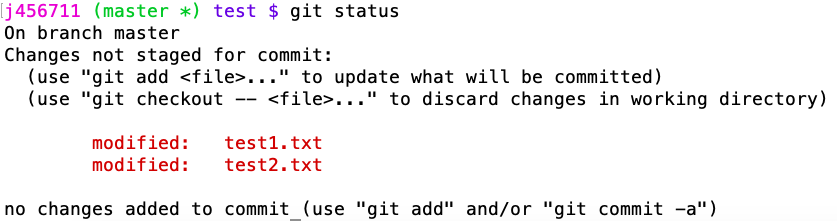
---------------------------

Assignment 1: Git & GitHub

--------------------------

//2

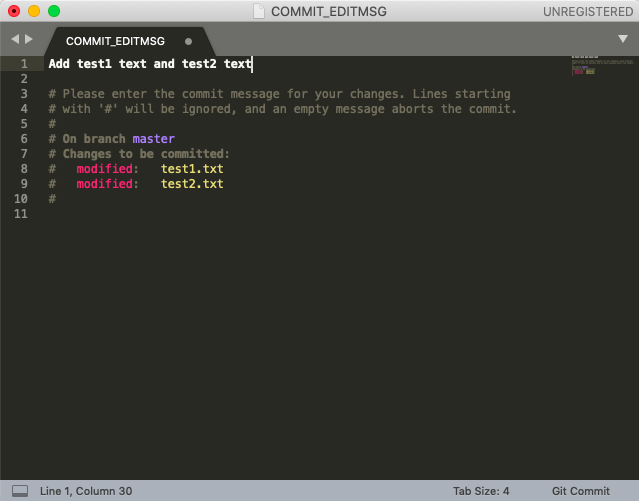
1. git status: 可以顯示在當下的branch中，從上一次的commit過後到現在，這之間有哪些檔案經過修改，而且這些修改尚未commit。如下圖，檔案test1.txt和test2.txt經過了修改，不過尚未commit，就會顯示出來。

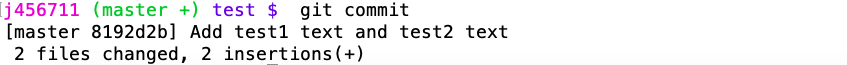


1. git add: 當檔案做了修改後，使用這個指令可以將檔案先存入staging area，可以一次add多個檔案。如下圖，add了兩個檔案。

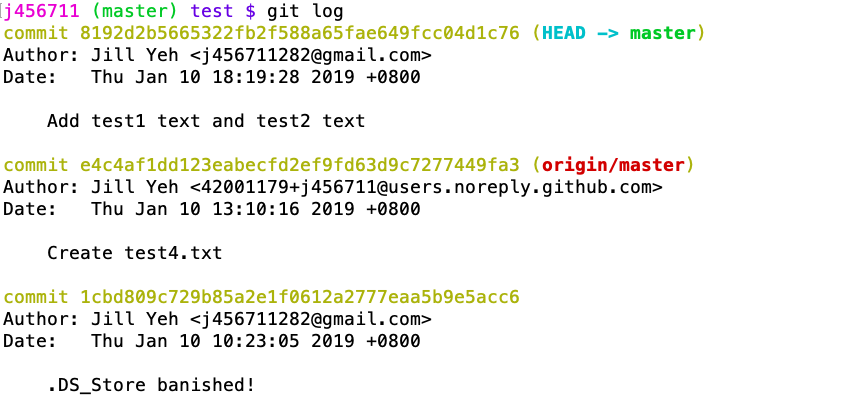


1. git commit: 會將檔案正式存入repository，並產生一組commit ID。有多個檔案時，會一起寫commit messages，作為一次commit。如下圖(一)，針對檔案先寫commit messages，之後就會跑出圖(二)commit的結果。

 (圖一)

