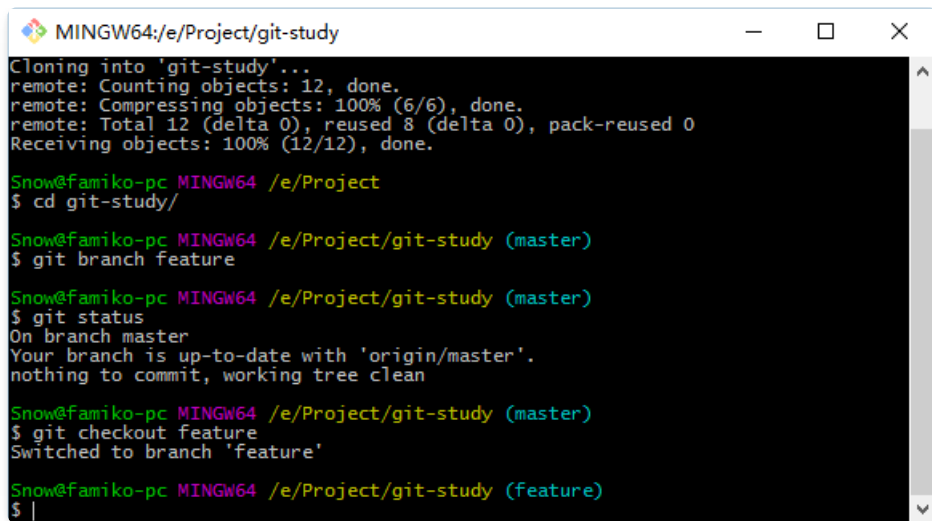
Snow@famiko-pc MINGW64 /e/Project/git-study (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working tree clean

Snow@famiko-pc MINGW64 /e/Project/git-study (master)
$ |
```

2. PNG

3 切换到新分支，开发新功能

```
git checkout feature
```

A terminal window titled 'MINGW64:/e/Project/git-study' showing the process of cloning a repository, checking out the master branch, creating a new branch named 'feature', and switching to it. The output shows successful cloning, followed by 'cd git-study/', 'git branch feature', 'git status' (showing the branch is up-to-date), and 'git checkout feature' (switching to the 'feature' branch).

```
MINGW64:/e/Project/git-study
Cloning into 'git-study'...
remote: Counting objects: 12, done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 12 (delta 0), reused 8 (delta 0), pack-reused 0
Receiving objects: 100% (12/12), done.

Snow@famiko-pc MINGW64 /e/Project
$ cd git-study/

Snow@famiko-pc MINGW64 /e/Project/git-study (master)
$ git branch feature

Snow@famiko-pc MINGW64 /e/Project/git-study (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working tree clean

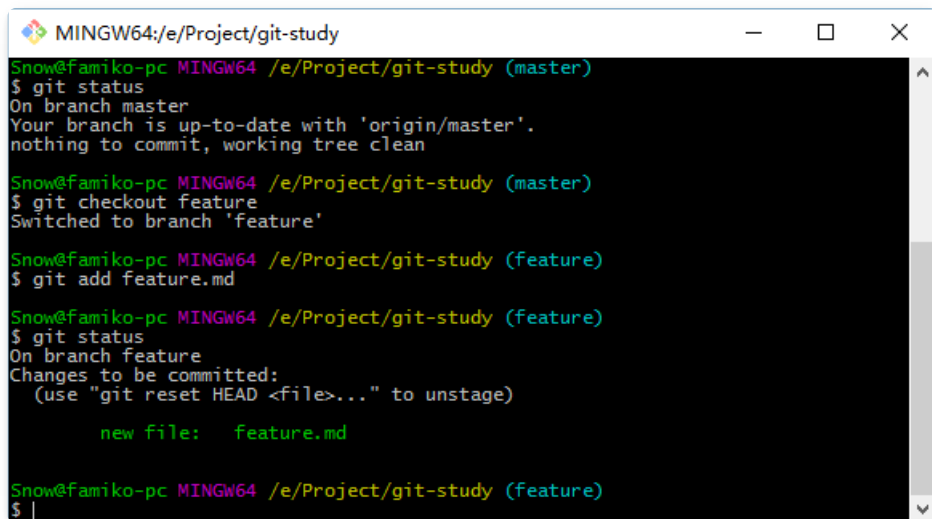
Snow@famiko-pc MINGW64 /e/Project/git-study (master)
$ git checkout feature
Switched to branch 'feature'

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$
```

3. PNG

- 4 新建了feature.md文件，将其添加跟踪，保存到暂存区

```
git add feature.md
git status
```

A terminal window titled 'MINGW64:/e/Project/git-study' showing the process of adding a new file to the 'feature' branch. The output shows 'git status' (nothing to commit), 'git checkout feature' (switching to 'feature'), 'git add feature.md' (staging the new file), and 'git status' (showing 'new file: feature.md' ready to be committed).

