**Git 与 GitHub**

**环境**

* GitHub 账号
* Git Bash

**Git 版本控制工作流程**

1. 设置用户名和邮箱，第一个命令查看设置信息，如已设置可以省略此步这里的用户名是 GitHub 的账号，邮箱是 GitHub 的绑定邮箱

git config --list

git config --global user.name ""

git config --global user.email ""

1. 在 GitHub 账号中创建一个新仓库
2. 克隆远程仓库到本地

git clone https://github.com/username/仓库名

1. 在本地仓库做一些增删改(可以把自己之前写的页面放进来)

touch a.txt

1. 查看仓库中文件的状态

git status git不能提交空文件夹

1. 将工作区的更改提交到暂存区

git add/rm file git add 01.css **git add . 提交所有文件**

1. 将暂存区的更改提交到本地仓库

git commit -m "message"

1. 将本地仓库的更改同步到远程仓库（即自己 GitHub 中）

git push (origin master) git push 01.css

1. push 时需要输入用户名和密码，先运行下面的命令，可以记录密码，以后 push 可以免密

git config --global credential.helper store

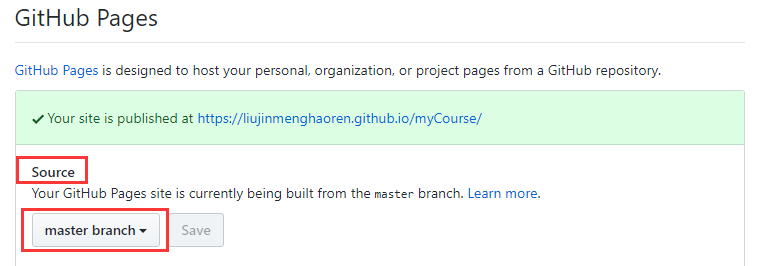
**团队合作**

1. 一个小组共用一个仓库，由团队中的某个成员创建；
2. 将组内所有人设置为本仓库的协作者，邀请方式为：settings —> Collaborators
3. 组员每次提交成果前，先拉取远程仓库最新的版本，再进行自己的操作

git pull (origin master)

**Git Pages**

设置 Git Pages 可以分配一个地址，直接显示网页的效果。操作如下：

1. 打开仓库
2. settings
3. Git Pages，将source 设置为 master branch[](https://github.com/liujinmenghaoren/myCourse/blob/master/web1/image/gitPages.png)

git rm -r文件夹名

git rm 文件名

git log 记录所有变更 git log -4 [近4次]

git mv 文件名 新名

git reset 五位哈希吗 –hard 重回版本

git reset 不被跟踪 绿->红

git commit -a -m "modify" 【add 和 commit】 或者-am

clear清空工作台

本地新建仓库

1. 新建文件夹

2. git init

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (master)

3. 文件夹内新建a.txt

4. git status 文件添加成功

5. git add . 此时查看状态Changes to be committed: 暂存区变化

6. git commit -m “modify”

nothing to commit, working tree clean 工作区和暂存区一致

1. git remote add origin 仓库链接 【远程创建仓库，同本地建立连接】
2. git pull origin master
3. git push -u origin master

分支

git branch 查看分支

git branch 分支名 添加分支

git branch -d 分支名 删除分支【确保合并之后，在主分支中删除】

git checkout 分支名 切换分支

git merge 子分支名 合并分支 【主分支中合并】

git checkout -b 分支名 【创建并切换】

例一：

新建文件夹

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest

$ git init

Initialized empty Git repository in

G:/github/branchTest/.git/

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (master)

新建文档

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (master)

$ git status

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (master)

$ git add .

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (master)

$ git commit -m "first"

[master (root-commit) ea32ba0] first

1 file changed, 1 insertion(+)

create mode 100644 test.txt

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (master)

$ git branch

\* master

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (master)

$ git branch develop

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (master)

$ git branch

develop

\* master

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (master)

$ git checkout develop

Switched to branch 'develop'

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (develop)

$ git branch

\* develop

master

修改文件

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (develop)

$ git status

On branch develop

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: test.txt

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

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (develop)

$ git commit -am "second"

[develop 9ce1ce2] second

1 file changed, 1 insertion(+), 1 deletion(-)

切换到主分支

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (develop)

$ git checkout master

Switched to branch 'master'

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (master)

$ git branch

develop

\* master

主分支仍是原文件

在master合分支文件

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (master)

$ git merge develop

Updating ea32ba0..9ce1ce2

Fast-forward

test.txt | 2 +-

1 file changed, 1 insertion(+), 1 deletion(-)

例二：

在主分支下【加一行ABC】

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (master)

$ git commit -am "add line ABC"

[master 5a88b3b] add line ABC

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

在develop分支下【加一行DEF】

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (master)

$ git checkout develop

Switched to branch 'develop'

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (develop)

$ git commit -am "add line DEF"

[develop c9d46de] add line DEF

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

合并分支出错

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (develop)

$ git checkout master

Switched to branch 'master'

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (master)

$ git merge develop

Auto-merging test.txt

CONFLICT (content): Merge conflict in test.txt

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

张三李四王五赵六

<<<<<<< HEAD

DEF

good

=======

ABC

bad

>>>>>>> develop

手动删除>>>>>> 和 ====== 内容

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (master|MERGING)

$ git commit -am "fix"

[master 5dbecc6] fix

qq@LAPTOP-UHH8IJBB MINGW64 /g/github/branchTest (master)

$ git branch -d "develop"

Deleted branch develop (was e4bf9d5).