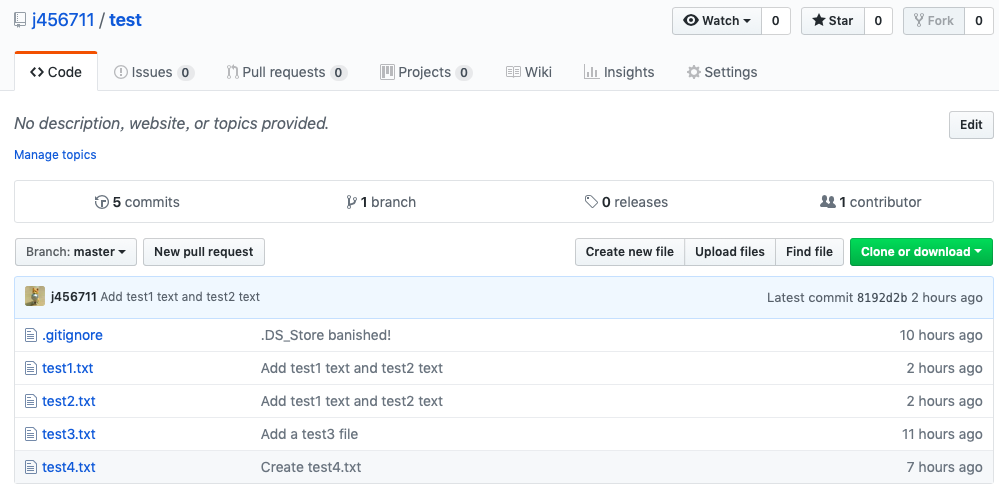
(圖二)

1. git log: 這個指令可以顯示在repository中，曾經commit過的commits，會從最近一次修改的commits開始顯示。如下圖，會顯示不同時間的commits，最上面的ID開頭8192為上一題中所送出的commit。

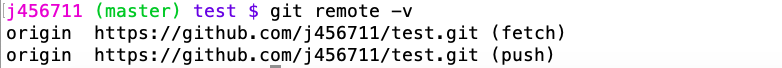


1. git push [Repo\_name] [Branch\_name]：這個指令會將目前在local的檔案存入到remote上，Repo\_name是檔案將存入的remote名稱(通常預設為origin)，Branch\_name是local分支的名稱。如下圖(一)，先前已經設定好origin代表的URL，而master則是代表local分支。push後可以看到圖(二)，在GitHub上成功上傳了檔案。

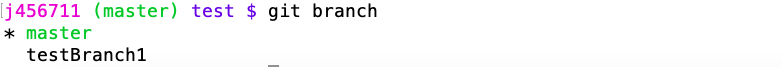
 (圖一)

 (圖二)

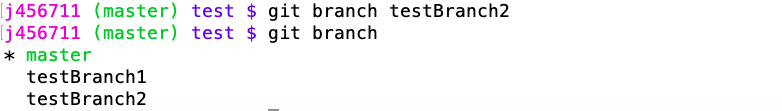
1. git remote -v：這個指令會顯示remote取得檔案及推送檔案的詳細URL位置。如下圖。



1. git branch：如果單純只下這個指令，可以檢查總共有哪些分支，以及目前位在哪個分支。如下圖，總共有兩個分支，\*字標示的是目前所在的分支。



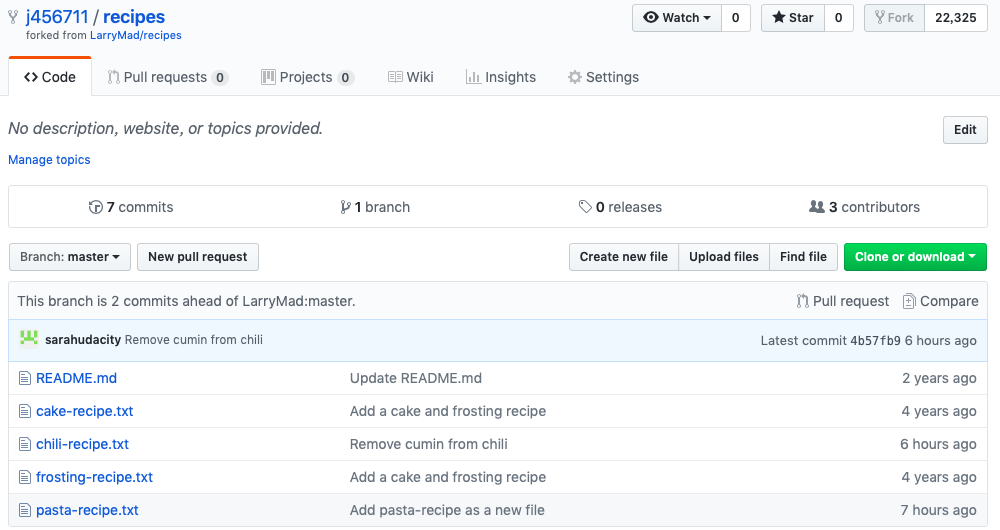
git branch branchName：這個指令可以新增分支。如下圖，新增了testBranch2：