```
MINGW64:/e/Project/git-study

Snow@famiko-pc MINGW64 /e/Project/git-study (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working tree clean

Snow@famiko-pc MINGW64 /e/Project/git-study (master)
$ git checkout feature
Switched to branch 'feature'

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git add feature.md

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git status
On branch feature
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

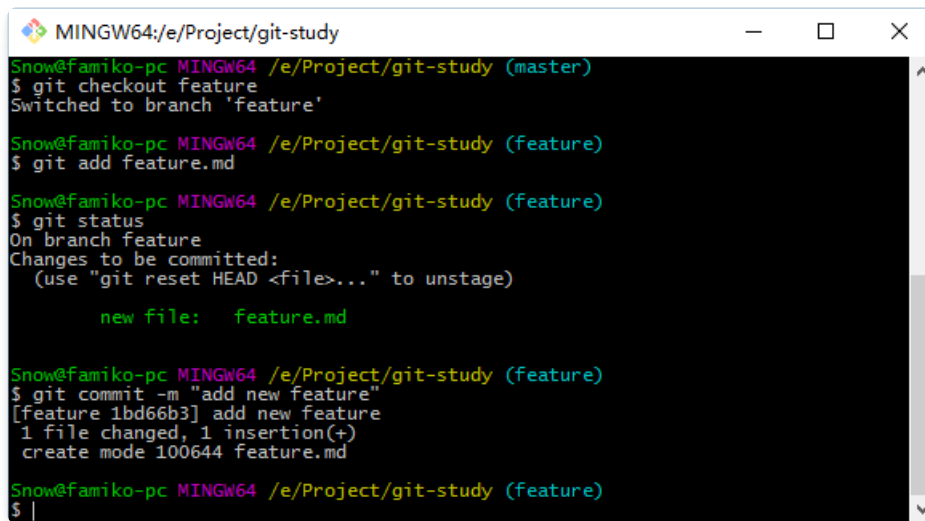
        new file:   feature.md

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$
```

4. PNG

- 5 将暂存区的文件进行一次提交

```
git commit -m "add new feature"
```



```
MINGW64:/e/Project/git-study
Snow@famiko-pc MINGW64 /e/Project/git-study (master)
$ git checkout feature
Switched to branch 'feature'

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git add feature.md

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git status
On branch feature
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        new file:   feature.md

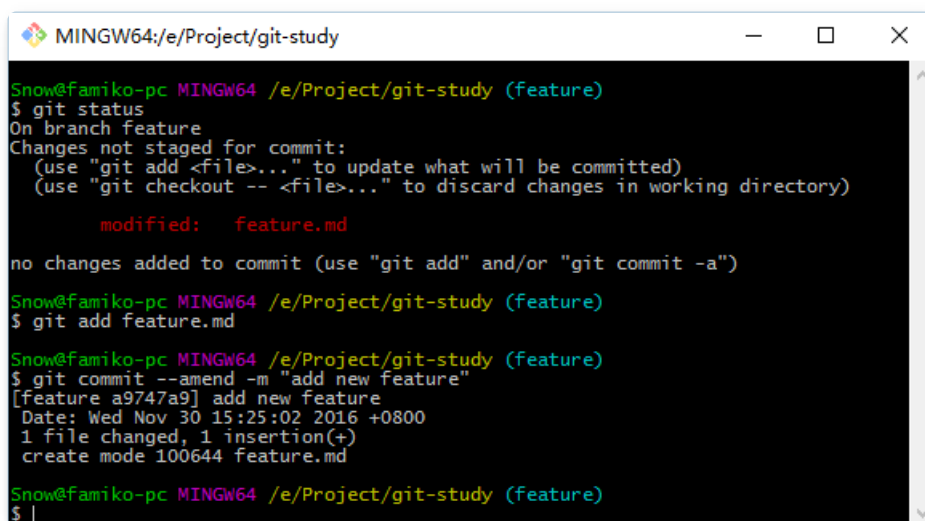
Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git commit -m "add new feature"
[feature 1bd66b3] add new feature
 1 file changed, 1 insertion(+)
 create mode 100644 feature.md

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ |
```

5. PNG

6 由于之前有bug，修改了bug，将它覆盖上一次提交。

```
git status
git add feature.md
git commit --amend -m "add new feature"
```



```
MINGW64:/e/Project/git-study
Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git status
On branch feature
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   feature.md

