**苦恼**

你有没有为了学git的经历一而再再而三查看别人的文档还是学不会的经历，只是它、因为你没有使用git的需要，更何况繁琐的命令行让人头疼，什么缓存区，分支，HEAD，合并分支这些让人头疼的东西敬而远之，有一款github官方的应用程序就是为了小白而生的，这篇教程里**不会介绍任何一个命令**但是看完之后你能轻松使用git。



**git介绍**

什么是git？间接的说git是分布式版本控制工具，这你或许看过很多遍了，但是我还是想在这里重复一遍，git可以在一下情境中很有用：

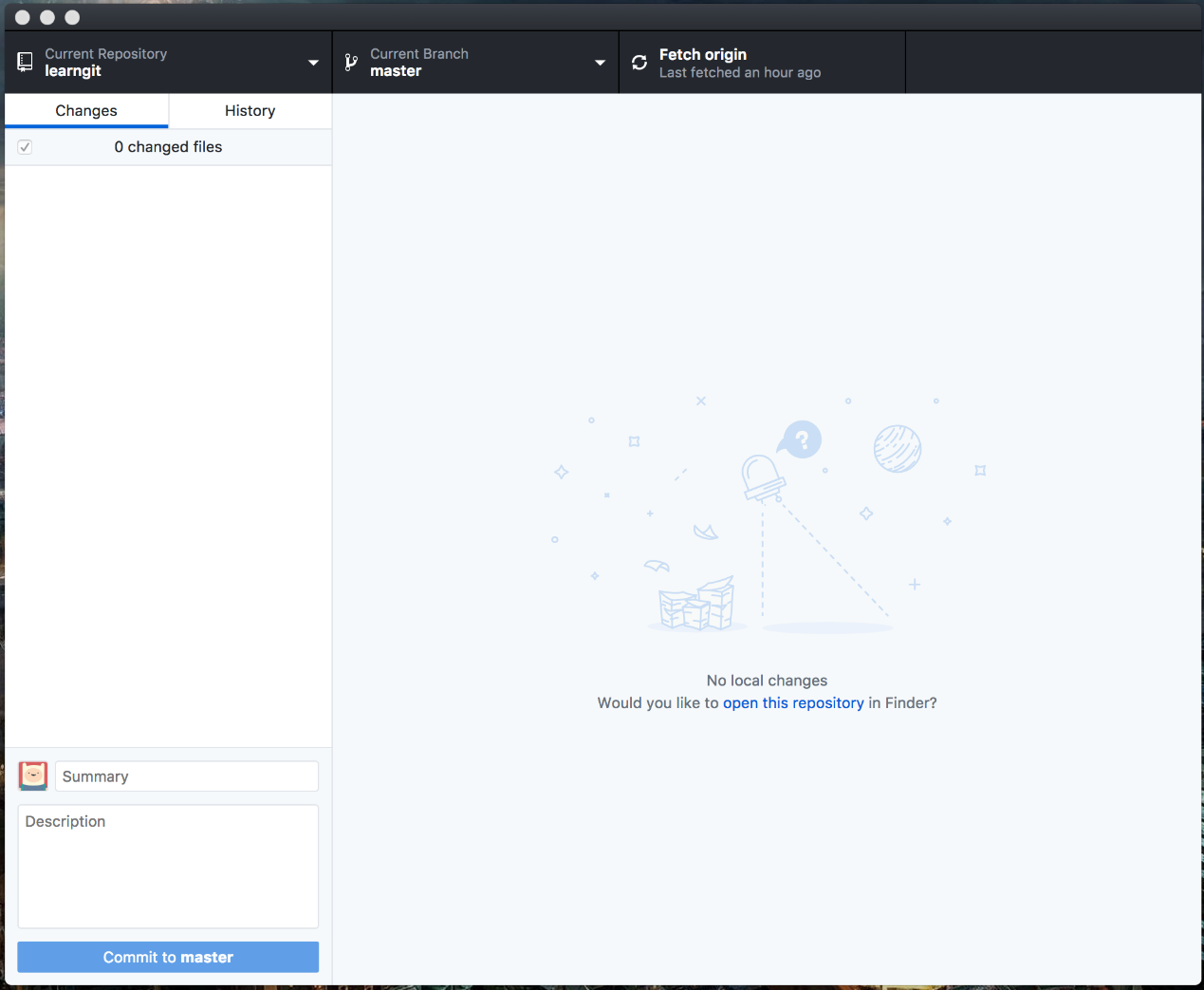
* 你在写一篇很长的论文，比如说论文的开头介绍， 相关的研究，论述，建议，总结。很显然这些都是不同部分，各个部分不相关联，但是总结起来就是你整个文章，但是论文嘛，总是改了又该，很烦人，因为会生成很多的文件。所以git可以帮你把不同的部分合并在一起但是不会生成任何多余的文件。**你可以迁出到任何你提交后的状态**。
* 其次就是软件开发，比如开发网页。有的人做前端，html，css，js调用一把梭，其他人很精通服务器，两者的业务分工明确，所以很适合分工合作，最后把两者的业务关联一下就可以看到网站了。这时候git也起到了相当重要的作用，因为是**分工合作**。

**Github Desktop**

这是[Github](https://link.jianshu.com?t=https%3A%2F%2Fdesktop.github.com" \t "_blank)的官方git软件，其实现在很多IDE都自带这种版本控制软件，学会了这个其他的版本控制软件上手就很简单了，这篇文章我只会分享自己常用的一些操作。

**界面介绍**

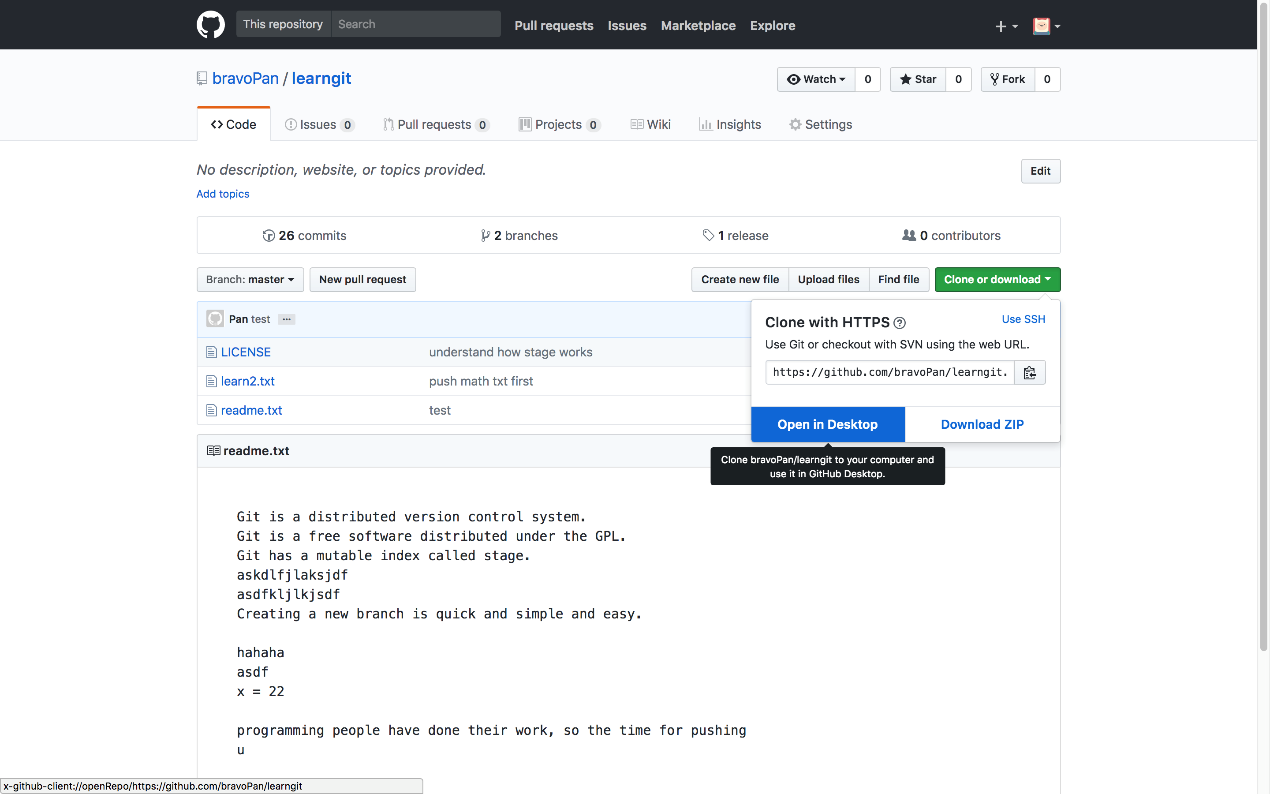
如果你打开这个软件后（下载地址在上方的超链接中），会发现应该如下所示。 左边的是可以切换添加进来的仓库，再也不需要cd来cd去了，白色框内是改变提醒，下面是提交修改。所以整个工作流程是有修改直接commit就行了。



