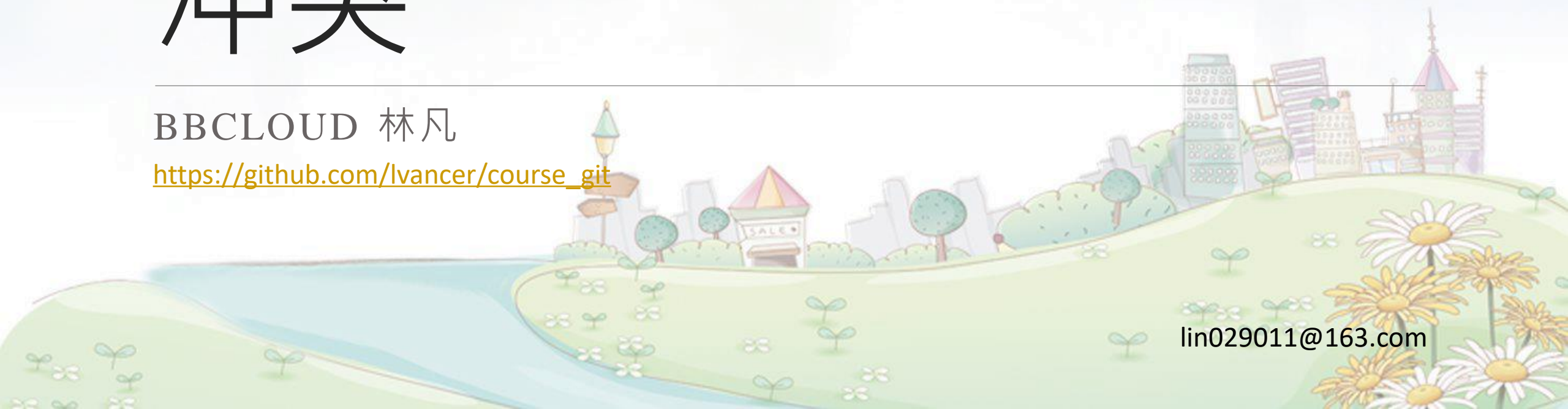


冲突

BBCLOUD 林凡

https://github.com/lvancer/course_git

lin029011@163.com



大纲

冲突

解决

练习



冲突

冲突是在多人协作开发过程中常见的情况。

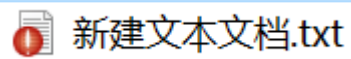
我们先学习在同一分支下的冲突问题。

在同一分支下，冲突的问题表现为多人对同一文件进行了修改。

模拟冲突：

1、先克隆同一个项目到两个文件夹。此时两个项目都是最新的

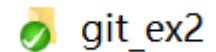
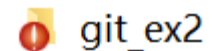
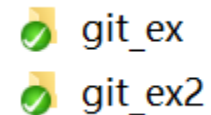
2、修改第一个项目中的文件，提交到仓库。