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

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git add feature.md

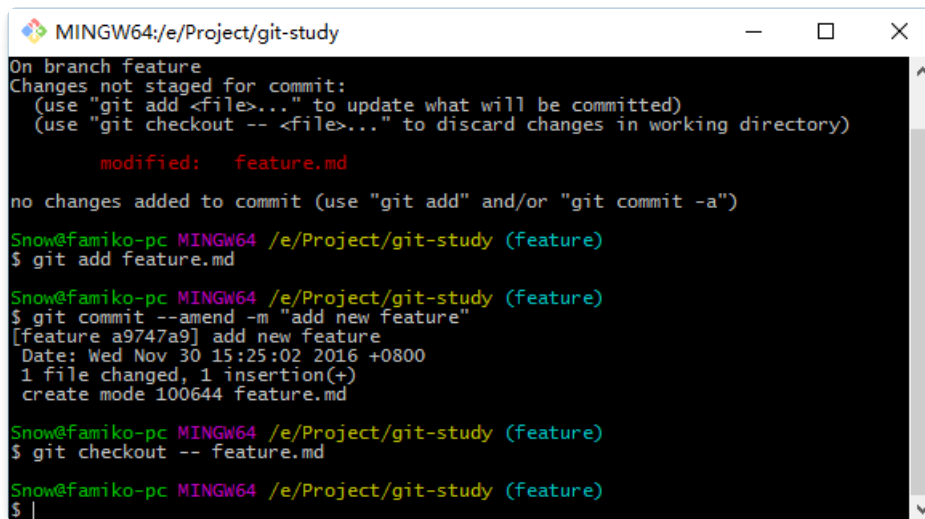
Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git commit --amend -m "add new feature"
[feature a9747a9] add new feature
 Date: Wed Nov 30 15:25:02 2016 +0800
 1 file changed, 1 insertion(+)
 create mode 100644 feature.md

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ |
```

6. PNG

7 误删了feature.md文件，从暂存区恢复

```
git checkout -- feature.md
```



```
MINGW64:/e/Project/git-study
On branch feature
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   feature.md

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

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git add feature.md

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git commit --amend -m "add new feature"
[feature a9747a9] add new feature
Date: Wed Nov 30 15:25:02 2016 +0800
1 file changed, 1 insertion(+)
create mode 100644 feature.md

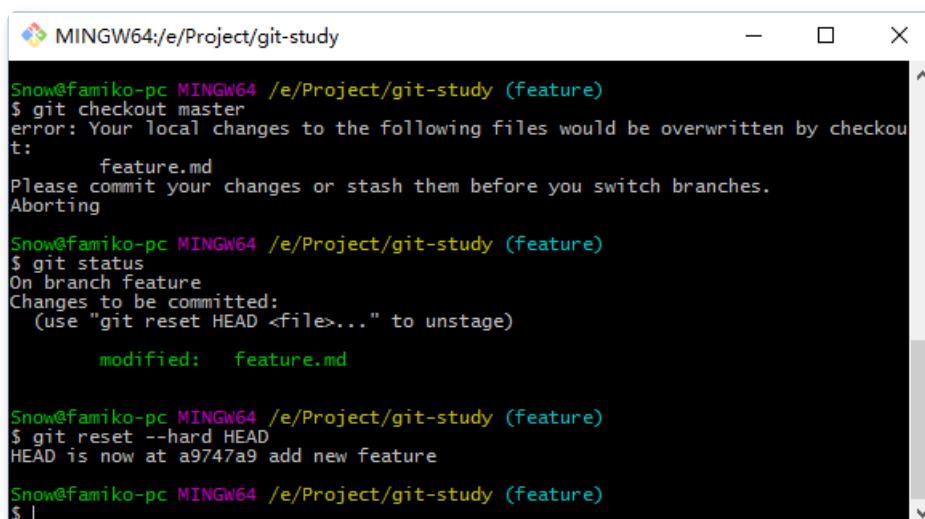
Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git checkout -- feature.md

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$
```

7. PNG

- 8 不想保留工作目录和暂存区的改变，回到当前分支最后一次提交的状态。

```
git reset --hard HEAD
```



```
MINGW64:/e/Project/git-study

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git checkout master
error: Your local changes to the following files would be overwritten by checkout:
       feature.md
Please commit your changes or stash them before you switch branches.
Aborting

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git status
On branch feature
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        modified:   feature.md

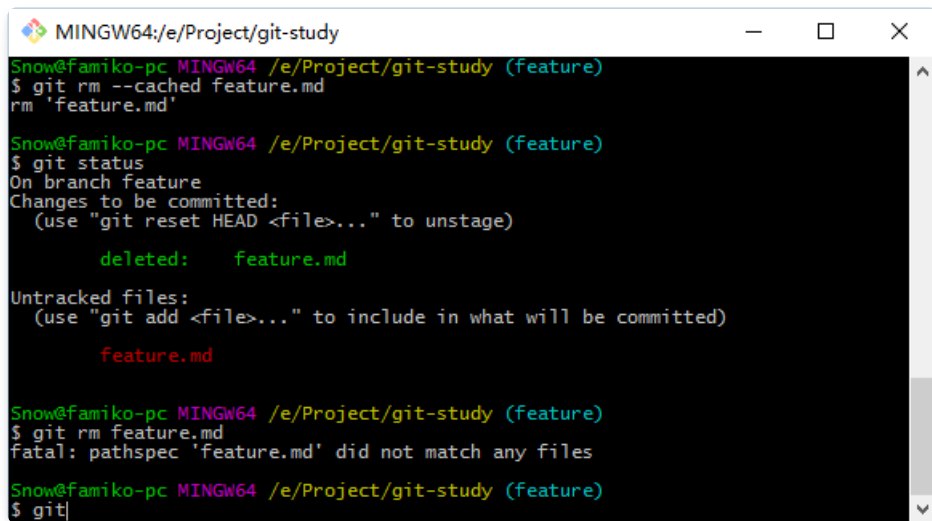
Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git reset --hard HEAD
HEAD is now at a9747a9 add new feature

