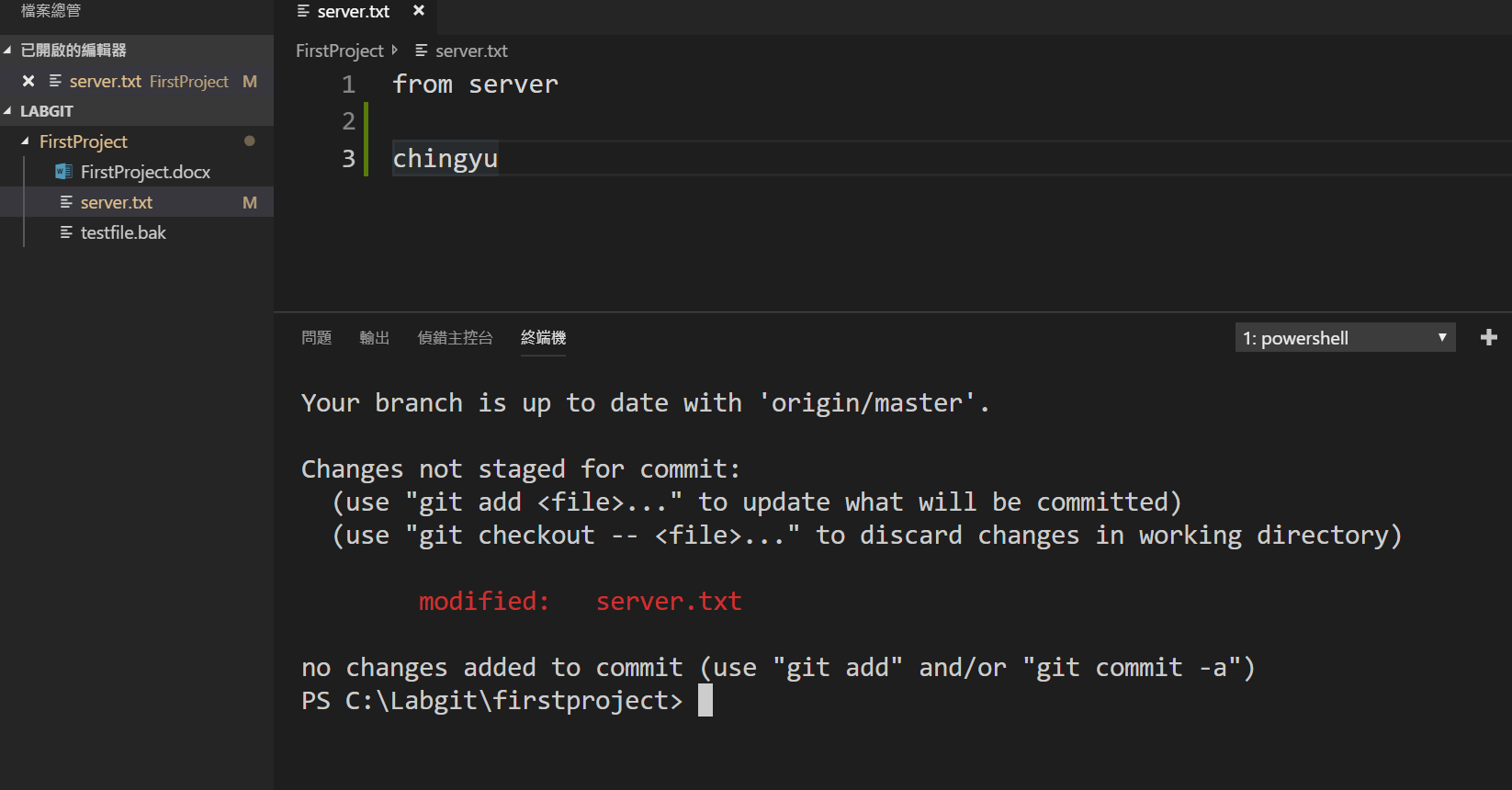
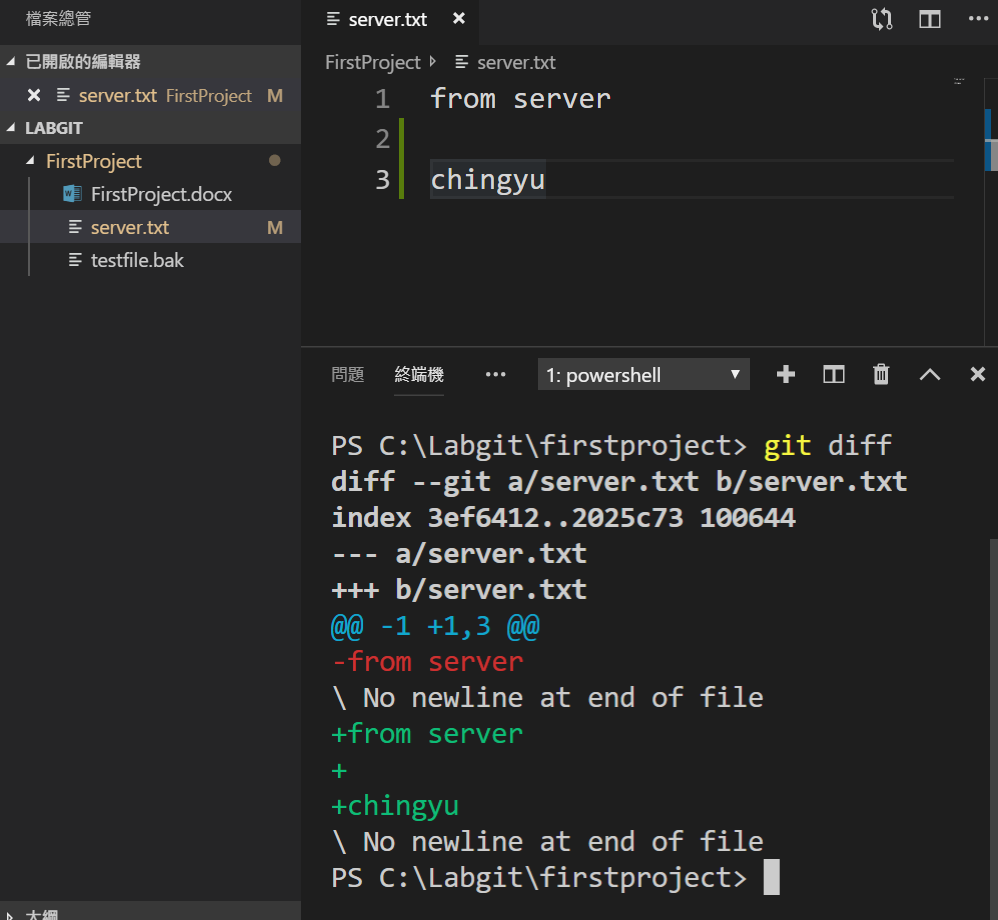
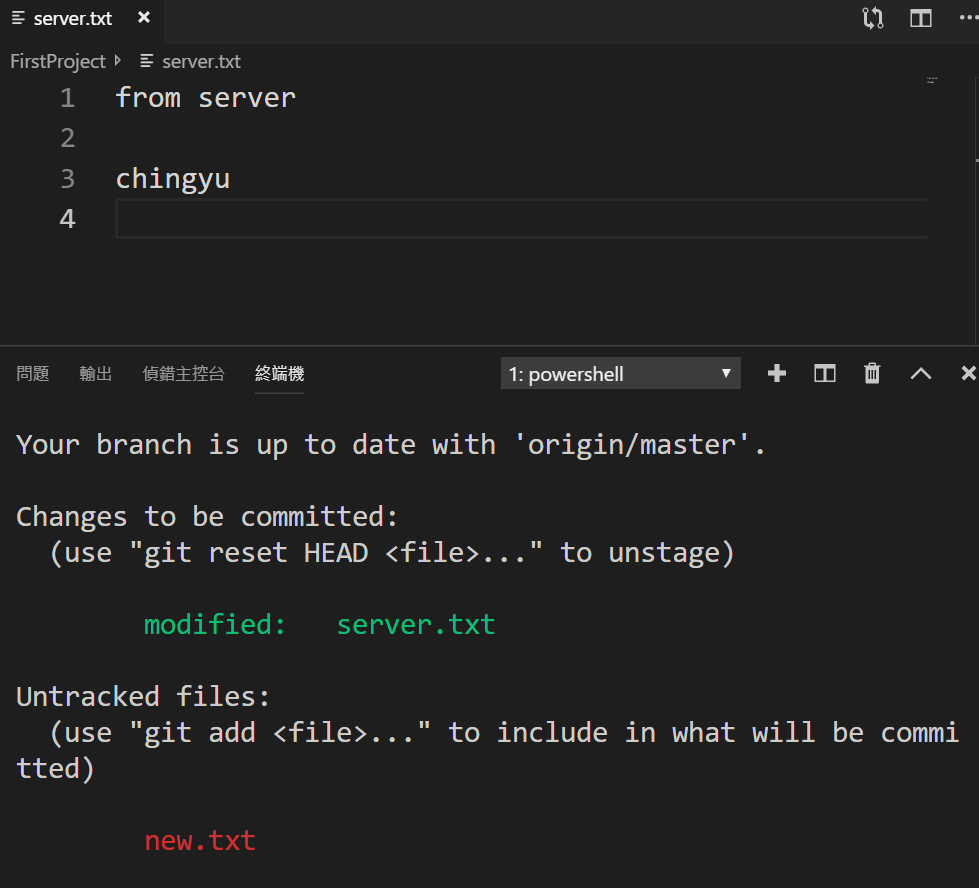
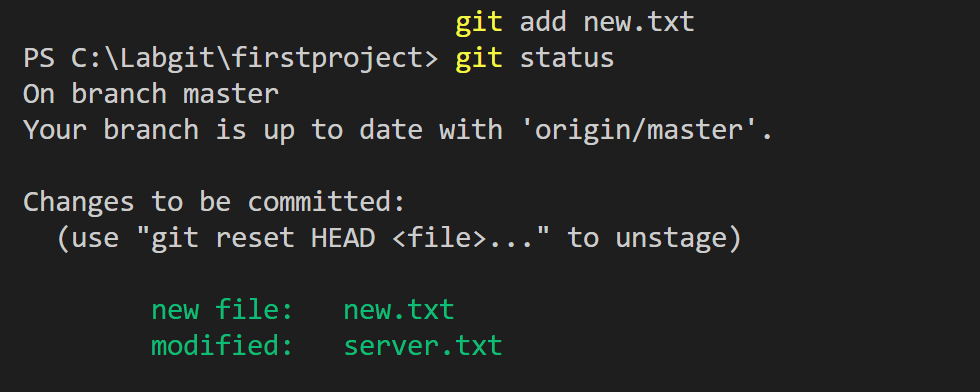
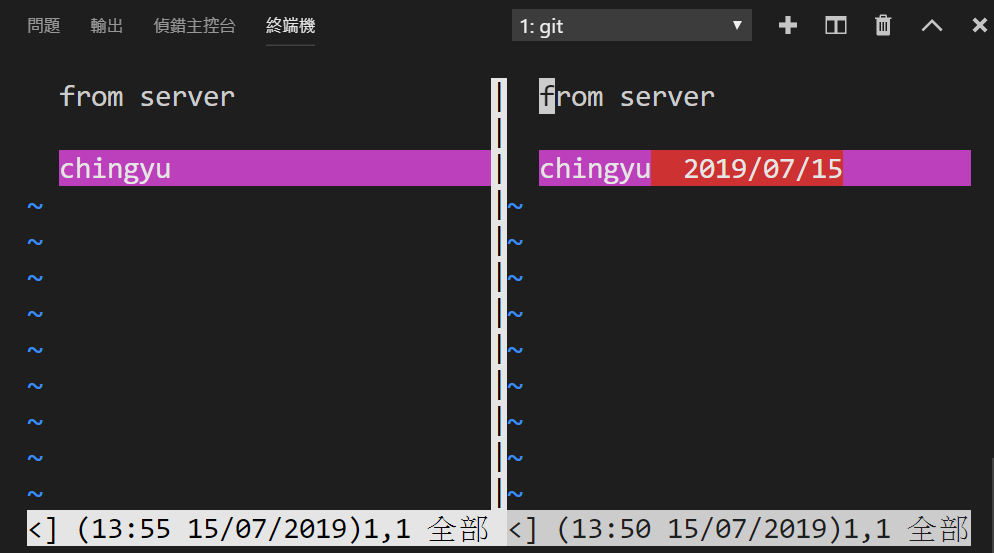
1. 實際操作-FirstProject
   1. 透過GitHub網頁選擇import repository，匯入測試專案<https://github.com/iiiTechDemo/FirstProject.git>
   2. 將GitHub上的檔案下載至LabGit 資料夾
      1. git clone <https://github.com/你自己的帳號/FirstProject>
      2. git status
      3. cd FirstProject
      4. git status
2. 開始執行底下指令前，確認當前路徑是否為GitHub上下載下來的專案
3. 開啟 server.txt 檔案，加入自己的名字後存檔；底下指令請分開執行，並逐步查看