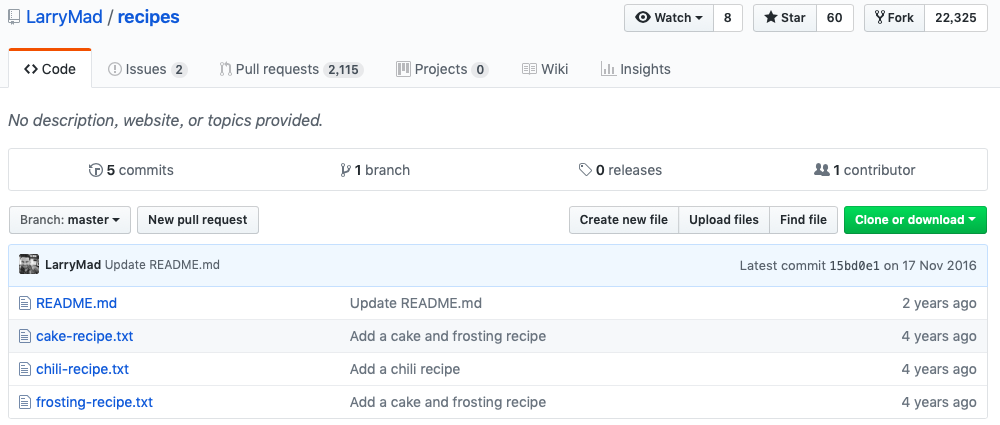


git branch -d branchName：這個指令可以刪除分支。如下圖，刪除了testBranch2：



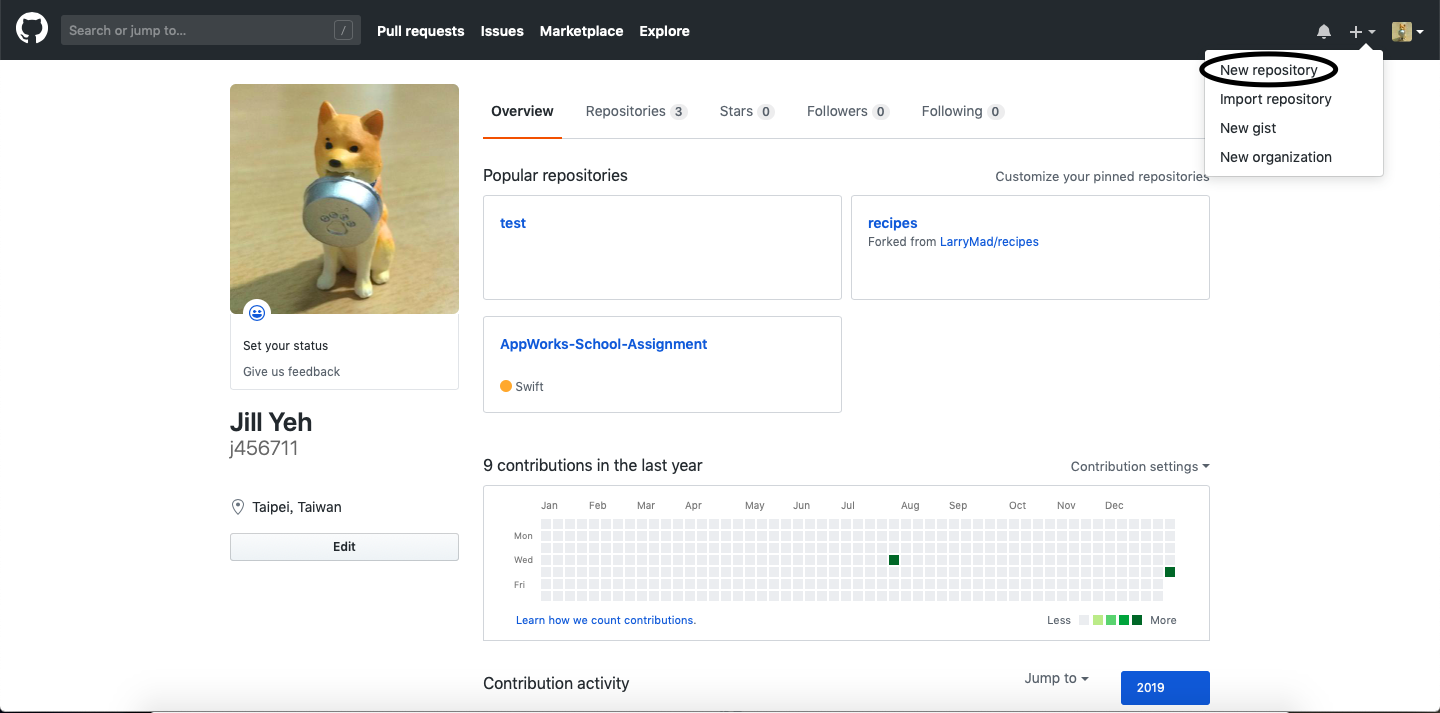
1. fork：GitHub會複製一份別人的專案到我們自己這邊，透過這個方式，我們對這個專案副本可以任意更動，而不會影響到原本的專案。另外也可以對原本的專案建議一些變更，讓原始擁有者可以得到回饋；原始擁有者也可以看到有多少人fork他的專案。如下圖(一)，我fork了一個別人的recipes專案，並且對專案進行了變更(新增了最底下的pasta-recipe.txt)。不過原始擁有者的專案，並沒有被我改變，如下圖(二)。

 (圖一)

 (圖二)