3、再修改第二个项目中的同一个文件，相当于另一个人在同时对项目进行了修改。

4、这时如果进行Commit操作，是没有问题的。因为Commit只提交到本地。

5、但当进行Push操作时，就会出错，这就是冲突。



冲突

冲突特征：

报错时，下面的标是红色的

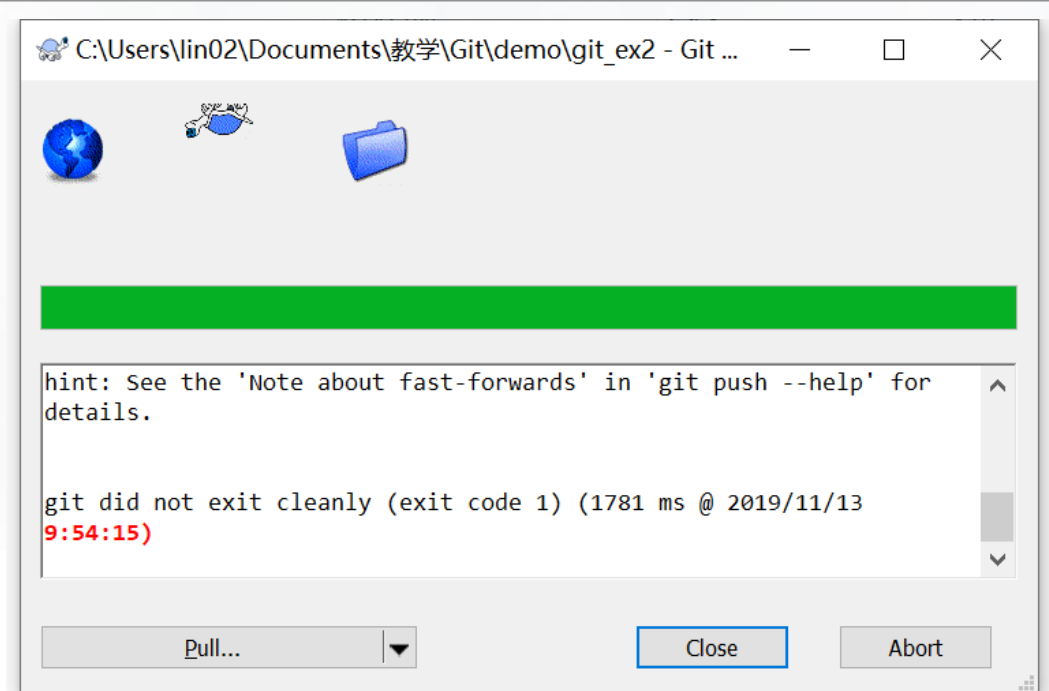


弹出框里显示

error: failed to push some refs to ...

git did not exit cleanly

并在左下角提示你去进行Pull操作。



解决

1、先进行Pull操作，但是还是会报错。

信息显示：


CONFLICT (content): Merge conflict in 新建文本文档

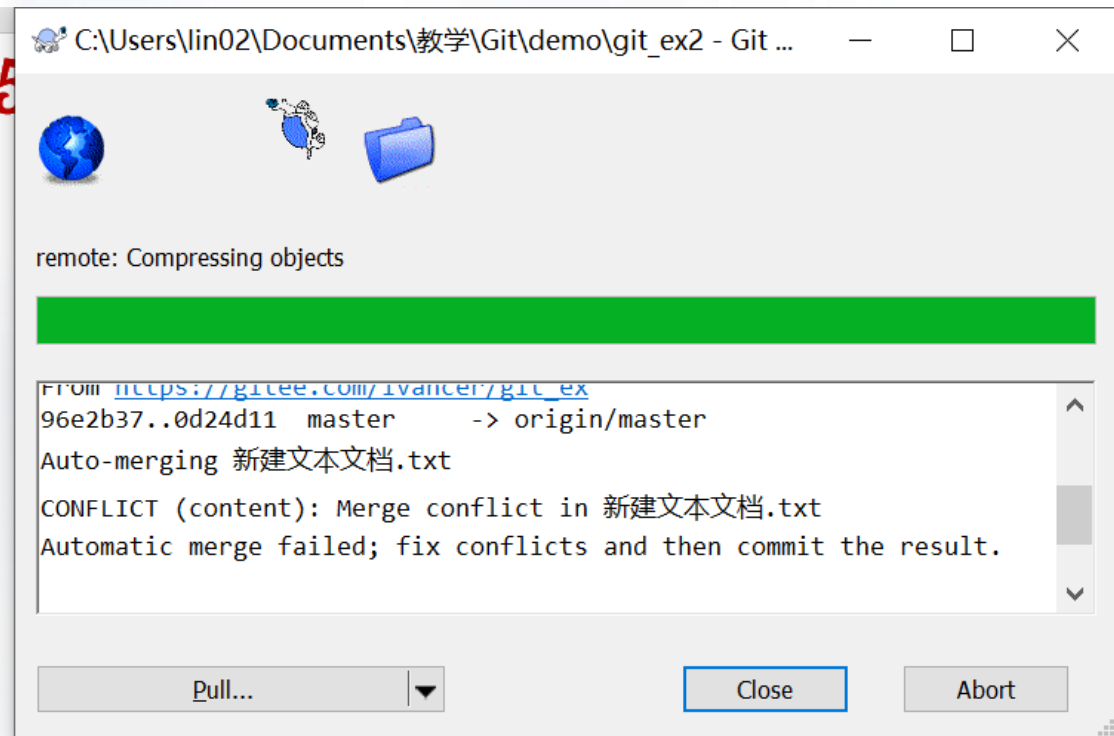
Automatic merge failed; fix conflicts and then commit

翻译过来就是遇到冲突，无法将线上仓库和本地的这个文件进行自动合并。

2、如果没有出错，那说明线上仓库和本地虽然有冲突，但是可以自动合并。

3、有冲突的文件会显示成警告状态。

 新建文本文档.txt



解决

4、打开冲突文件，看到出现了奇怪的符号。

```
dddd
```

```
<<<<<<< HEAD
```

```
嘻嘻222
```

```
=====
```

```
呵呵
```

```
>>>>>>> 0d24d118a2d009ed3f6ab1dce960f87e64e1501d
```