   1. git status
   2. git diff 
   3. git checkout -- server.txt 還原
   4. git status
   5. git diff
4. 重新加入自己的名字，將「工作區」檔案修改添加到「暫存區」；但，如果添加到「暫存區」後想取消的話?
   1. git status
   2. git add server.txt
   3. git status
   4. git diff
   5. git reset HEAD server.txt
   6. git checkout -- server.txt
5. 再重新加入名字，並且執行 git add server.txt
6. 在專案裡面手動新增一個檔案 new.txt，將「工作區」新增的檔案添加到「暫存區」
   1. git status 
   2. git add new.txt
   3. git status 
7. 再次打開server.txt檔案，加入今天日期，再次將「工作區」檔案修改添加到「暫存區」
   1. git status
   2. git ststus –s 縮短
   3. git diff
   4. git difftool --tool=vimdiff
      1. 如果跳出詢問視窗輸入y
      2. :q可以跳出編輯畫面 🡺 真的跳不出去，可以點一下垃圾桶結束並且重啟
   5. git add server.txt
   6. git status
8. 將幫「暫存區」的資料打包並且下提交前的註解
   1. git commit -m '第一次更新上傳'
   2. 這時候發現還有一個檔案忘記打包?
      1. git add new2.txt
      2. 加入到前一次的commit
         1. git commit --amend --no-edit
         2. git commit --amend -m 'New commit message'
9. 實際將檔案更新至倉儲
   1. git push origin master
   2. 到GitHub網頁上查看
10. 挑選不必要列管的檔案
    1. 新增一個 .gitignore 檔案
    2. 在 .gitignore 檔案中加入：\*.bak
    3. 修改\*.bak，執行git查看資料夾現況
    4. 在 .gitignore 檔案中加入：\*.xlsx
    5. 再次查看