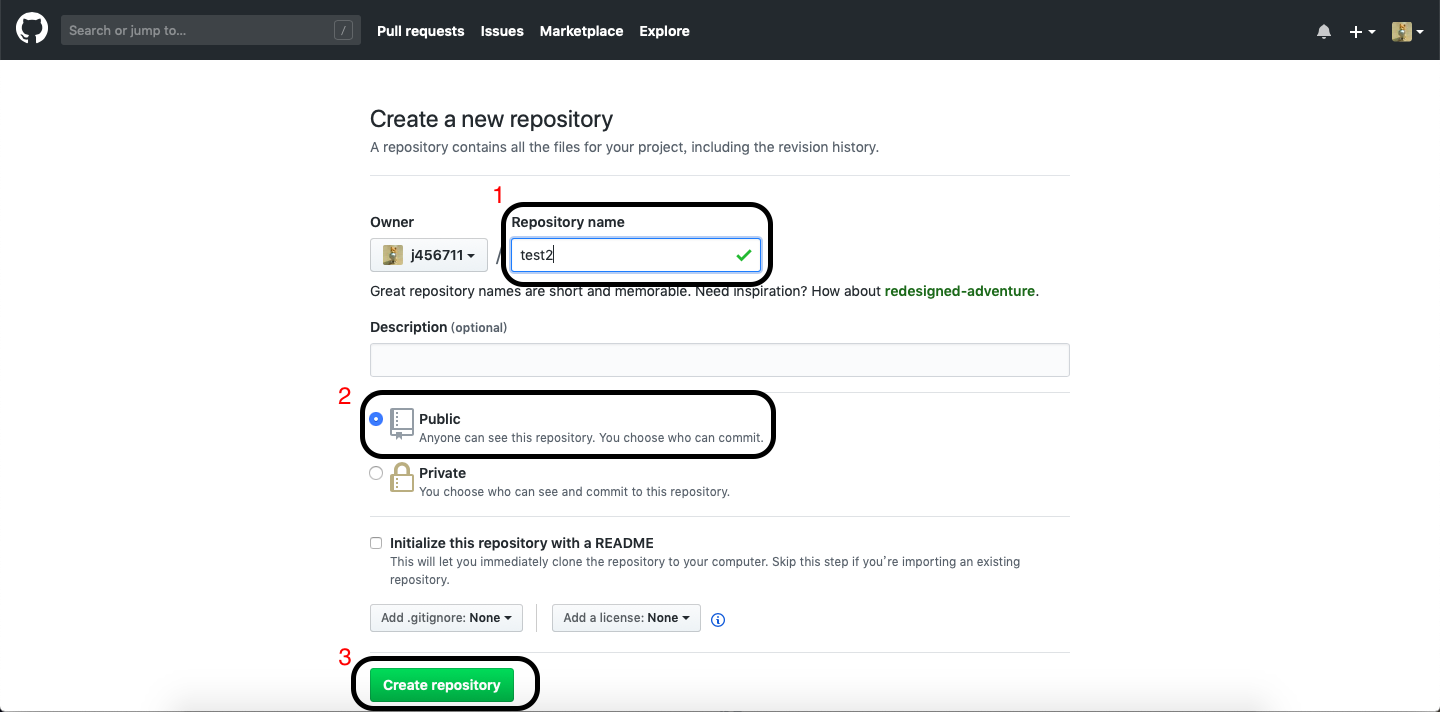
//3

1. 首先進入到GitHub，並在右上角的＋選單中，選擇New repository。

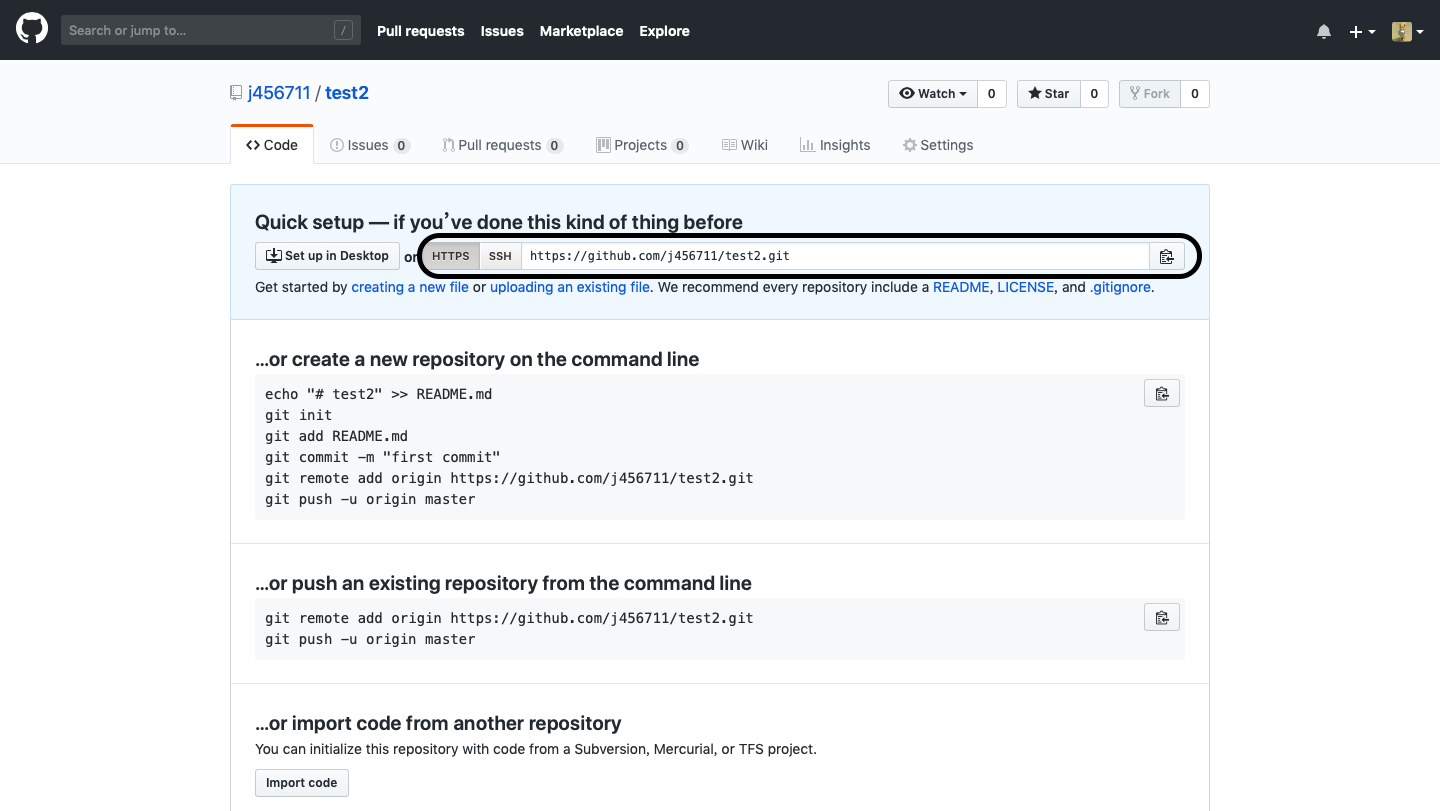


1. 進入到Create a new repository的頁面後，第一步填入專案名稱test2，第二步將專案設為公開，第三步按下Create repository。

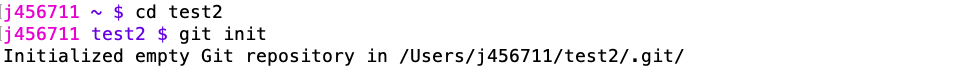
關於Initialize this repository with a README選項，如果我們要在這個專案內輸入既有的repository的話，就不需要勾選這個選項。