本地版本

线上版本

这就是git里展示冲突的方式，HEAD是本地的最新版本内容，

下面的一串数字代表线上仓库的一个版本，每一次commit都会产生一个这样的版本号

所以我们可以知道的是本地版本是“嘻嘻222”，线上是“呵呵”，所以有冲突。

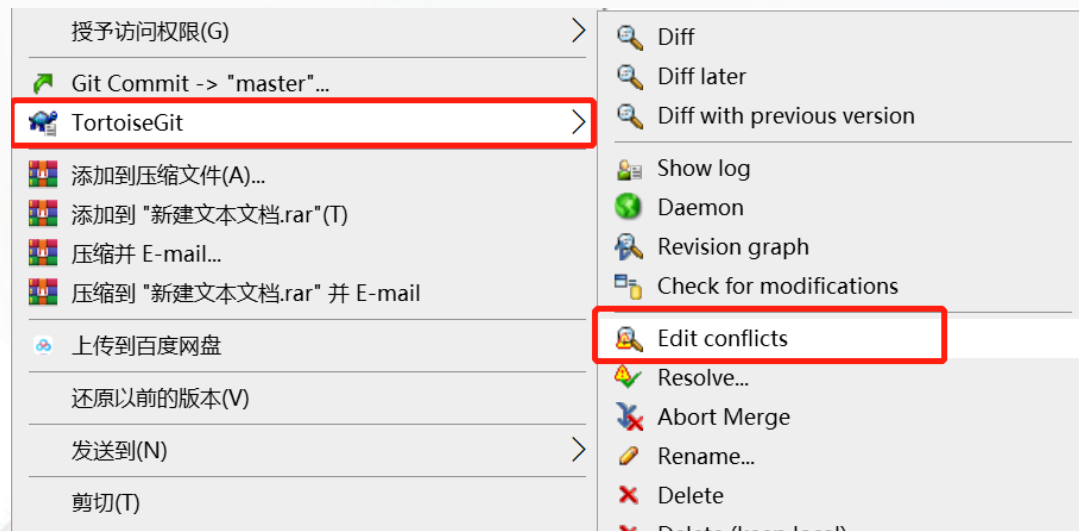
解决

5、一种解决方式是直接进行修改，将冲突的几行删除，修改成我要的。

6、另一种方式是使用冲突编辑工具。

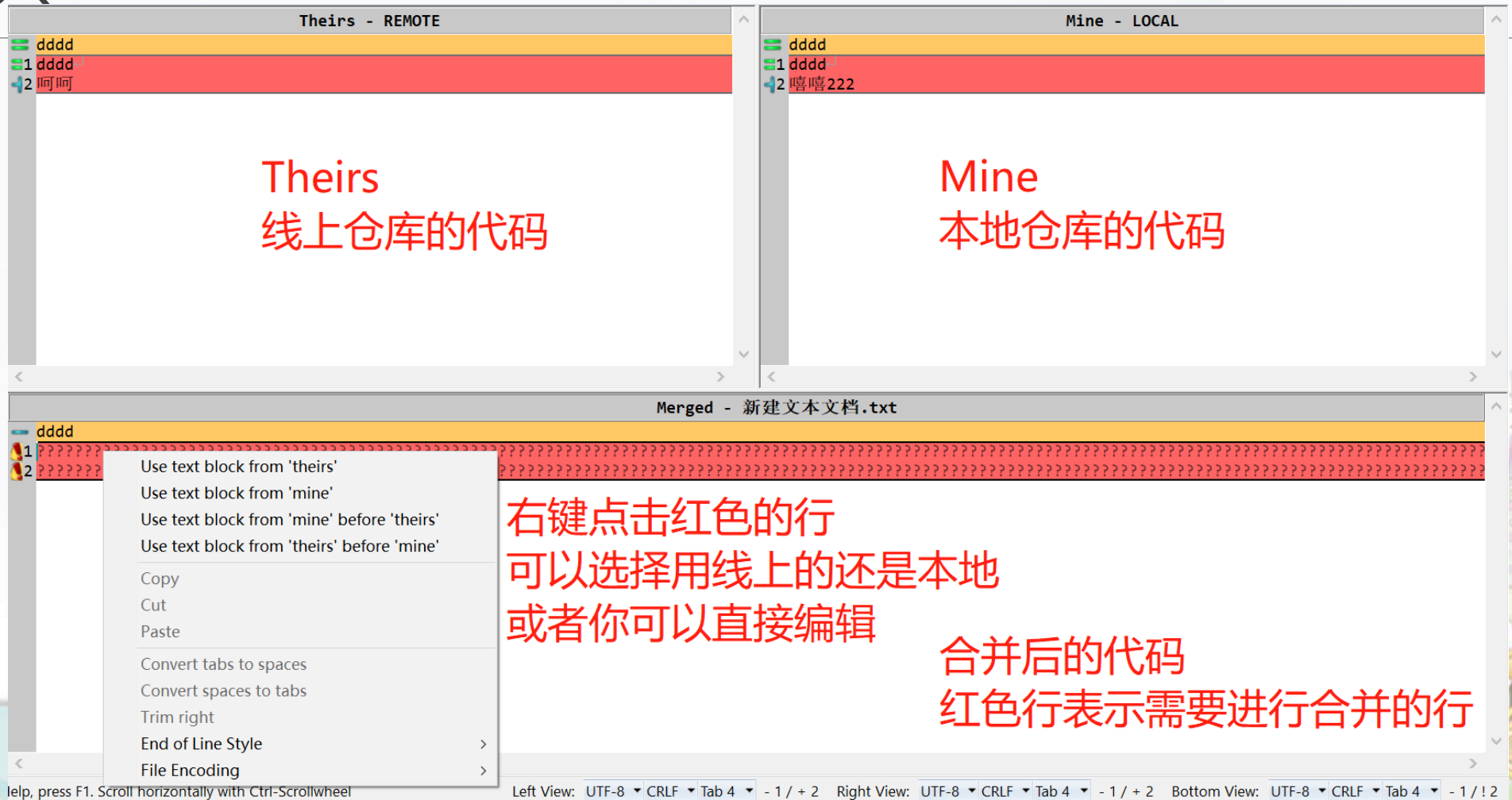
右键文件，**TortoiseGit** => **Edit conflicts**