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$
```

8. PNG

- 9 把feature.md从暂存区移除，但是保留在工作目录上，恢复到未追踪状态

```
git rm --cached feature.md
```

A terminal window titled 'MINGW64:/e/Project/git-study' showing the following commands and output:

```
Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git rm --cached feature.md
rm 'feature.md'

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git status
On branch feature
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        deleted:    feature.md

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        feature.md

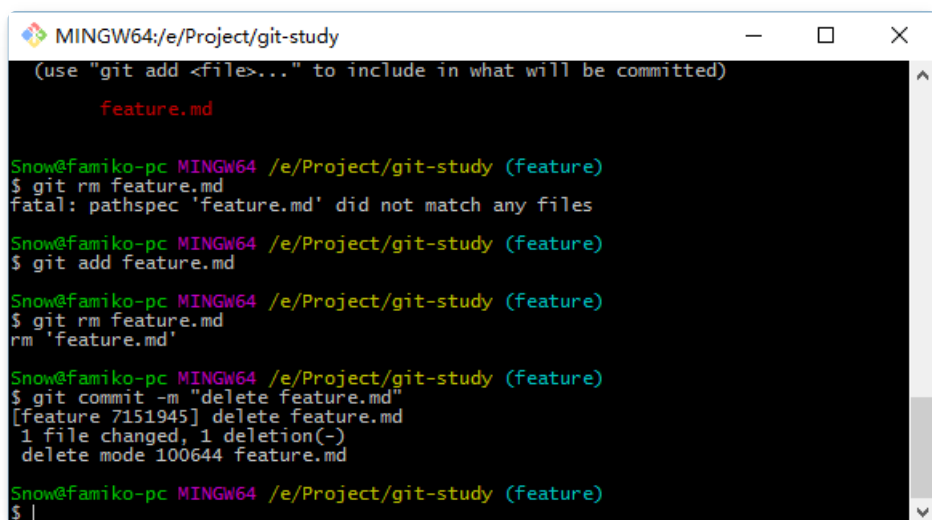
Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git rm feature.md
fatal: pathspec 'feature.md' did not match any files

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git
```

9. PNG

- 10 把feature.md从暂存区和工作目录中移除，并提交

```
git rm feature.md
git commit -m "delete feature.md"
```

A terminal window titled 'MINGW64:/e/Project/git-study' showing the following commands and output:

```
(use "git add <file>..." to include in what will be committed)

        feature.md

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git rm feature.md
fatal: pathspec 'feature.md' did not match any files

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git add feature.md

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git rm feature.md
rm 'feature.md'

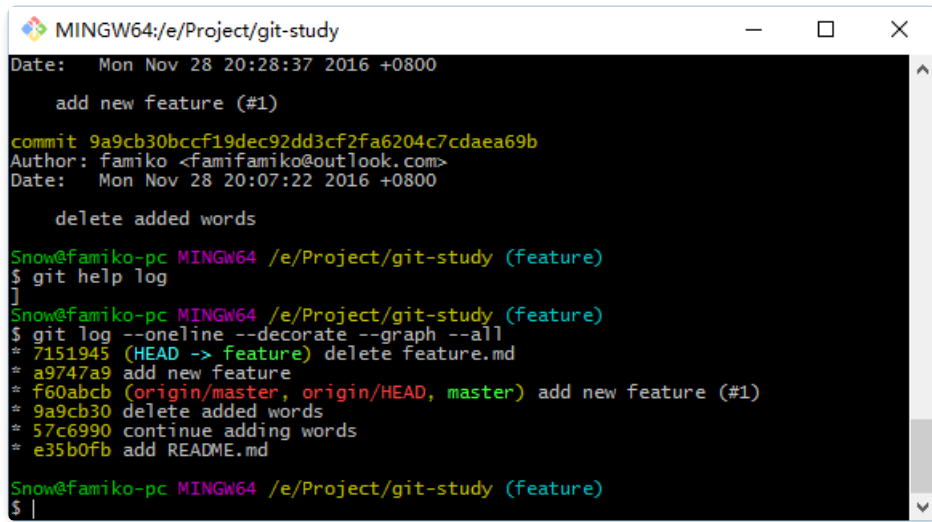
Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git commit -m "delete feature.md"
[feature 7151945] delete feature.md
1 file changed, 1 deletion(-)
delete mode 100644 feature.md

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$
```

10. PNG

- 11 查看commit历史，但是对于log指令参数不熟悉