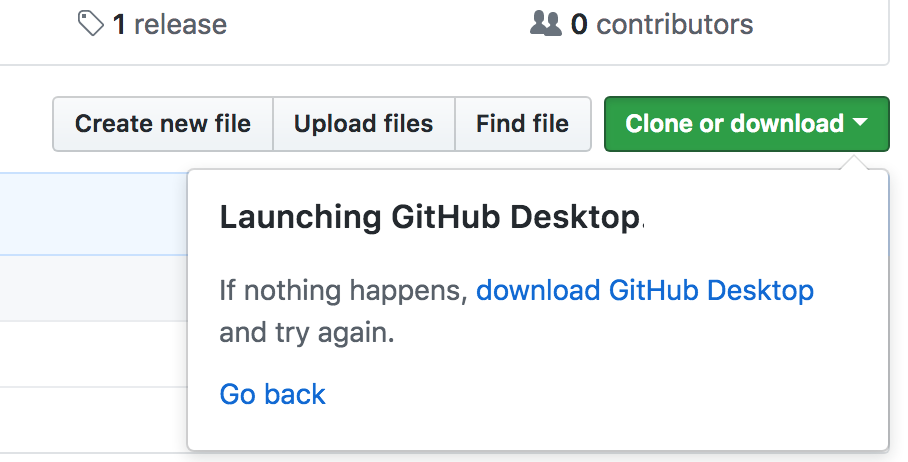
请注意你可以从左下角看到我的头像这是以为我已经安装了git，这篇文章侧重介绍Github Desktop的使用，软件的介绍而不是安装工具，所以如果你还没有安装git，请移步[廖老师的安装教程](https://link.jianshu.com?t=https%3A%2F%2Fwww.liaoxuefeng.com%2Fwiki%2F0013739516305929606dd18361248578c67b8067c8c017b000%2F00137396287703354d8c6c01c904c7d9ff056ae23da865a000" \t "_blank)

**创建一个本地仓库**

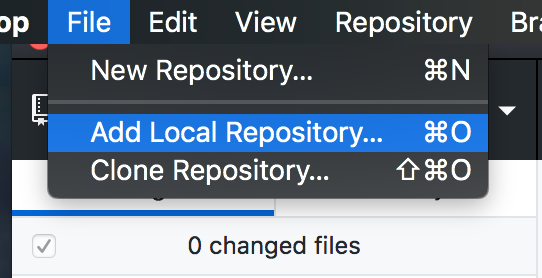
好了现在我们直接在我的[github](https://link.jianshu.com?t=https%3A%2F%2Fgithub.com%2FbravoPan%2Flearngit)上面克隆一个仓库并且在桌面打开，这个仓库已经初始化好了，所以不需要任何命令

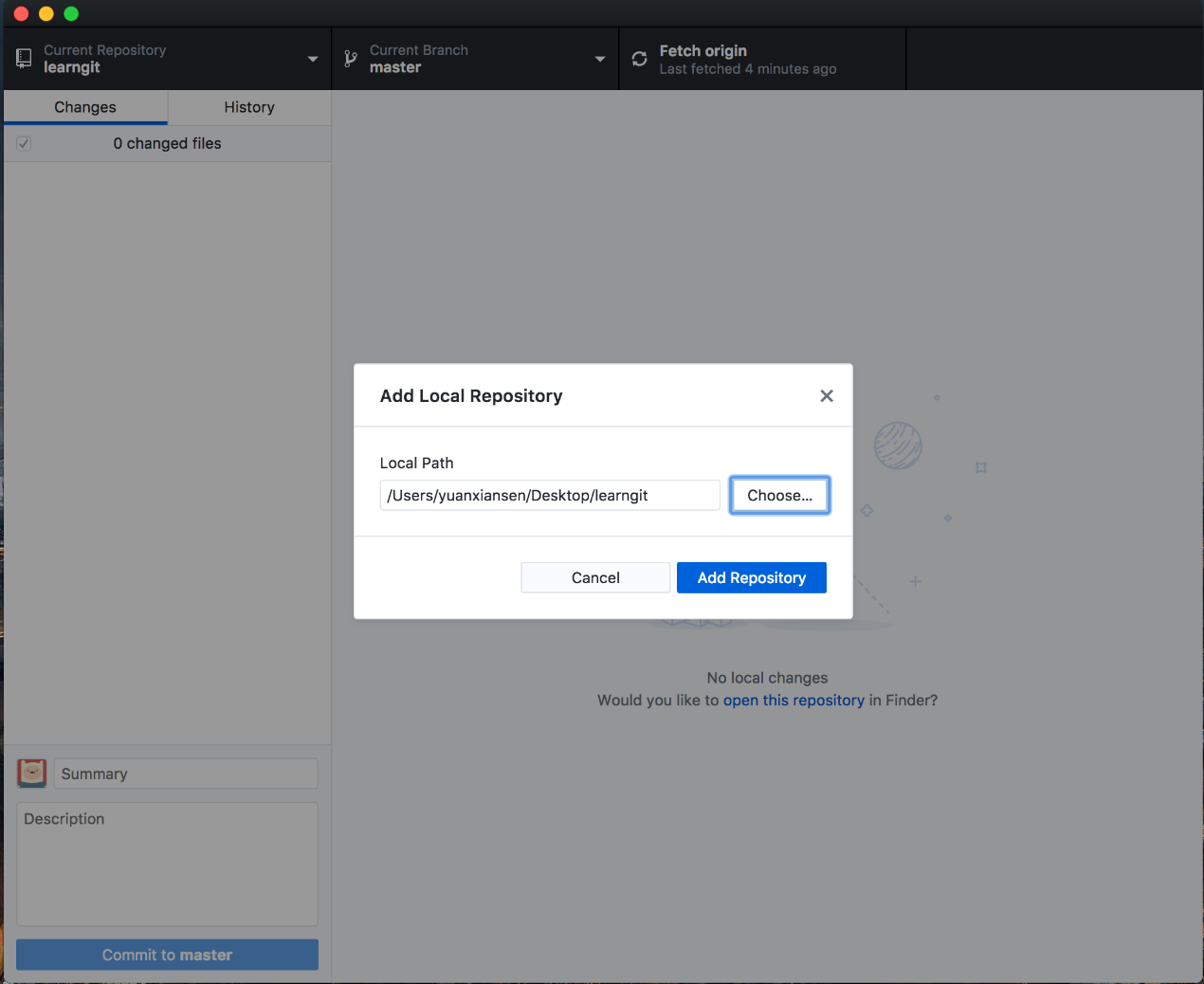


Github Desktop果然是亲儿子， 在网站下载会自动导入到Github Desktop中



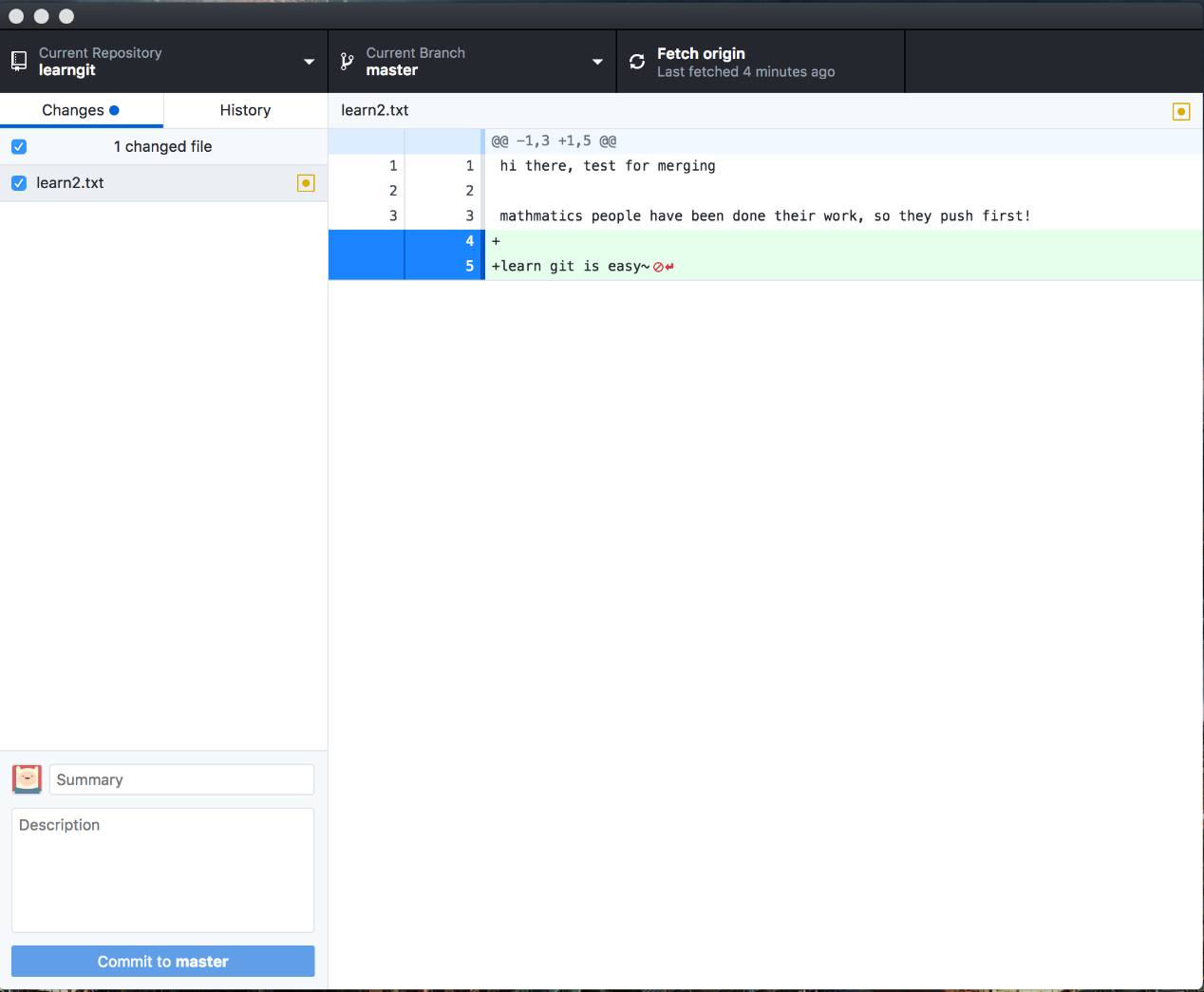
那如果没有自动导入怎么办内，假如你的learngit文件夹在桌面上，可以添加本地仓库，这是一样的



