## 1.安装git

Git的下载安装可以直接参考大佬的博客：

<https://www.liaoxuefeng.com/wiki/0013739516305929606dd18361248578c67b8067c8c017b000/00137396287703354d8c6c01c904c7d9ff056ae23da865a000>

## 2.使用前的准备

将git与GitHub账号绑定：

打开Git Bash，输入：

git config --global user.name “Your Name”

git config --global user,email [email@example.com](mailto:email@example.com)

表示这台机器上的所有git仓库都会使用这个用户名和邮件地址

## 3.第一次使用的操作

**创建版本库：**

用git init在当前文件夹创建管理文件（一个名为.git的文件夹），然后可以用git管理该目录下的所有文件

**添加远程仓库**

版本库创建后只属于本地的git上，要提交和获取远程git中的内容需要将本地库与远程库关联起来

由于本地Git仓库和GitHub仓库之间的传输是通过SSH加密的，所以要进行一下设置：

1.创建SSH Key

在命令行中输入：

ssh-keygen -t rsa -C [email@example.com](mailto:email@example.com)

可以在用户主目录下创建一个.ssh目录，这个目录下包含SSH的私钥（id\_rsa）和公钥（id\_rsa.pub）

注意每一个ssh文件只能加密一个远程仓库！

然后把id\_rsa.pub文件发给郭省吾，他会把公钥加入他的账户列表中，不然是无法进行提交的。

2.关联远程仓库

输入

git remote add origin https://github.com/HatsuneMK00/nianshi\_repository.git

就可以关联到项目的远程仓库(origin是默认的远程库的名字)

3.获取远程仓库的内容

git pull origin master

4.创建本地的dev分支

git checkout -b dev

## 4.日常操作

git pull origin master

获取远程库内容到本地，origin后加的是分支名

Git checkout master

改变当前正在操作的分支

**提交更改：**

Git add filename

把文件filename添加到暂存区，可以用.表示所有的文件

可以用Git status显示工作目录和暂存区的状态

Git commit -m “message”

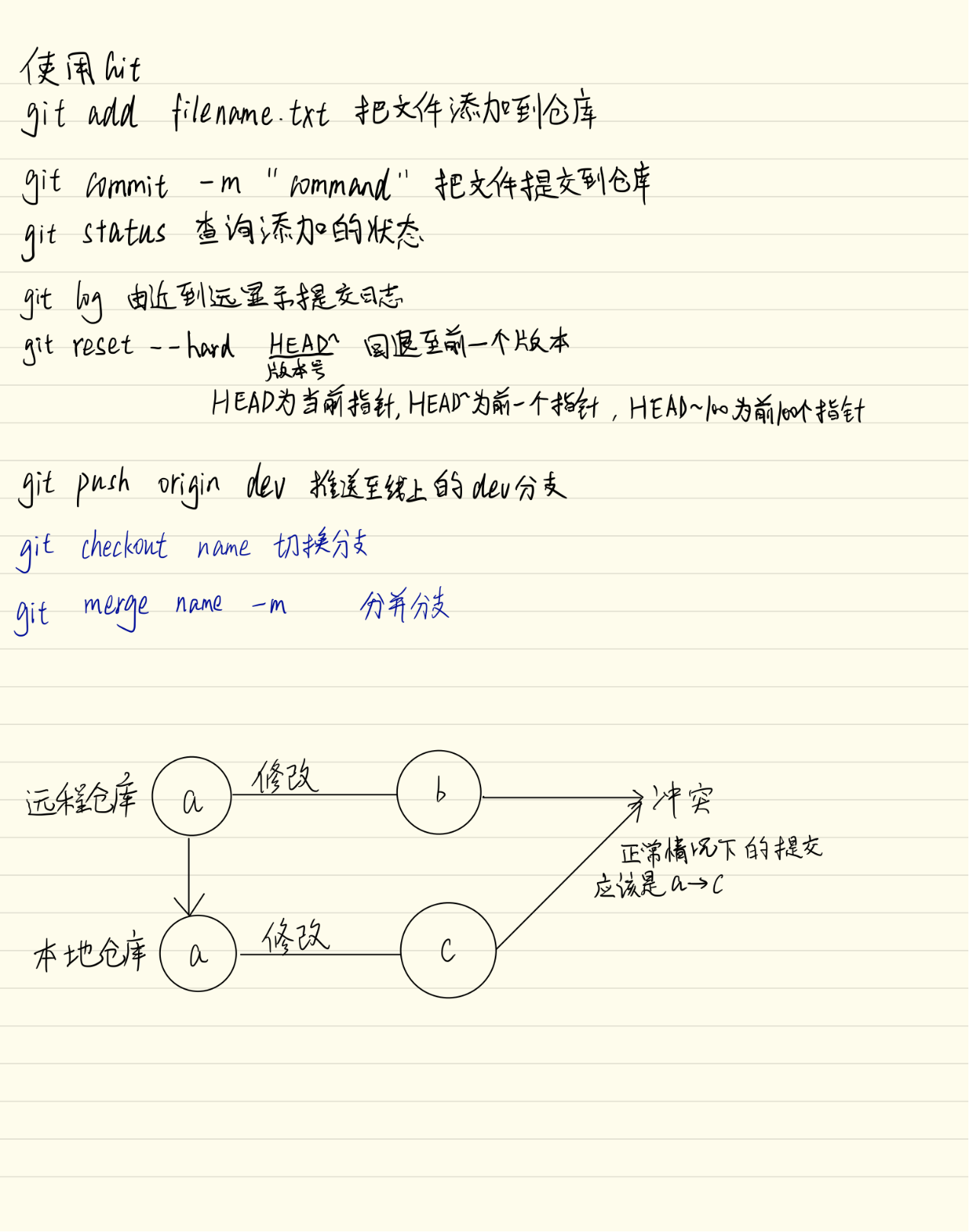
将所有add的文件提交到仓库(本地)，-m后跟的是对这次提交的说明。可以使用多次add后一次对所有add的文件进行提交

git push origin dev

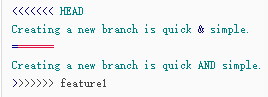
将本地修改内容推送至远程库,origin后同样是分支名

**冲突处理：**

当本地仓库与远程仓库的版本不一致时（例如你pull了远程仓库后，有人在你提交之前对远程仓库进行了修改），push操作会提示出现冲突，这时需要将不同的版本合并。



首先用pull获取当前的远程库，git会试图合并这两个版本，当发生冲突的文件不是同一文件时，可以自动对这些修改进行合并；当发生冲突的文件为同一个文件，git会标记出有差异的地方（如下图），然后我们要对这些地方进行手动的修改，修改之后要去掉标记



然后再进行提交操作，这个时候就可以成功提交