```
git help log
git log --oneline --decorate --graph --all
```



```
MINGW64:/e/Project/git-study
Date: Mon Nov 28 20:28:37 2016 +0800

add new feature (#1)

commit 9a9cb30bccf19dec92dd3cf2fa6204c7cdaea69b
Author: famiko <famifamiko@outlook.com>
Date: Mon Nov 28 20:07:22 2016 +0800

delete added words

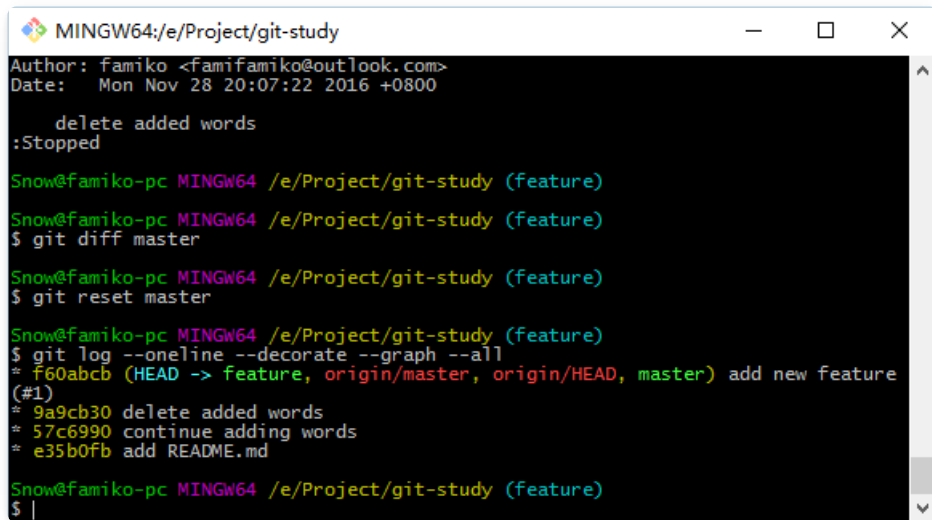
Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git help log
]
Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git log --oneline --decorate --graph --all
* 7151945 (HEAD -> feature) delete feature.md
* a9747a9 add new feature
* f60abcb (origin/master, origin/HEAD, master) add new feature (#1)
* 9a9cb30 delete added words
* 57c6990 continue adding words
* e35b0fb add README.md

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$
```

11. PNG

12 此时删除了feature.md文件，与master相同，回退到master分支重新开始

```
git diff master
git reset master
git log --oneline --decorate --graph --all
```



```
MINGW64:/e/Project/git-study
Author: famiko <famifamiko@outlook.com>
Date: Mon Nov 28 20:07:22 2016 +0800

delete added words
:Stopped

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git diff master

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git reset master

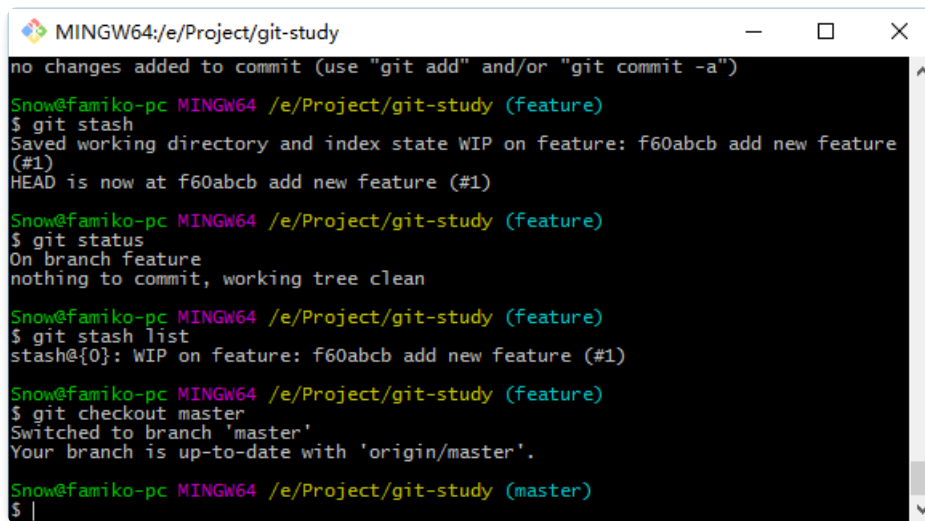
Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git log --oneline --decorate --graph --all
* f60abcb (HEAD -> feature, origin/master, origin/HEAD, master) add new feature (#1)
* 9a9cb30 delete added words
* 57c6990 continue adding words
* e35b0fb add README.md

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$
```

12. PNG

13 修改了README.md，此时想在保留目前工作状态下，查看master中README.md

```
git stash
git status
git stash list
git checkout master
```

A terminal window titled 'MINGW64:/e/Project/git-study' showing a sequence of git commands. The user starts on the 'feature' branch, stashes the current state, checks the status, lists the stash, and then checks out the 'master' branch.