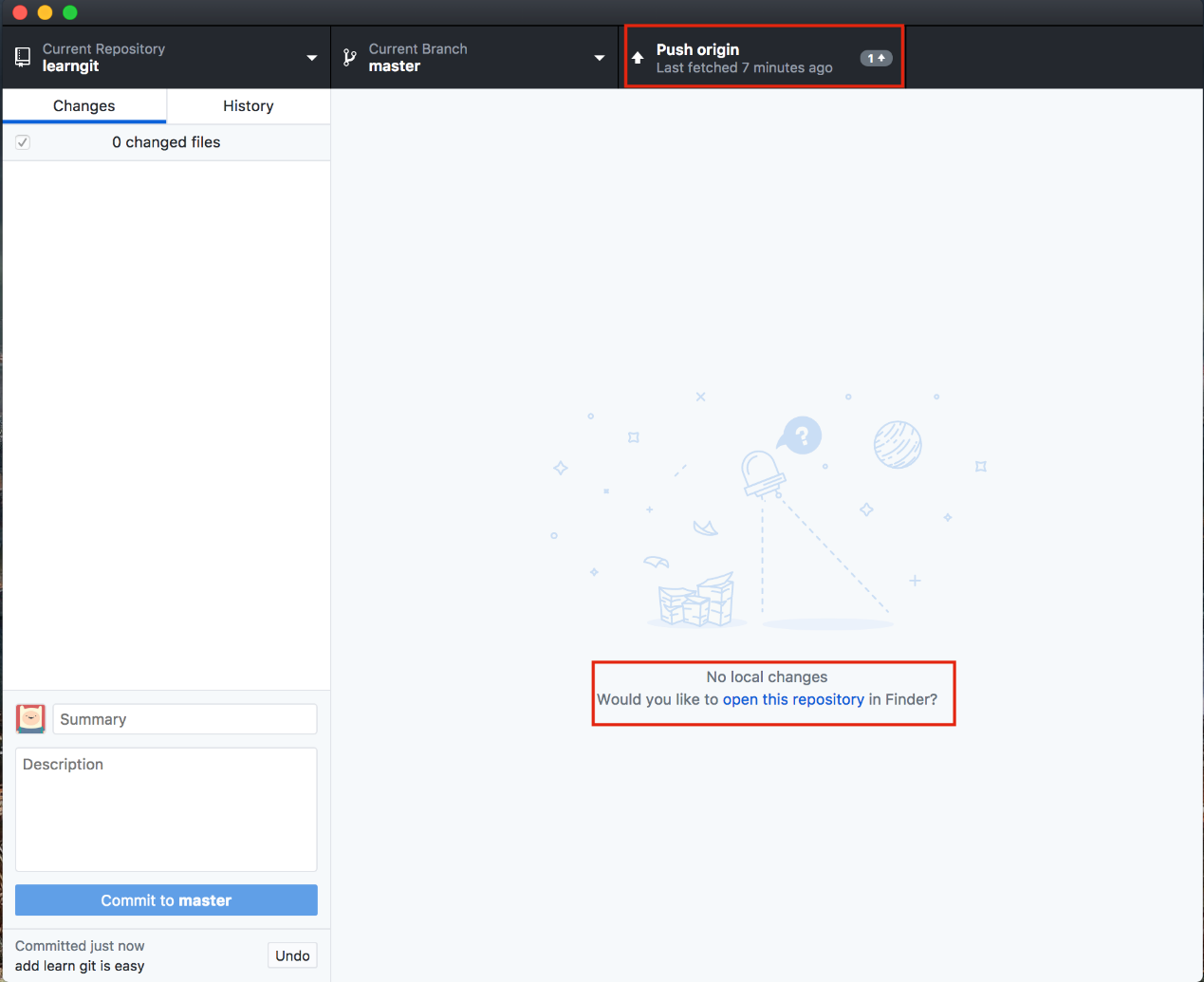


**提交改变**

好了咱们已经有了本地仓库了，那么现在当然是什么改变都没有，也不需要提交什么。可以看到我的learngit中有一个learn2.txt文件，我们就做一些修改吧！  
在其中添加一行learning git is easy.看看有什么变化吧！



看到了吧，已经显示了我们在第4，5行新添加了改变(其中第4行为空白行), 那现在就可以在summary写上简短的改变描述，比如我写的是add learn git is easy之后点击commit to master就可以了。

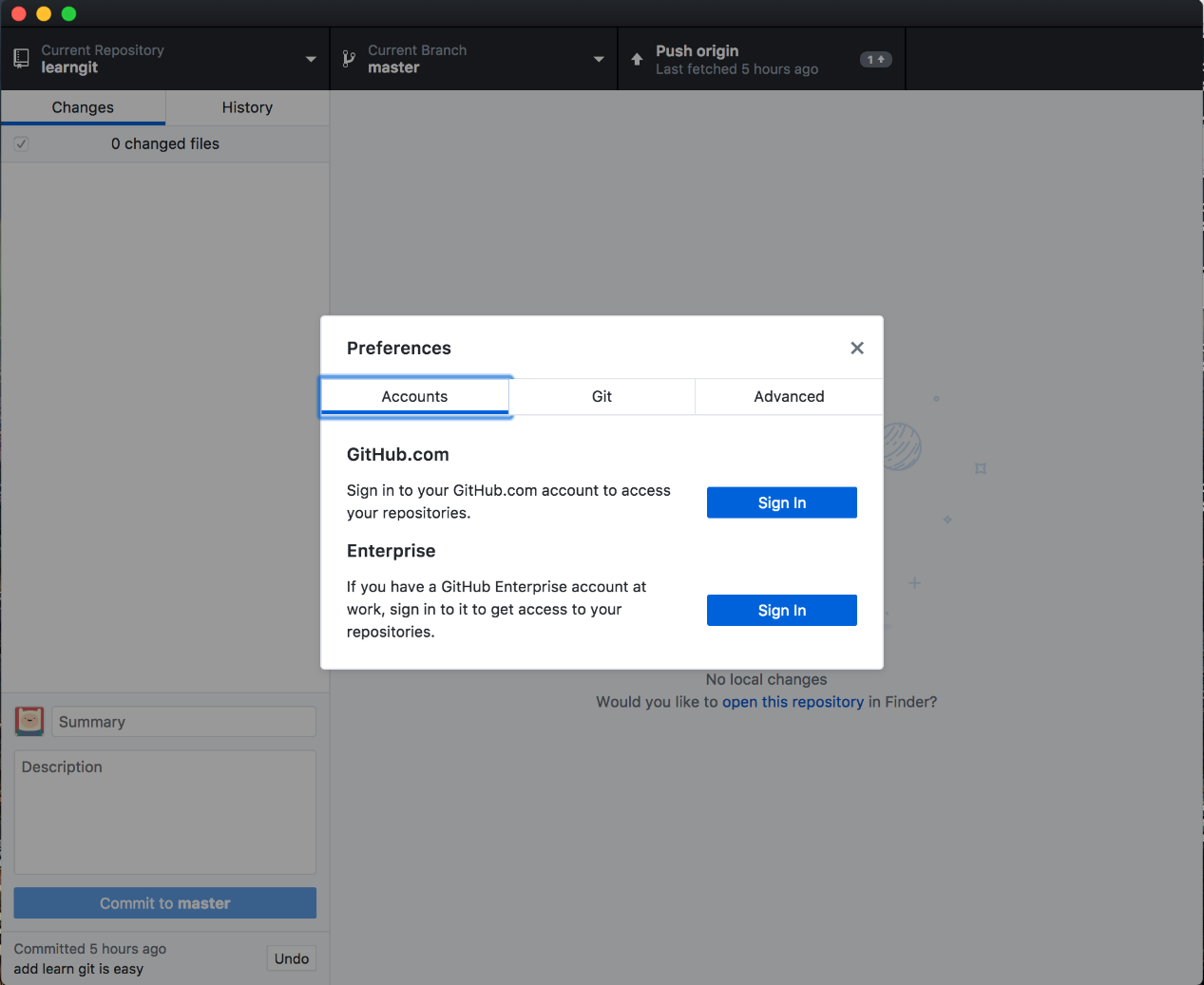


看到了吧，现在显示本地没有改变，但是上面push origin显示了1，代表的是我们与远程的github不同步，本地有一个更新，就是我们新加的2行，但是github并没有更新，推送远程分支我之后会讲。