1. 接著會出現如下的畫面，複製黑色框中的URL，這個代表稍後要創建的remote的位置。

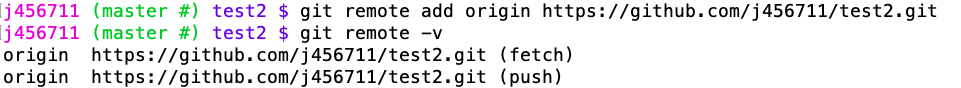


1. 現在切換到終端機。先進入到local的test2資料夾內，並使用git init創建一個空的repository。

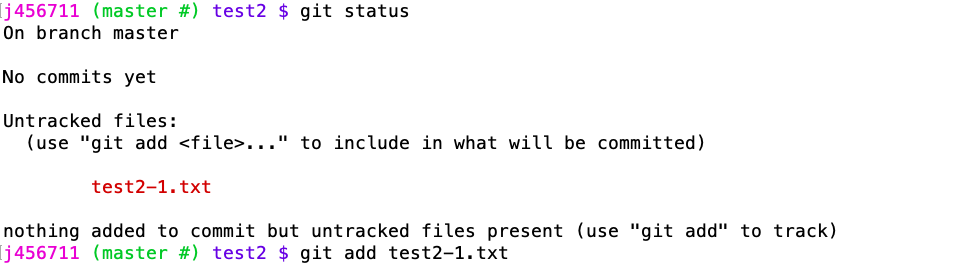


1. 再來要創建remote，使用git remote add，並將remote命名為origin，再指定它的路徑。路徑為剛才從GitHub上複製的URL。

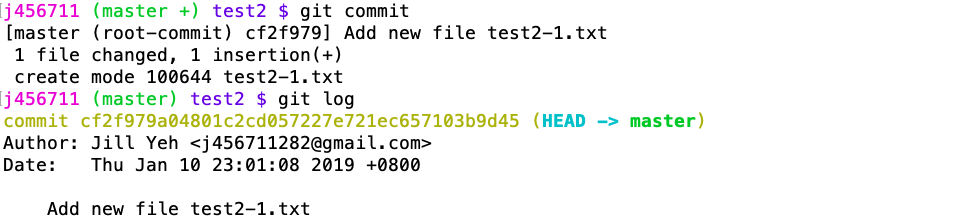
新增好路徑後，使用git remote -v檢查路徑是否正確。



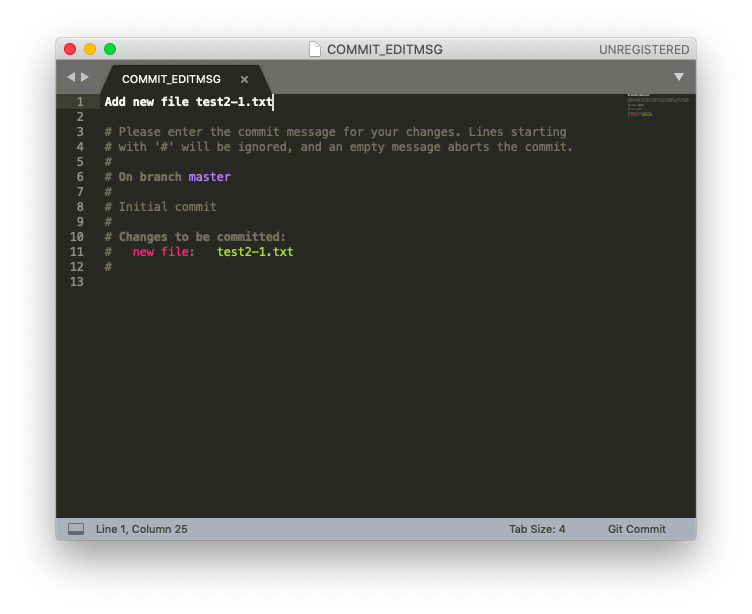
1. 此時，利用git status檢查一下目前的分支中的檔案狀態，狀態顯示No commits yet，也就是尚未有檔案commit的紀錄。因此現在開始上傳test2-1.txt，先利用git add將此檔案存入staging area。