```
MINGW64:/e/Project/git-study
no changes added to commit (use "git add" and/or "git commit -a")

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git stash
Saved working directory and index state WIP on feature: f60abcb add new feature (#1)
HEAD is now at f60abcb add new feature (#1)

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git status
On branch feature
nothing to commit, working tree clean

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git stash list
stash@{0}: WIP on feature: f60abcb add new feature (#1)

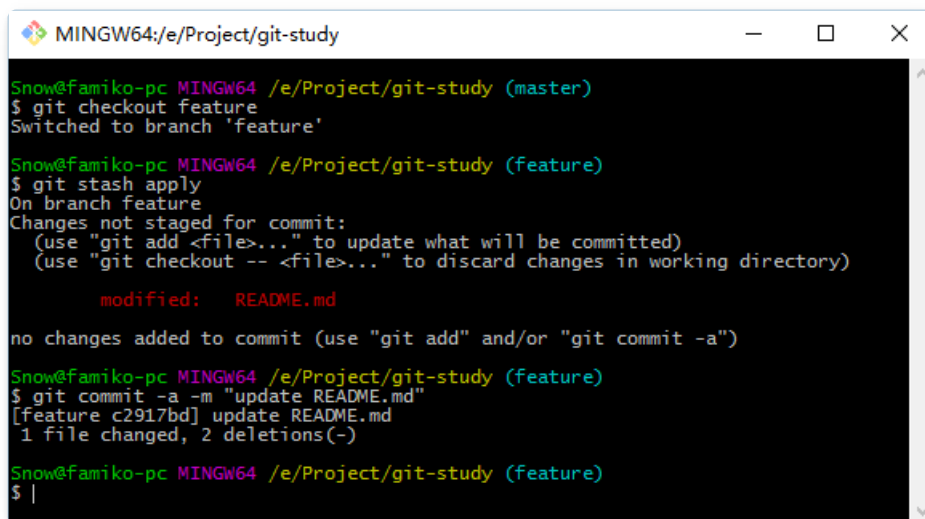
Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git checkout master
Switched to branch 'master'
Your branch is up-to-date with 'origin/master'.

Snow@famiko-pc MINGW64 /e/Project/git-study (master)
$
```

13. PNG

14 恢复到之前的状态，提交

```
git checkout feature
git stash apply
git commit -a -m "update README.md"
```

A terminal window titled 'MINGW64:/e/Project/git-study' showing the user switching back to the 'feature' branch, applying the stash, and committing the changes to README.md.

```
MINGW64:/e/Project/git-study

Snow@famiko-pc MINGW64 /e/Project/git-study (master)
$ git checkout feature
Switched to branch 'feature'

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git stash apply
On branch feature
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   README.md

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

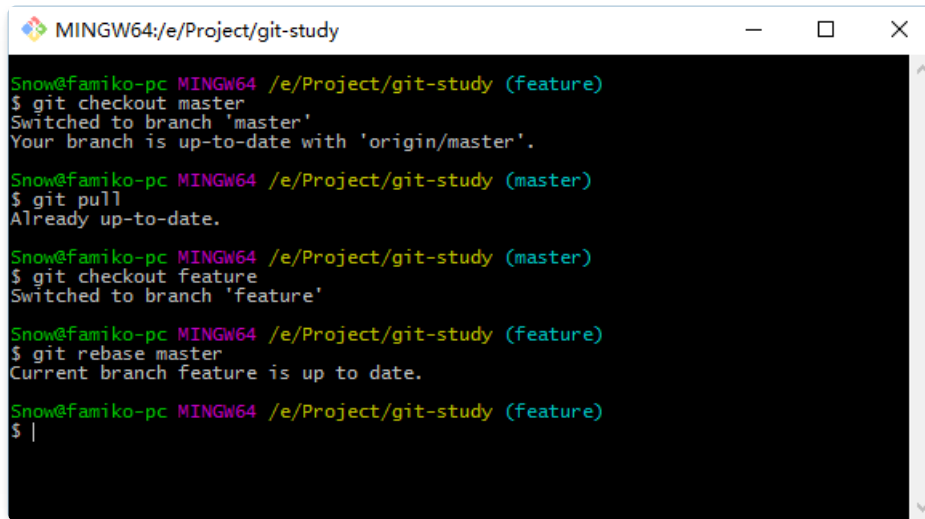
Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git commit -a -m "update README.md"
[feature c2917bd] update README.md
1 file changed, 2 deletions(-)

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$
```

14. PNG

15 第二天继续干活，先更新master

```
git checkout master
git pull
git checkout feature
git rebase master
```



```
MINGW64:/e/Project/git-study

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git checkout master
Switched to branch 'master'
Your branch is up-to-date with 'origin/master'.

Snow@famiko-pc MINGW64 /e/Project/git-study (master)
$ git pull
Already up-to-date.