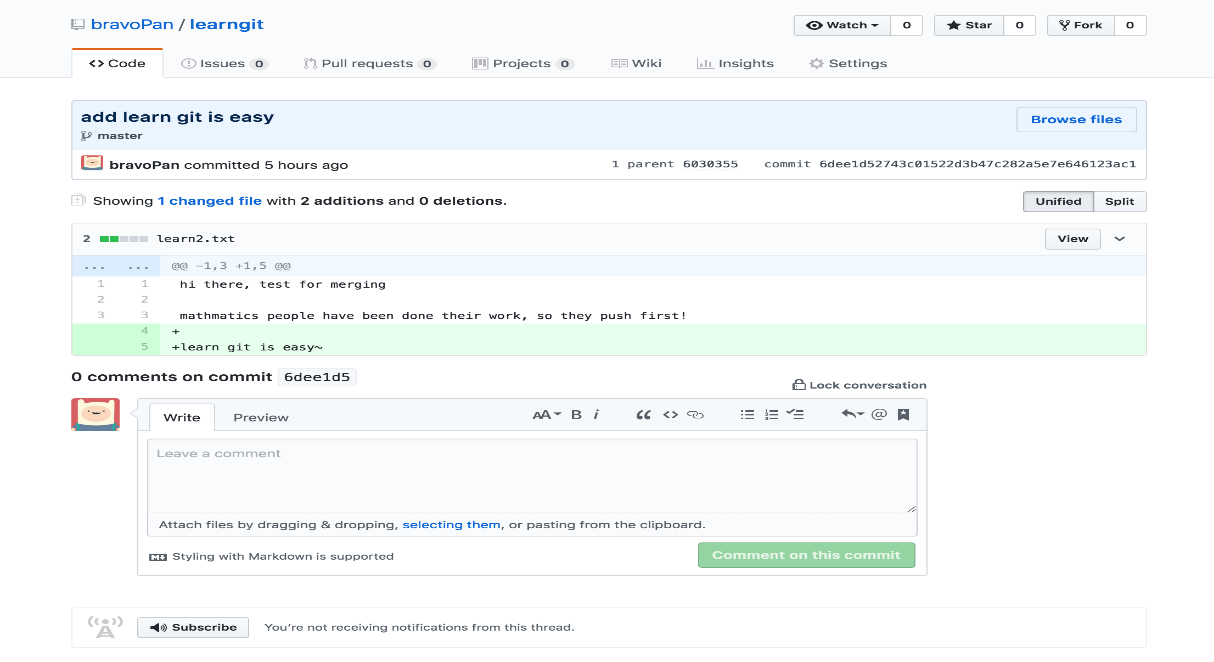
**推送到远程仓库**

平常我们都会看到的别人github上面优秀的项目，github是一个远程仓库，你可以把它当作展示用的，或者别人可以克隆你的仓库。所以如果你想看到自己本地仓库在github上面展示出来每次就得push出来，就像是我们上面commit后本地比远程多了一次更新，需要更新远程仓库push本地修改。

* 首先你得有自己的[github账号](https://link.jianshu.com?t=https%3A%2F%2Fgithub.com%2F)，这是为了我们的远程推送到github上的
* 在自己的github desktop登陆自己的github账号密码



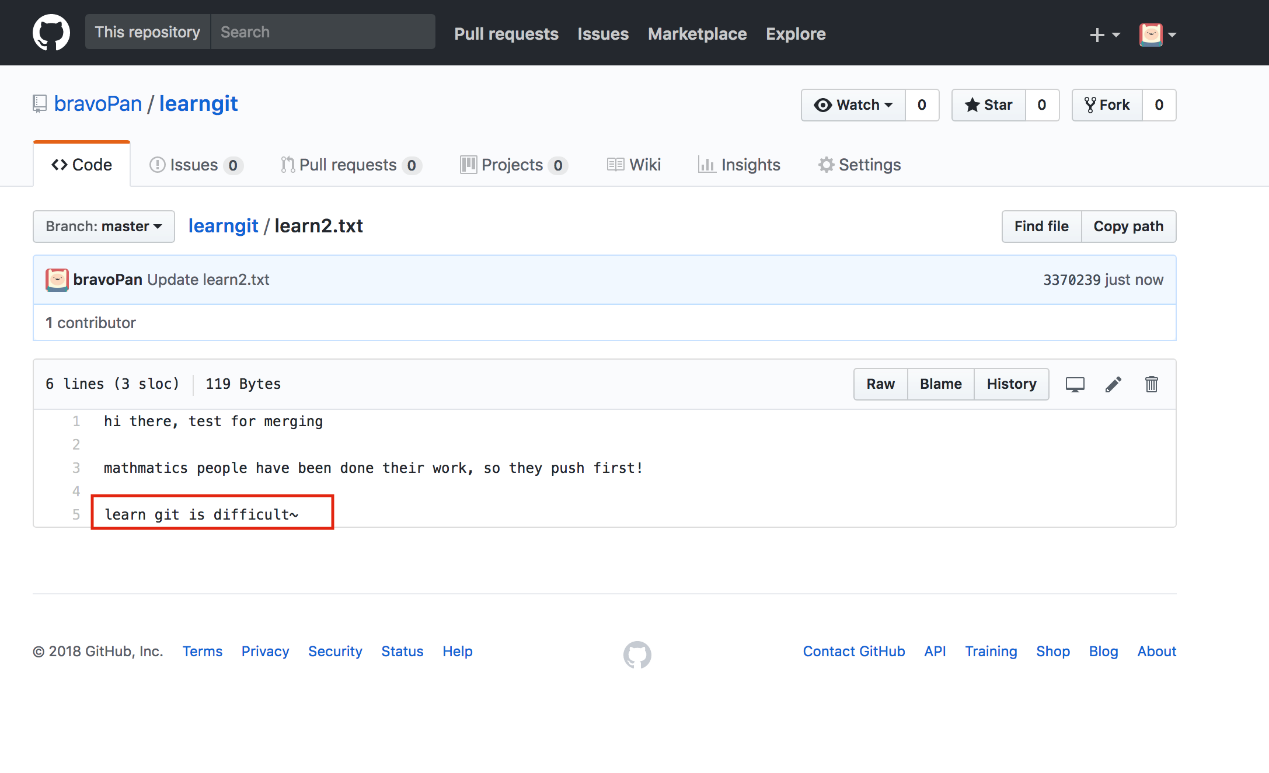
登陆后就可以推送啦， 点击push origin就可以啦～ 你可以登陆github查看learngit仓库，看看其中的变化



