

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。

```
j456711 (master *) test $ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   test1.txt
        modified:   test2.txt

no changes added to commit_ (use "git add" and/or "git commit -a")
```

2. **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。

```
j456711 (master *) test $ git add test1.txt
j456711 (master *) test $ git add test2.txt
```

3. **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)

```
j456711 (master +) test $ git commit
[master 8192d2b] Add test1 text and test2 text
2 files changed, 2 insertions(+)
```

(圖二)

4. **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。

```
j456711 (master) test $ git log
commit 8192d2b5665322fb2f588a65fae649fcc04d1c76 (HEAD -> master)
Author: Jill Yeh <j456711282@gmail.com>
Date: Thu Jan 10 18:19:28 2019 +0800

    Add test1 text and test2 text

commit e4c4af1dd123eabecfd2ef9fd63d9c7277449fa3 (origin/master)
Author: Jill Yeh <42001179+j456711@users.noreply.github.com>
Date: Thu Jan 10 13:10:16 2019 +0800

    Create test4.txt

commit 1cbd809c729b85a2e1f0612a2777eaa5b9e5acc6
Author: Jill Yeh <j456711282@gmail.com>
Date: Thu Jan 10 10:23:05 2019 +0800

    .DS_Store banished!
```

5. **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。

```
j456711 (master) test $ git push origin master
Counting objects: 4, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (4/4), 392 bytes | 392.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0)
To https://github.com/j456711/test.git
e4c4af1..8192d2b master -> master
```

(圖一)

The screenshot shows the GitHub interface for the repository 'j456711/test'. At the top, there are buttons for Watch (0), Star (0), and Fork (0). Below this is a navigation bar with links for Code, Issues (0), Pull requests (0), Projects (0), Wiki, Insights, and Settings. A message states 'No description, website, or topics provided.' with an 'Edit' button. Below this, statistics show '5 commits', '1 branch', '0 releases', and '1 contributor'. There are buttons for 'Branch: master', 'New pull request', 'Create new file', 'Upload files', 'Find file', and 'Clone or download'. The commit history is displayed as a table:

Commit Hash	Message	Time Ago
j456711	Add test1 text and test2 text	Latest commit 8192d2b 2 hours ago
.gitignore	.DS_Store banished!	10 hours ago
test1.txt	Add test1 text and test2 text	2 hours ago
test2.txt	Add test1 text and test2 text	2 hours ago
test3.txt	Add a test3 file	11 hours ago
test4.txt	Create test4.txt	7 hours ago

(圖二)

6. **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。

```
j456711 (master) test $ git remote -v
origin https://github.com/j456711/test.git (fetch)
origin https://github.com/j456711/test.git (push)
```

7. **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。

```
j456711 (master) test $ git branch
* master
  testBranch1
```

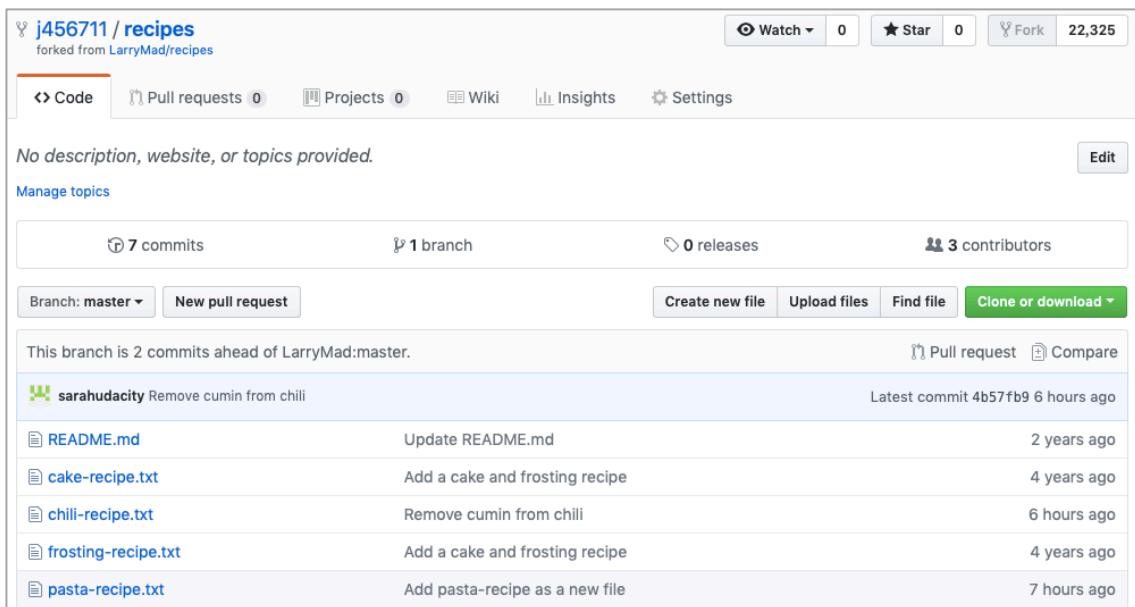
git branch branchName：這個指令可以新增分支。如下圖，新增了testBranch2：

```
j456711 (master) test $ git branch testBranch2
j456711 (master) test $ git branch
* master
  testBranch1
  testBranch2
```

