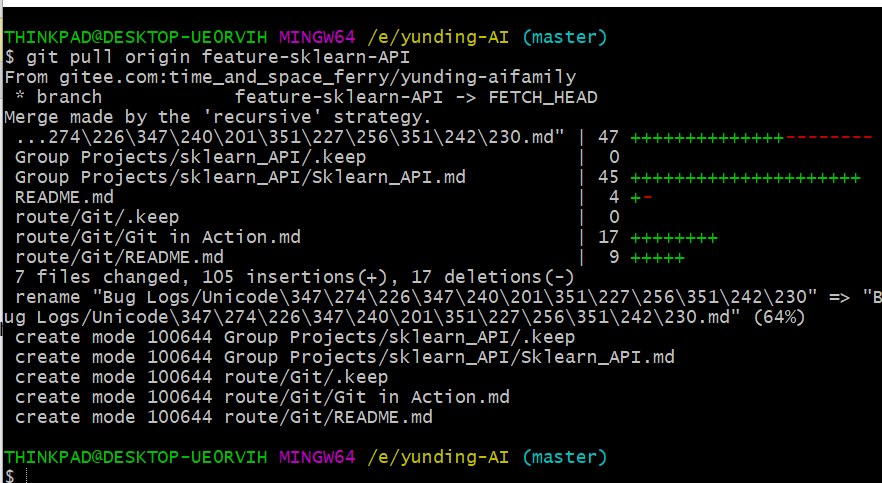
1. 在创建文件夹（根目录为英文，否则会有乱码）”yunding-AI”（名字任意），在文件夹中打开git bash
2. git init 初始化本地仓库
3. git remote add origin ssh\_address
4. git pull origin master 此时本地仓库内已经有了远程仓库的master分支下的所有内容

（3、4点可以通过git clone代替，git clone默认会把远程仓库的默认分支整个给clone下来，但只会在本地默认创建一个master分支；如果远程还有其他的分支，此时用git branch -a查看所有分支；但在咱们的仓库，默认分支改成了dev分支，所以Git clone实际拉取的是dev分支）



1. git pull origin feature-sklearn-API 此时默认为将远程仓库的feature-sklearn-API分支与本地的master分支合并



1. 合并过后发现，因为远程master分支比feature分支多一些东西，所以比较麻烦，这样在本地工作完后，推送分支会导致feature提前合并了一些master的东西，其实不影响，但不规范。
2. New 5. 第五步可以换一种方法，直接在本地创建 以 远程仓库的feature分支 为基线的 本地新分支 git checkout -b feature-sklearn-API origin/feature-sklearn-API

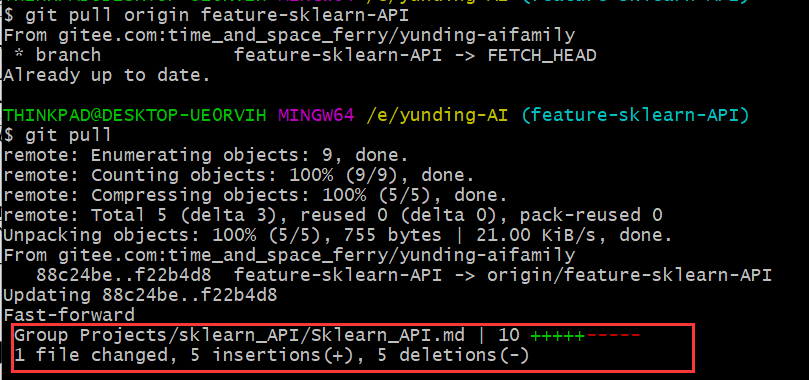
这样本地就有了远程的feature分支，并完成了自动切换。前期准备完成，现在可以开始在本地用Typora进行协作，然后推送了。

1. 通过Typora进行工作，工作后保存。
2. git status — git commit -am “add/updata/delete sklearn.xxx.xxx”
3. 开始推送，注意推送到对应分支，可以直接使用 git push或（git push 远程仓库名branch-name），因为之前拉取的时候已经完成了关联。

推送时要注意，如果推送失败，这说明远程仓库对应的分支比你目前的本地分支新，也就是存在冲突（这是因为在你New 5拉取远程分支后的这段时间里，你的同事老王也拉取了该分支，并先于你提交了他的工作，是不是很欠打）但日子还要过，你还指望着月底的奖金，于是你再次拉取远程分支，拉取后在本地文件内会出现“<<< >>>”诸如此类的冲突提示符，需要你手动对其进行修改（解决冲突），最后删去这些提示符。之后你重新推送。

1. 今日份工作已经完成，现在时间飞逝，10个小时过去了，第二天8点，你打开电脑，要做的第一件事就是拉取远程仓库的对应分支（因为在你睡觉的这段时间，说不定有卷王继续工作，你需要去合并他的内容到你的本地分支）

Git pull 或 git pull origin feature-sklearn-API



1. 注意。拉取分支时，会自动合并远程分支到本地分支，这就存在发生冲突的可能性，有可能卷王A只是在你的基础上进行了补充，这种情况不存在冲突，使用默认的合并策略即可。但如果卷王A觉得你写的都是垃圾，把你的内容全部删除，重写了一通，这是就存在冲突了，需要你做一些取舍。