是不是和我们在github desktop里看到的一样呢？ ：>

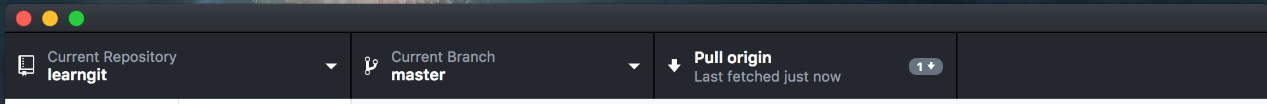
**更新本地仓库**

比如说现在远程仓库已经被更新了，有可能是你的同事提交了他的一部分，但是在你的本地仓库并没有更新，现在怎么办呢？ 很简单，一键fetch

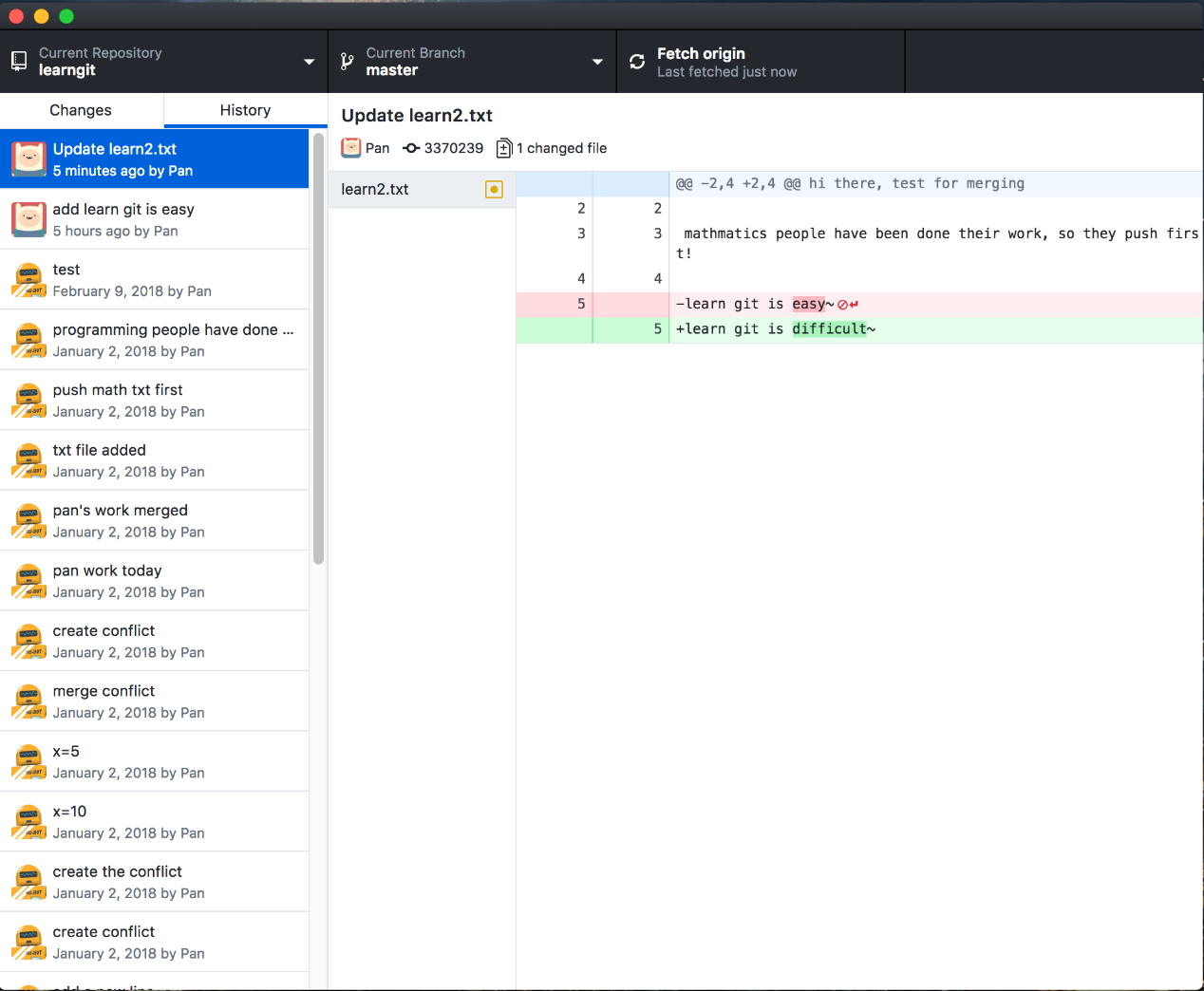


把easy改成了difficult

这时候点击fetch，可以看到下面的pull origin

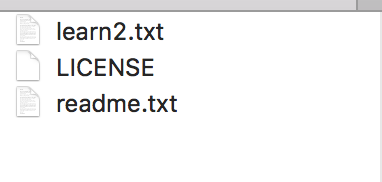


好啦～点击pull origin就可以把远程的difficult更新到本地了~ 看看里面的history就知道干了些什么了。

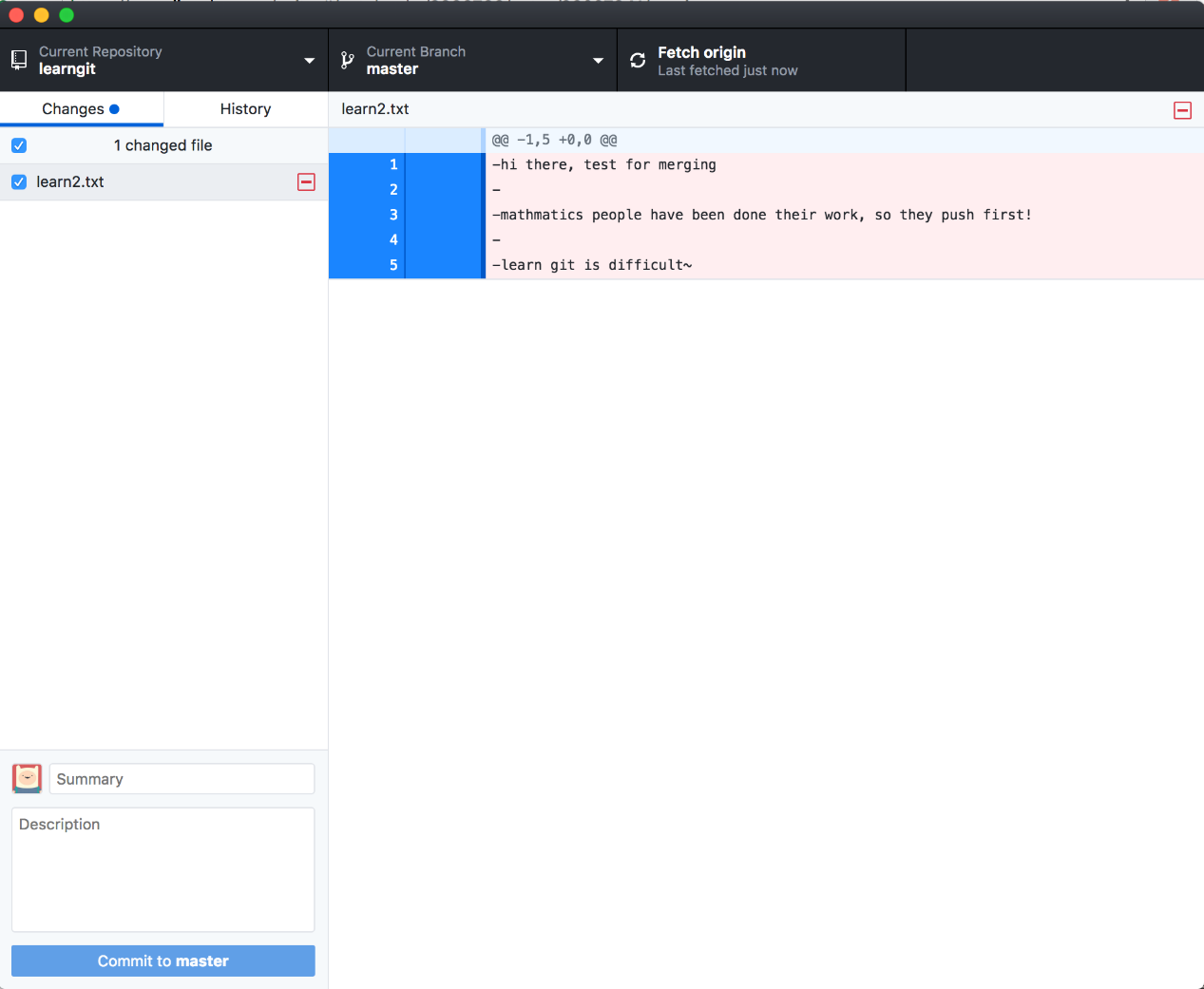


**版本回退**

有很多时候我们在当前这一步骤做了一些不可挽回的错误，比如说删除了重要的文件以后再也找不到了，这时候使用版本回退可以回退到任何一个commit过的状态。 现在看看咱们的文件夹都有哪些文件？