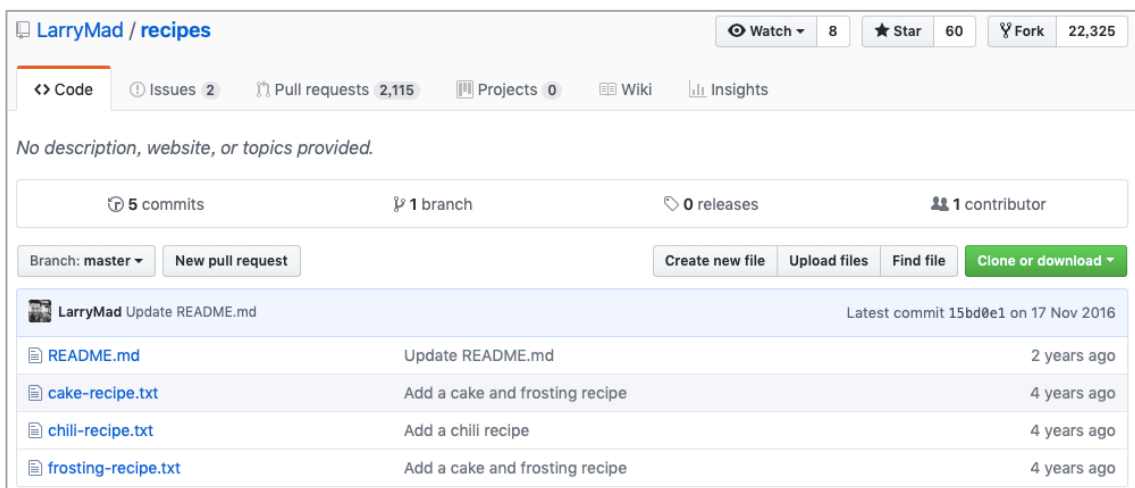
git branch -d branchName：這個指令可以刪除分支。如下圖，刪除了testBranch2：

```
j456711 (master) test $ git branch -d testBranch2
Deleted branch testBranch2 (was 8192d2b).
```