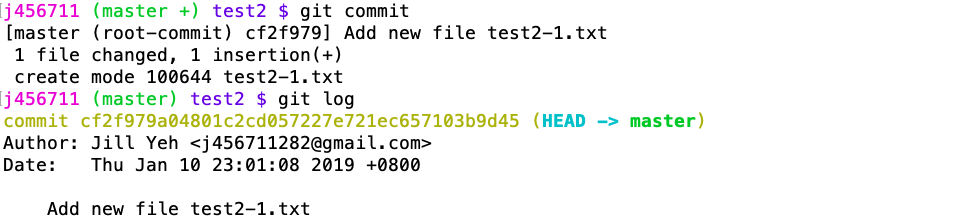
1. 接著，利用git commit來將檔案加入到repository。



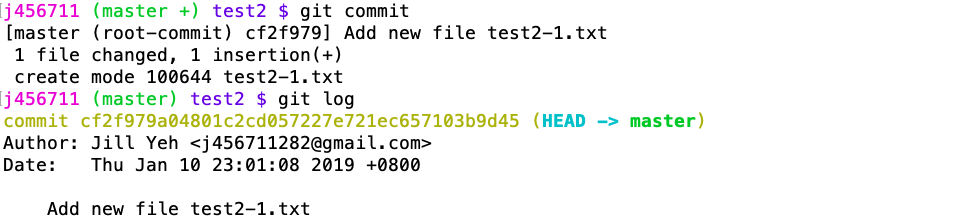
加上commit message: Add new file test2-1.txt。



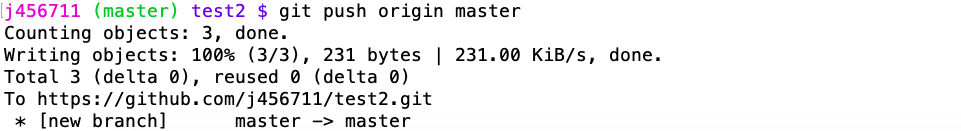
終端機會跑出以下結果：



再利用git log檢查commit的狀態，可以看到已經成功commit，並產生一組commit ID了。



1. 成功commit後，就可以利用git push加上remote名稱origin和local分支master，將檔案推送到GitHub上。



1. 最後重新整理GitHub網頁，就可以看到剛才push的檔案test2-1.txt出現在GitHub中了。