

## SimpleUserOfGit

笔记本: 我的第一个笔记本

创建时间: 2021/1/6 17:11

更新时间: 2021/1/7 13:27

作者: 罗罗罗罗罗

URL: [https://mp.weixin.qq.com/s?\\_\\_biz=MjM5NzMyMjAwMA==&mid=2651492431&i...](https://mp.weixin.qq.com/s?__biz=MjM5NzMyMjAwMA==&mid=2651492431&i...)

### 克隆远端仓库代码

git clone <https://github.com/Sammie2020/HelloGitHub.git>

### 进去HelloGitHub目录, 配置开发者用户名和邮箱

git config user.name Sammie2020

git config user.email [Sammie2020@qq.com](mailto:Sammie2020@qq.com)

### 本地

#### 添加月份分支

git branch 202101V

#### 修改分支名称

git branch -m 202101V 202101B

#### 查看分支

git branch

#### 删除分支

git branch -d 202101B

#### 切换分支

git checkout 202101B

#### 查看文件变更状态

git status

```
Sammi@LAPTOP-0MB99MMD MINGW32 /d/git/Git/HelloGitHub (202101B)
$ git status
On branch 202101B
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   README.md

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

changes not staged for commit: (改动文件未提交到暂存区)

#### 添加文件变更到暂存区

git add README.md

#### 添加所有变更文件到暂存区

git add .

添加后用git status查看文件当前状态, changes to be committed (文件已提交到暂存区)

```
Sammi@LAPTOP-OMB99MMD MINGW32 /d/git/Git/HelloGitHub (202101B)
$ git status
On branch 202101B
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   README.md

Sammi@LAPTOP-OMB99MMD MINGW32 /d/git/Git/HelloGitHub (202101B)
$
```

提交文件变更到版本库

`git commit -m 'modify readme.md (这里写提交的原因)'`

添加后不想提交

`git reset`

将本地代码推送到服务器

`git push origin 202101B`

origin 当前git的服务器地址

```
Sammi@LAPTOP-OMB99MMD MINGW32 /d/git/Git/HelloGitHub (202101B)
$ git push origin 202101B
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 428 bytes | 428.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for '202101B' on GitHub by visiting:
remote:   https://github.com/Sammie2020/HelloGitHub/pull/new/202101B
remote:
To https://github.com/Sammie2020/HelloGitHub.git
 * [new branch]      202101B -> 202101B
```

远程分支已删除，但是本地仓库`git branch -av`查看的时候还是显示，查看远程分支状态

`git remote show origin`

```

Sammi@LAPTOP-0MB99MMD MINGW32 /d/git/Git/HelloGitHub (202101B)
$ git branch -av
* 202101B          c1080fc modify README.MD
default           3c1db46 Update README.md
remotes/origin/123 faf2b1f Add a.txt
remotes/origin/202101B c1080fc modify README.MD
remotes/origin/HEAD -> origin/default
remotes/origin/ceshi01 cccfe1b Add ceshi2x.txt
remotes/origin/ceshi02 ec14461 add ceshi2.txt in ceshi02 branch
remotes/origin/default 3c1db46 Update README.md
remotes/origin/master  cccfe1b Add ceshi2x.txt
remotes/origin/readme  a11c185 modify file ces.txt

Sammi@LAPTOP-0MB99MMD MINGW32 /d/git/Git/HelloGitHub (202101B)
$ git remote show origin
* remote origin
Fetch URL: https://github.com/Sammie2020/HelloGitHub.git
Push URL:  https://github.com/Sammie2020/HelloGitHub.git
HEAD branch: default
Remote branches:
  202101B          tracked
  default          tracked
  refs/remotes/origin/123      stale (use 'git remote prune' to remove)
  refs/remotes/origin/ceshi01  stale (use 'git remote prune' to remove)
  refs/remotes/origin/ceshi02  stale (use 'git remote prune' to remove)
  refs/remotes/origin/master   stale (use 'git remote prune' to remove)
  refs/remotes/origin/readme   stale (use 'git remote prune' to remove)
Local branch configured for 'git pull':
  default merges with remote default
Local refs configured for 'git push':
  202101B pushes to 202101B (up to date)
  default pushes to default (up to date)

Sammi@LAPTOP-0MB99MMD MINGW32 /d/git/Git/HelloGitHub (202101B)

```

#### 删除远程分支已被删除的分支

`git remote prune origin`

```

Sammi@LAPTOP-0MB99MMD MINGW32 /d/git/Git/HelloGitHub (202101B)
$ git remote prune origin
Pruning origin
URL: https://github.com/Sammie2020/HelloGitHub.git
* [pruned] origin/123
* [pruned] origin/ceshi01
* [pruned] origin/ceshi02
* [pruned] origin/master
* [pruned] origin/readme

```

#### 拉取服务器最新的代码到本地

`git pull origin 202101B`

如果服务器代码做了变动，而你本地的代码也有变动，拉取的代码就有可能跟你本地的改动冲突，一般情况下 `Git` 会自动处理这种冲突合并，但如果改动的是同一行，那就需要手动来合并代码，编辑文件，保存最新的改动，再通过 `git add .` 和

`git commit -m 'XXX'` 来提交合并。

#### 查看版本提交记录（提交人/日期/提交原因）

`git log`

提交记录过多可以按J下翻，k上翻，Q退出查看

#### 设置哪些内容不需要推送到服务器，这是一个配置文件

`.gitignore`

#### 为项目标记里程碑

`git tag origin SimpleUseOfGit`

以上为git的简单使用总结