Snow@famiko-pc MINGW64 /e/Project/git-study (master)
$ git checkout feature
Switched to branch 'feature'

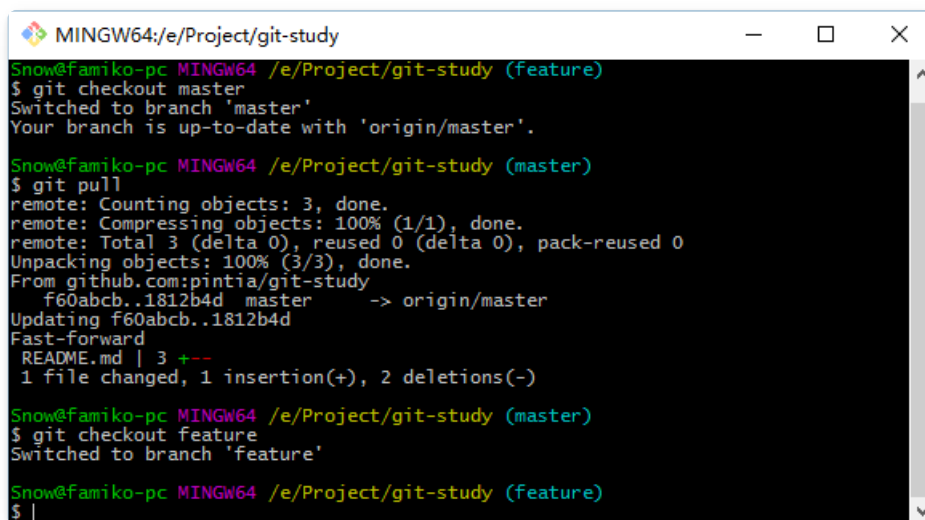
Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git rebase master
Current branch feature is up to date.

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ |
```

15. PNG

16 第三天继续干活，先更新master

```
git checkout master
git pull
git checkout feature
git rebase master
```



```
MINGW64:/e/Project/git-study

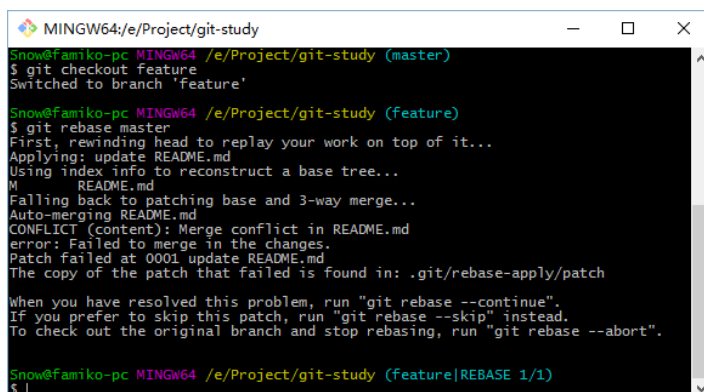
Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git checkout master
Switched to branch 'master'
Your branch is up-to-date with 'origin/master'.

Snow@famiko-pc MINGW64 /e/Project/git-study (master)
$ git pull
remote: Counting objects: 3, done.
remote: Compressing objects: 100% (1/1), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From github.com:pintia/git-study
   f60abcb..1812b4d master -> origin/master
Updating f60abcb..1812b4d
Fast-forward
 README.md | 3 +--
 1 file changed, 1 insertion(+), 2 deletions(-)

Snow@famiko-pc MINGW64 /e/Project/git-study (master)
$ git checkout feature
Switched to branch 'feature'

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ |
```

16-1. PNG



```
MINGW64:/e/Project/git-study

Snow@famiko-pc MINGW64 /e/Project/git-study (master)
$ git checkout feature
Switched to branch 'feature'

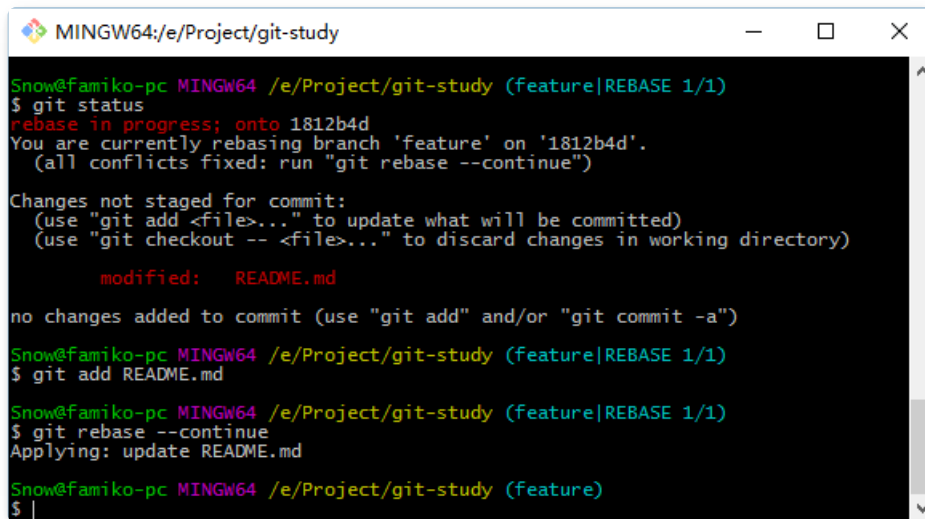
Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git rebase master
First, rewinding head to replay your work on top of it...
Applying: update README.md
Using index info to reconstruct a base tree...
M README.md
Falling back to patching base and 3-way merge...
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
error: Failed to merge in the changes.
Patch failed at 0001 update README.md
The copy of the patch that failed is found in: .git/rebase-apply/patch

When you have resolved this problem, run "git rebase --continue".
If you prefer to skip this patch, run "git rebase --skip" instead.
To check out the original branch and stop rebasing, run "git rebase --abort".