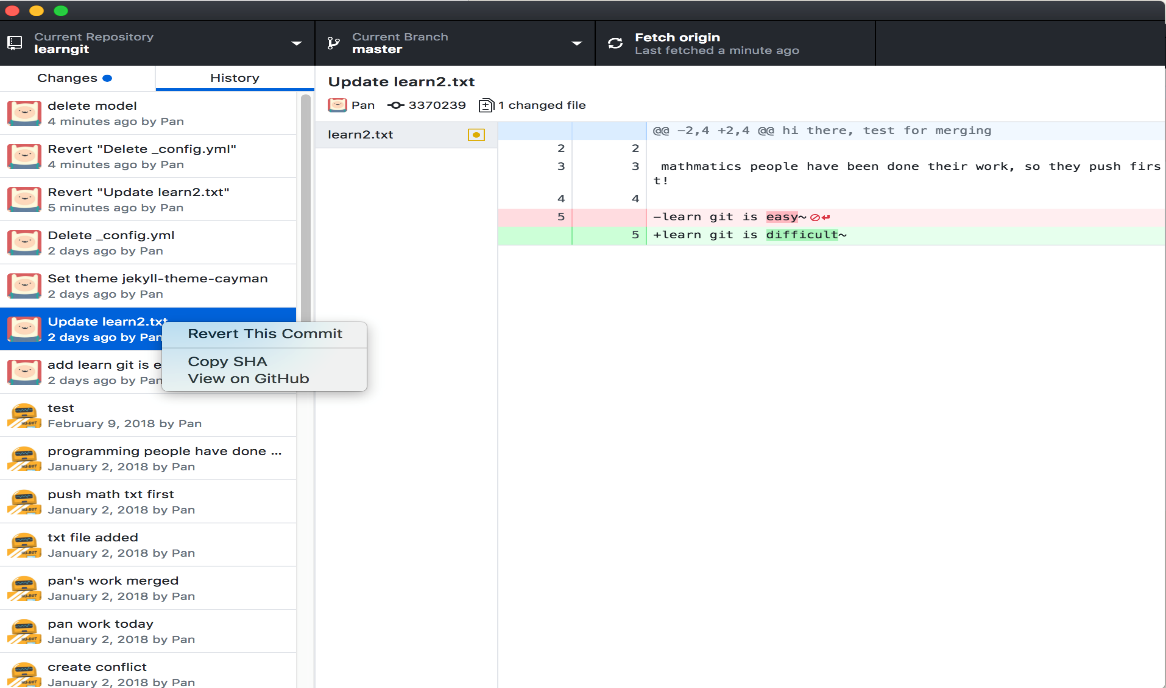


比如说我们不小心删除了learn2.txt，这个文件非常重要，怎么样回复到之前存在的状态呢？

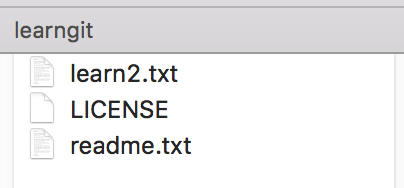


learn2.txt被删除后的changes

打开history你会发现有很多commit后的历史记录,其中有我们之前的update learn2.txt。所以右键它会显示revet this commit

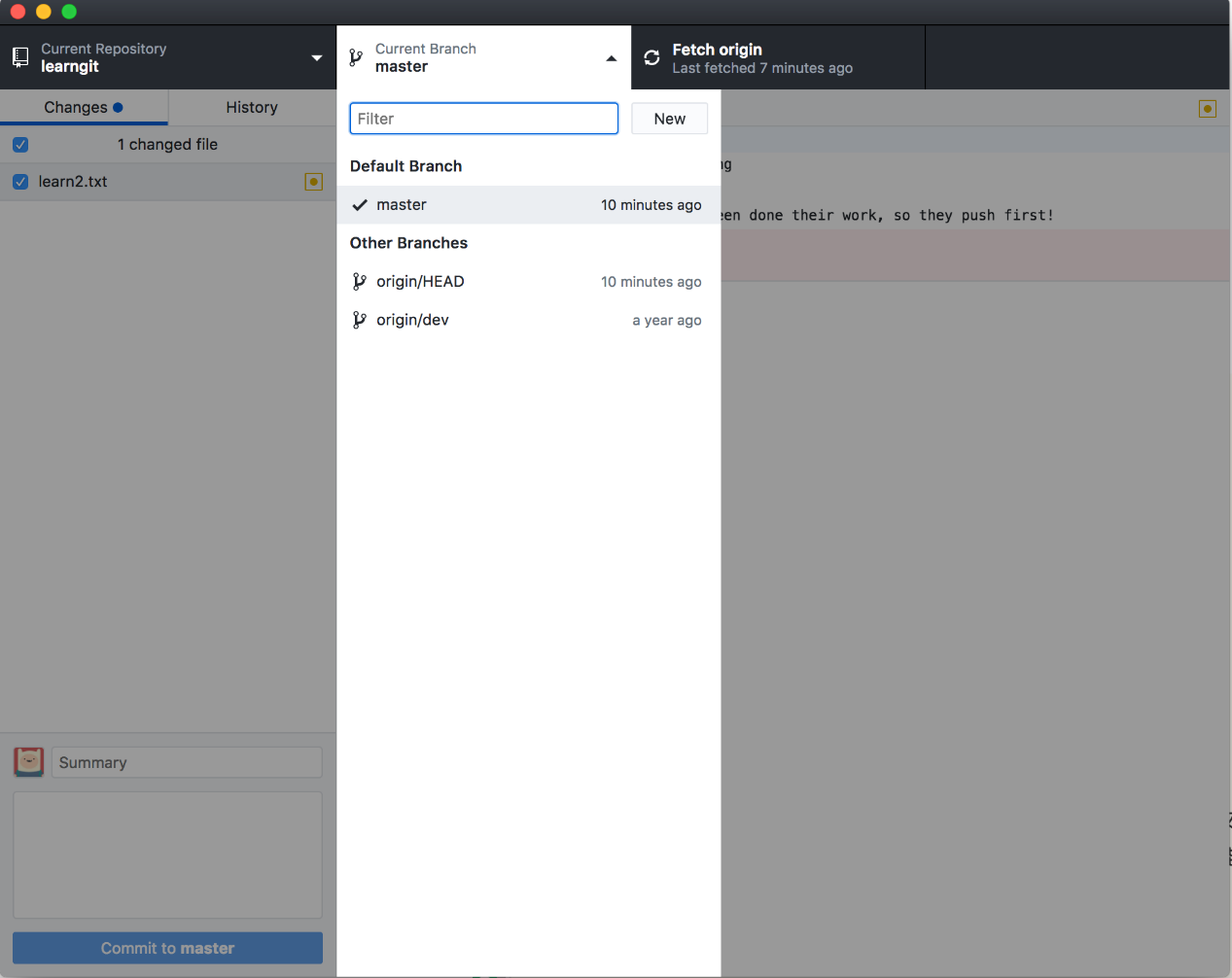


好了，现在看看你的文件夹吧 :> 是不是回来了呢？

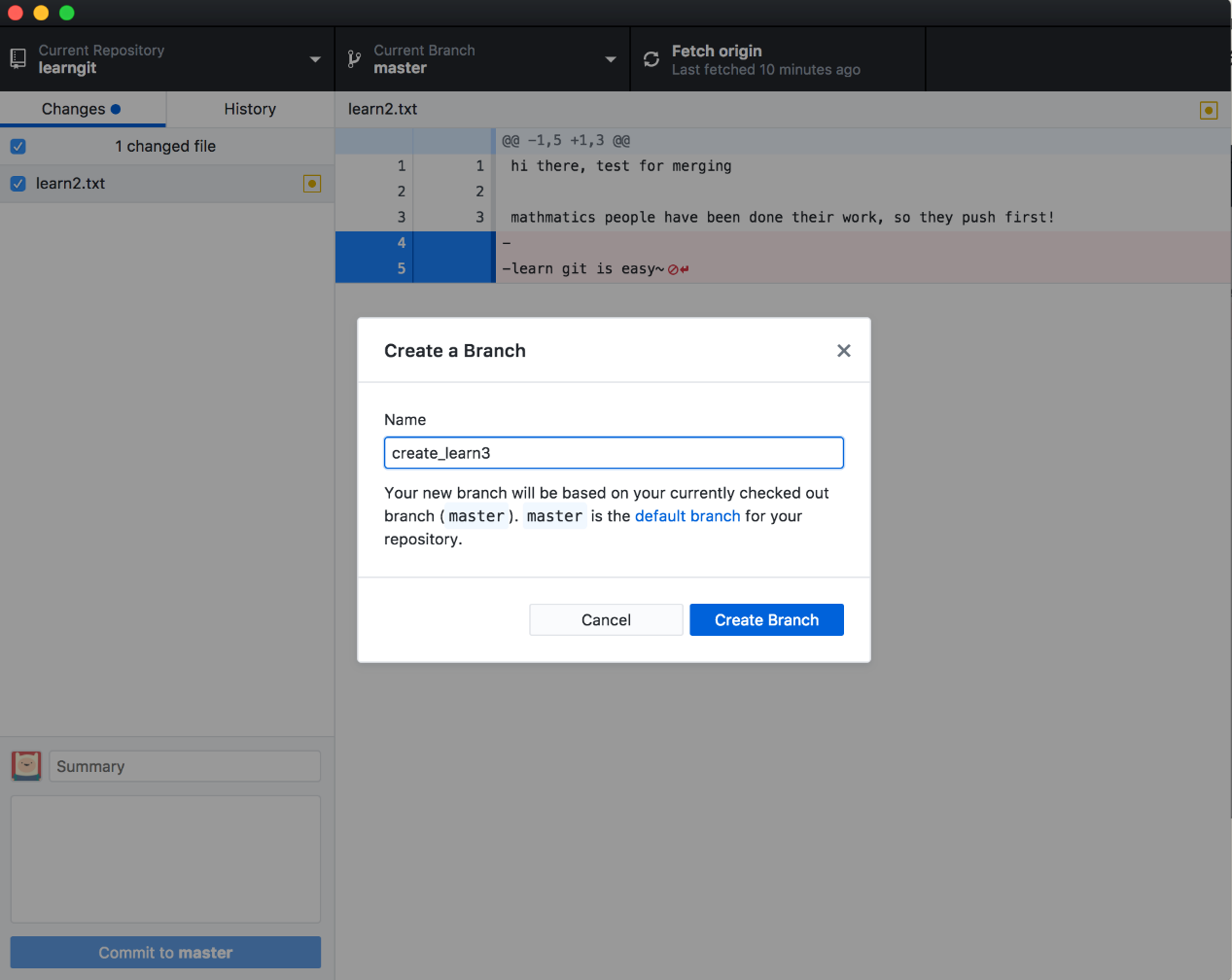


**创建分支**

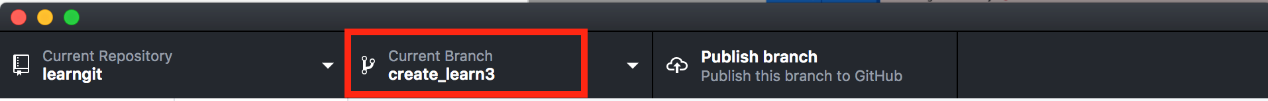
什么是分支呢？就像本篇文章刚开始提到的分支是用来创建新功能但是你又不希望破话现在的成果，害怕会对现在的进度造成影响，所以是一种试验性的功能。 那怎么创建呢？ 这也是很简单的，打开首页的current branch会看到default branch是master，这是所有git仓库的默认主分支，都叫master，origin是你github的分支，关联的是服务器端。



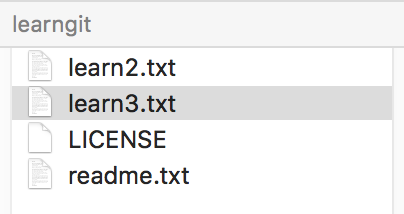
我们尝试创建一个新的分支，点击new，创建一个名为create\_learn3的分支