dddd
嘻嘻222



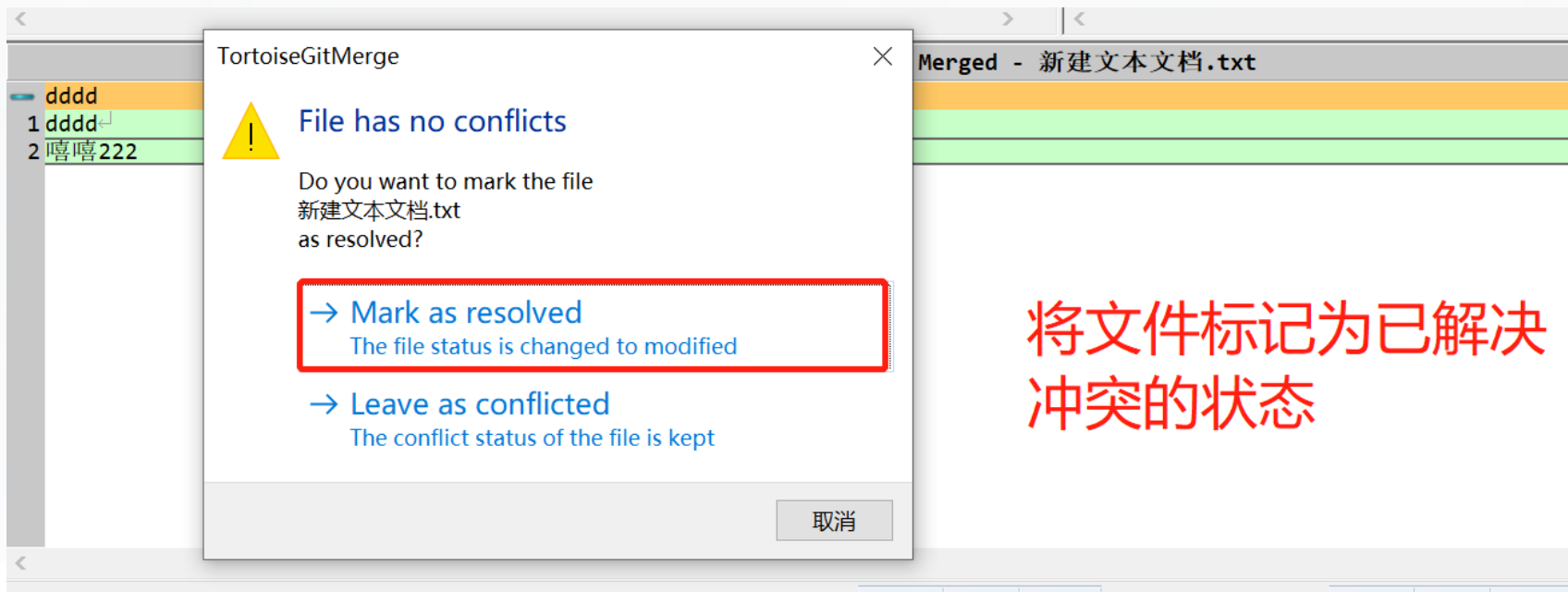
解决

7、



冲突

8、编辑完成后，保存。



将文件标记为已解决
冲突的状态

9、之后就可以正常进行commit和push操作了。

新建文本文档.txt

练习

尝试个人模拟冲突，或者几个人共同使用一个git项目做一个真实的冲突。

练习如何解决冲突，成功上传。