Snow@famiko-pc MINGW64 /e/Project/git-study (feature|REBASE 1/1)
$ |
```


17 出现冲突，手动修正冲突之后，继续合并

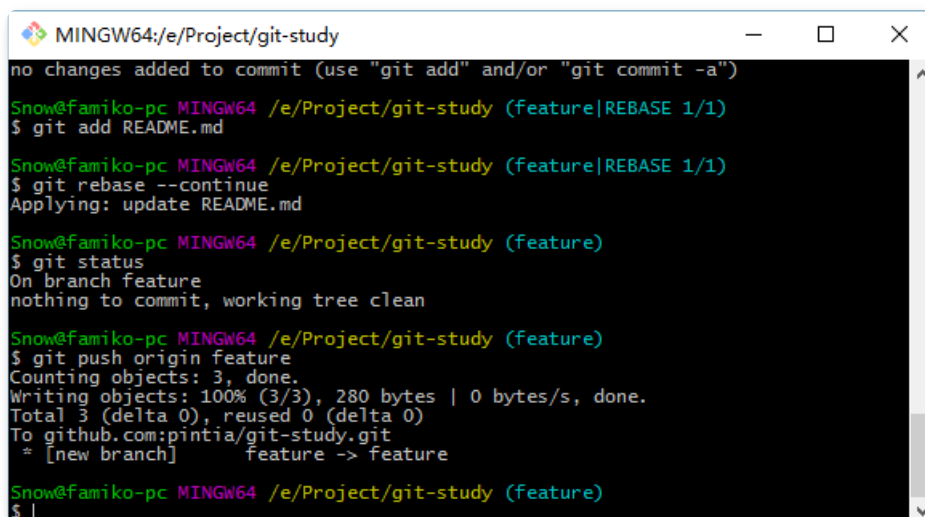
```
git status
git add README.md
git rebase --continue
```

A terminal window titled 'MINGW64:/e/Project/git-study' showing the output of a git rebase operation. The prompt is 'Snow@famiko-pc MINGW64 /e/Project/git-study (feature|REBASE 1/1)'. The user runs '\$ git status', which shows 'rebase in progress; onto 1812b4d' and a conflict in 'README.md'. The message says 'You are currently rebasing branch 'feature' on '1812b4d'. (all conflicts fixed: run "git rebase --continue")'. The user then runs '\$ git add README.md', which shows 'Changes not staged for commit: (use "git add <file>..." to update what will be committed) (use "git checkout -- <file>..." to discard changes in working directory) modified: README.md'. The user then runs '\$ git rebase --continue', which shows 'Applying: update README.md'. The prompt changes to 'Snow@famiko-pc MINGW64 /e/Project/git-study (feature)'.

17. PNG

18 完成新功能开发，推送到远程仓库对应分支下

```
git push origin feature
```

A terminal window titled 'MINGW64:/e/Project/git-study' showing the output of a git push operation. The prompt is 'Snow@famiko-pc MINGW64 /e/Project/git-study (feature)'. The user runs '\$ git status', which shows 'On branch feature' and 'nothing to commit, working tree clean'. The user then runs '\$ git push origin feature', which shows 'Counting objects: 3, done. Writing objects: 100% (3/3), 280 bytes | 0 bytes/s, done. Total 3 (delta 0), reused 0 (delta 0) To github.com:pintia/git-study.git * [new branch] feature -> feature'. The prompt changes to 'Snow@famiko-pc MINGW64 /e/Project/git-study (feature)'.

18. PNG

19 继续完善代码，推到远程仓库

```
git add feature.md
git commit -m "add feature.md"
git push origin feature
```

```
MINGW64:/e/Project/git-study

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git add feature.md

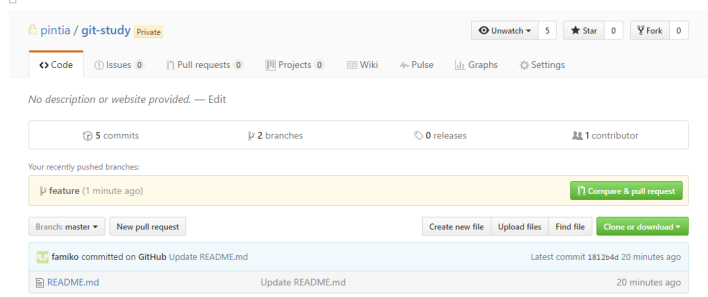
Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git commit -m "add feature.md"
[feature 898d7e4] add feature.md
1 file changed, 1 insertion(+)
create mode 100644 feature.md

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ git push origin feature
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 281 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To github.com:pintia/git-study.git
69860e7..898d7e4 feature -> feature

Snow@famiko-pc MINGW64 /e/Project/git-study (feature)
$ |
```

19. PNG

20 时机成熟，发起一个pull request



Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also compare across forks.