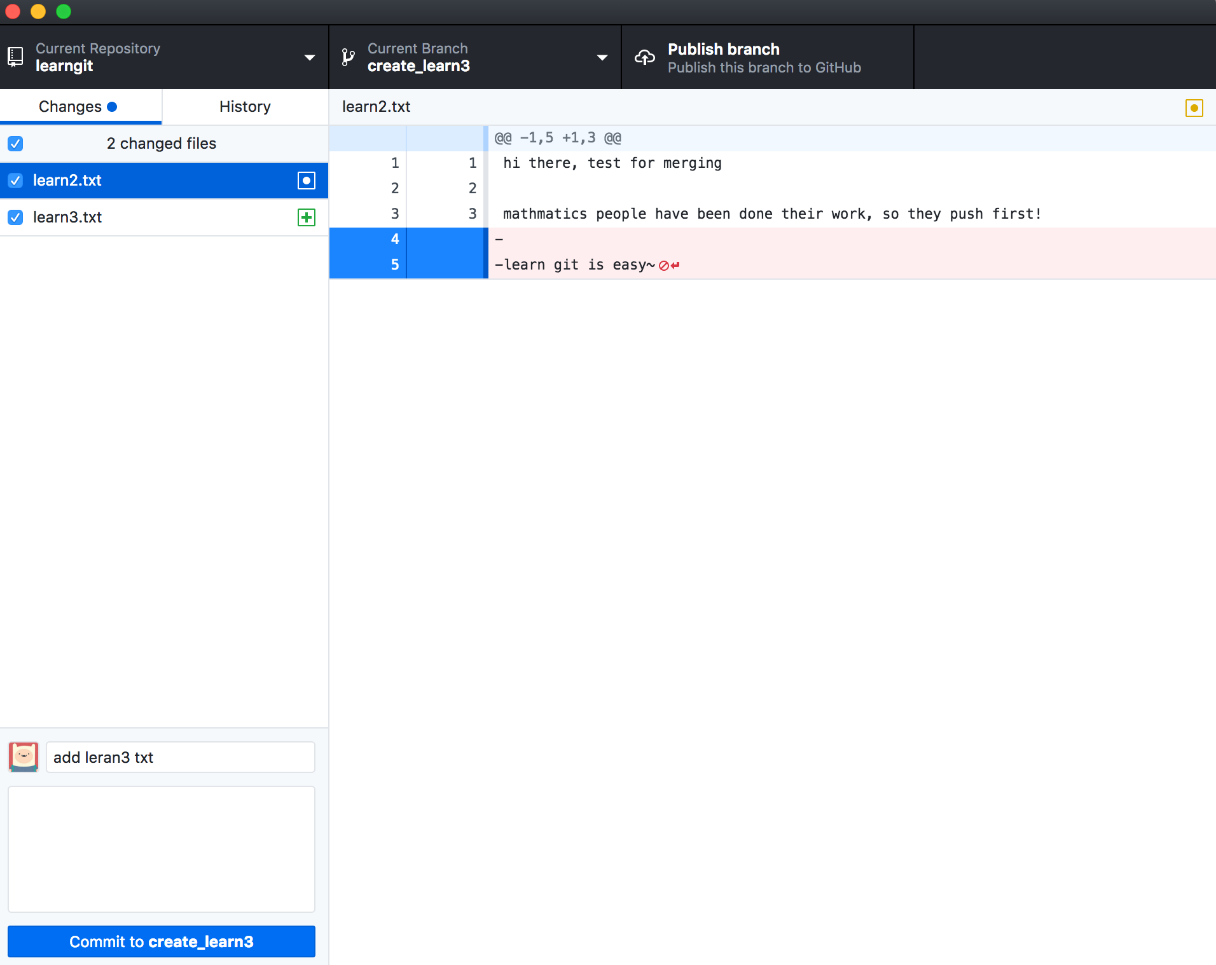
如果你现在仔细观察的话会发现原来的master分支变成了create\_learn3，这说明我们当前处于create\_learn3的分支里



点击create Branch就可以了,当然了我们会在这个分支里创建一个learn3.txt的文件夹



好了你看到我创建了一个learn3的文件，接下来就是提交我们的改变，让git记住当前分支的状态



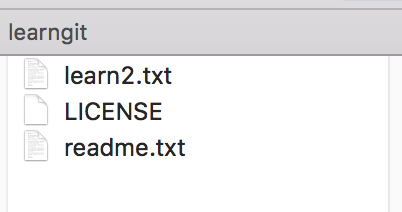
提交learn3.txt

现在如果我们切换回master分支



切换回master分支

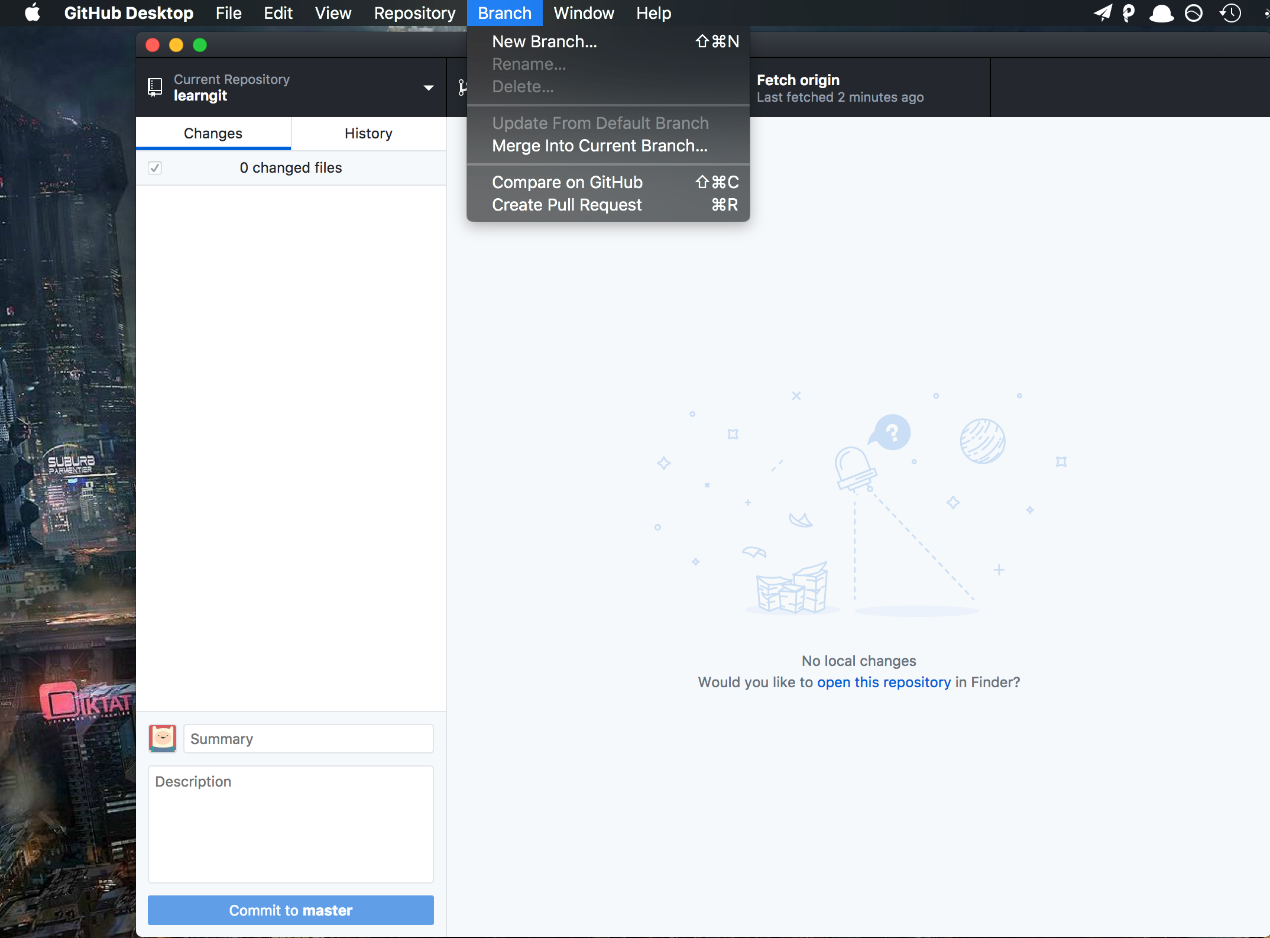
然后查看文件夹，你会发现神奇的事😮



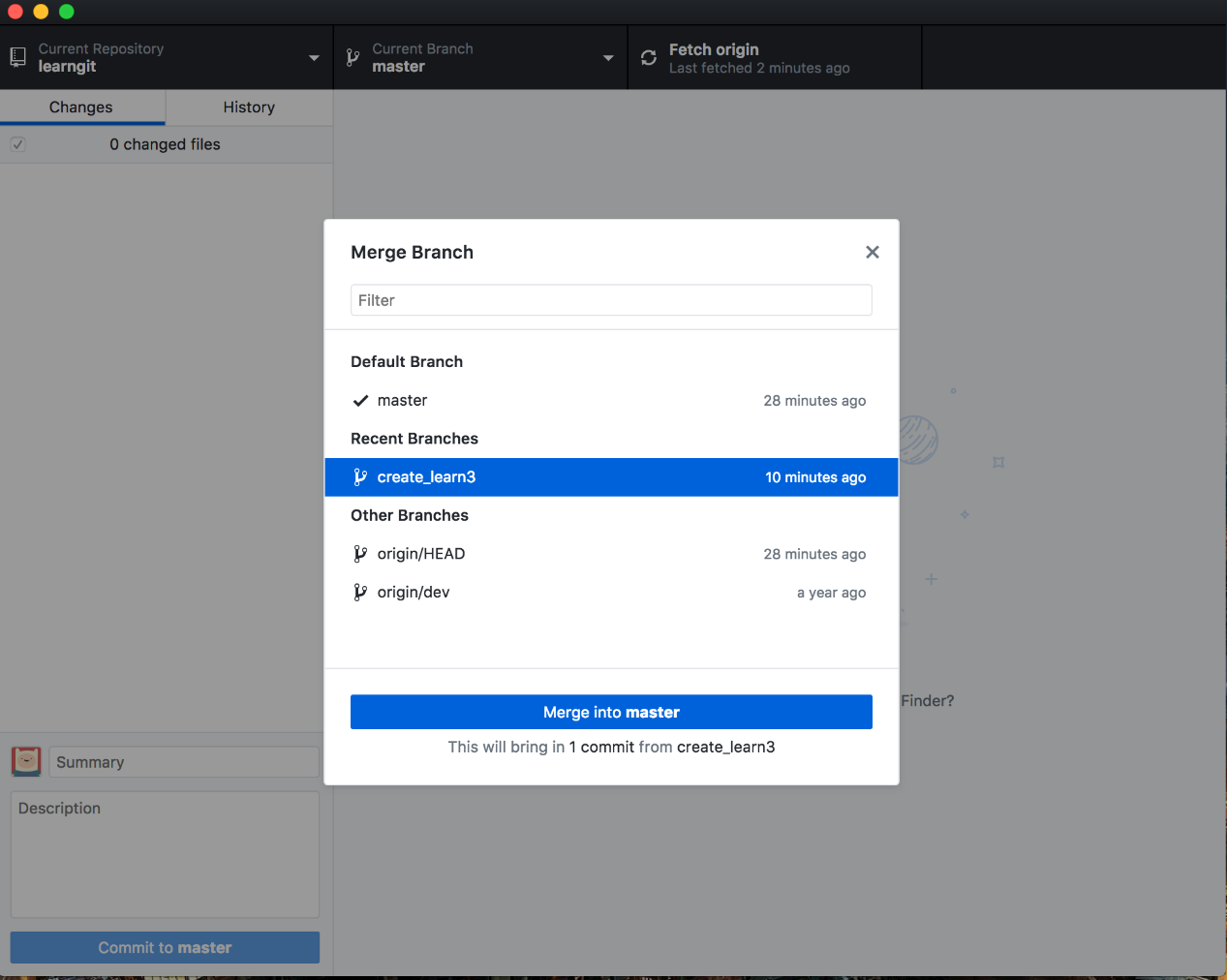
竟然没有我们刚刚创建的create\_learn3.txt，这是因为我们在create\_learn3分支里提交了改变，**现在比master分支早一个commit**. 你现在知道分支的作用了吧！它不会改变我们主分支，如果你在其他分支创建commit，它只会改变其他分支的状态，而对于master状态不会做出任何改变！

**合并分支**

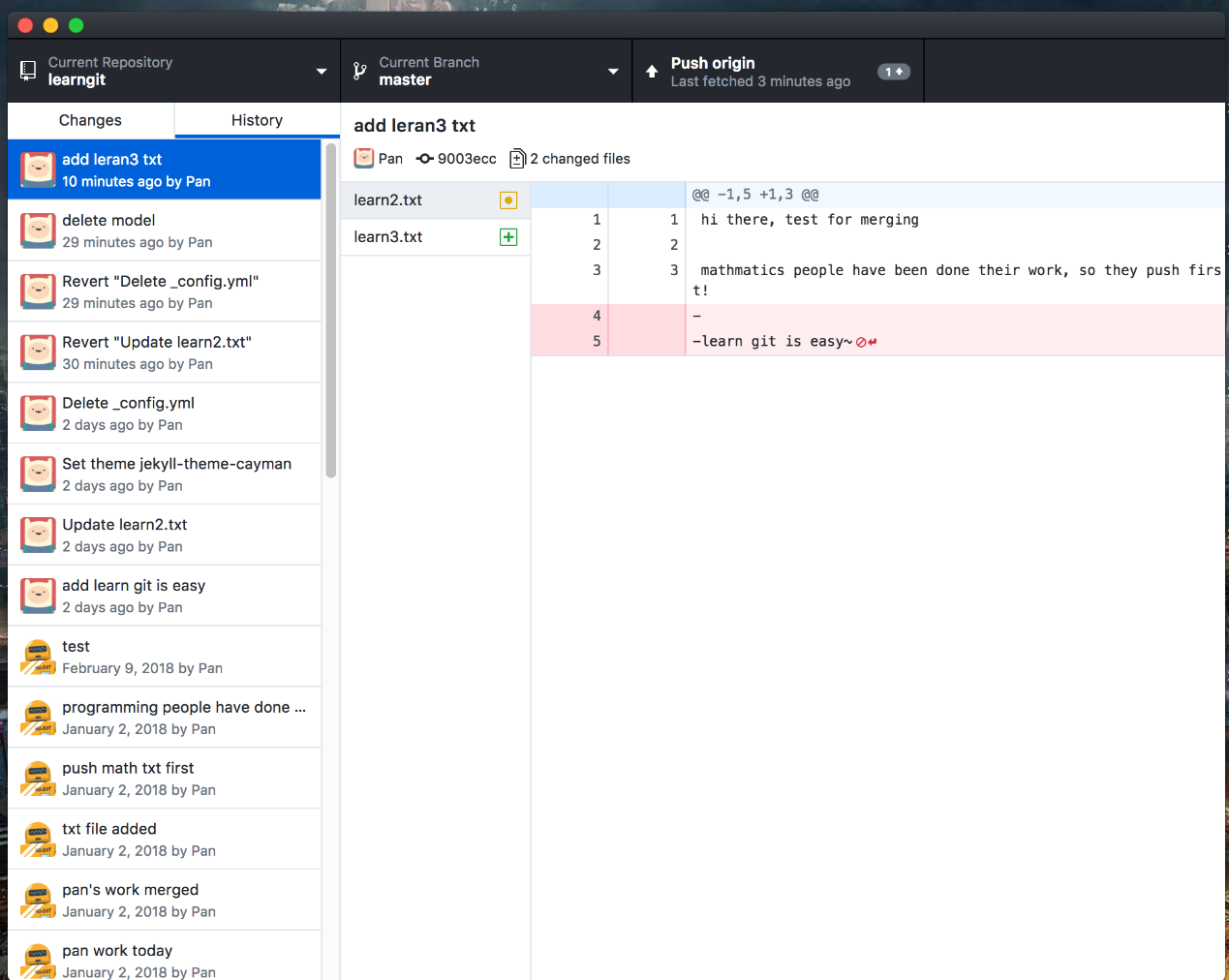
现在你也许想创建learn3.txt是一个不错的试验，我想把它合并到我原来的master分支，那怎么做呢？ 首先打开branch选项



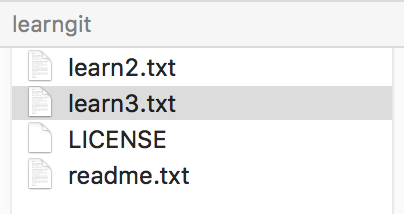
点击其中的merge into curren branch(当前处于master分支,永远都是把其他分支merge到当前！) ，然后选择一个分支，就是我们的create\_learn3



点击merge into master，你会发现多了一个commit，他是从我们create\_learn3分支继承过来的



最后再看看我们的文件夹



真的多了learn3.txt呢！

**End**

好了最基本的git操作我已经演示完了，真的没有用到一个命令，不知道你是否学会了这个得心应手可以省时省力的工具呢?

。

**方法一：**

    从GitHub下载文件一直非常慢，查看下载链接发现最终被指向了Amazon的服务器，下载地址是http://github-cloud.s3.amazonaws.com/，从国内访问Amazon非常慢，所以总是下载失败，解决方法时更改host文件，使该域名指向香港的服务器：

更改hosts文件：

* Windows

更改C:\Windows\System32\drivers\etc\hosts文件，在文件中追加219.76.4.4 github-cloud.s3.amazonaws.com, 将域名指向该IP即可

* Mac

执行 sudo vi /etc/hosts 追加 219.76.4.4 github-cloud.s3.amazonaws.com

**最后执行ipconfig /flushdns命令，刷新 DNS 缓存。**

**方法二：**

[**https://www.ipaddress.com/**](https://www.ipaddress.com/)**使用 IP Lookup 工具获得下面这两个github域名的ip地址，该网站可能需要梯子，输入上述域名后，分别获得github.com和github.global.ssl.fastly.net对应的ip，比如192.30.xx.xx和151.101.xx.xx。准备工作做完之后，打开的hosts文件中添加如下格式，IP修改为自己查询到的IP：  
  
192.30.xx.xx github.com  
151.101.xx.xx github.global.ssl.fastly.net  
  
最后执行ipconfig /flushdns命令，刷新 DNS 缓存。修改后的下载速度能达到 200KB/S 以上。**