8. **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）。



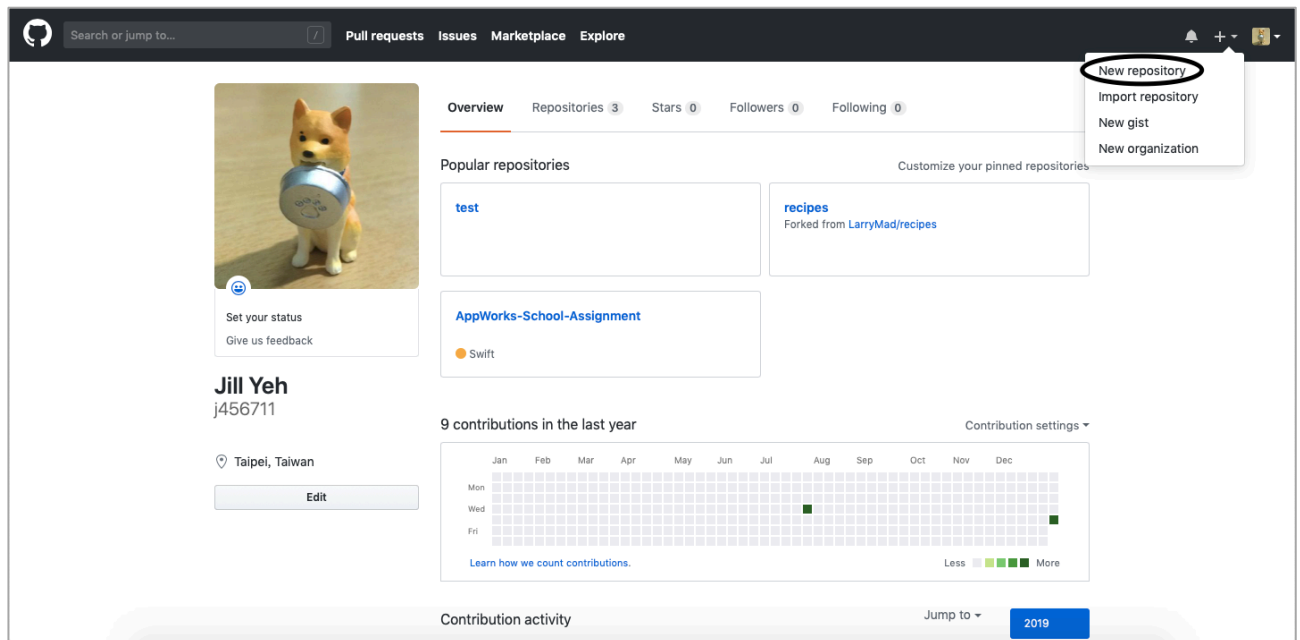
（圖一）



（圖二）

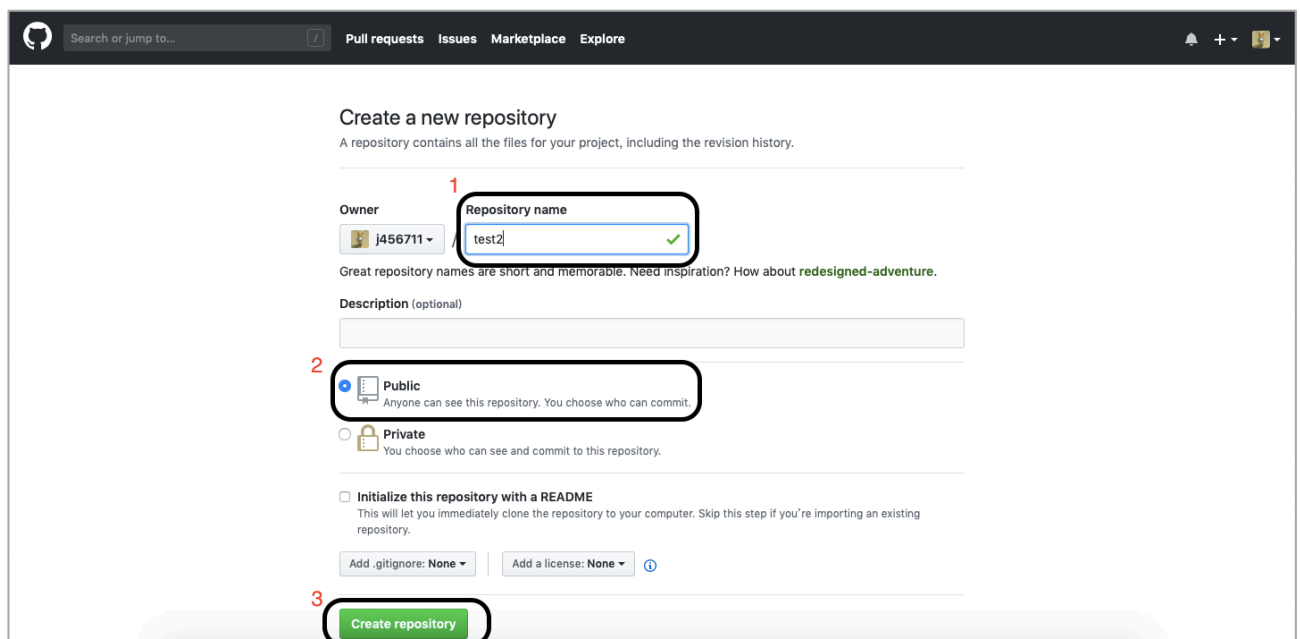
//3

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

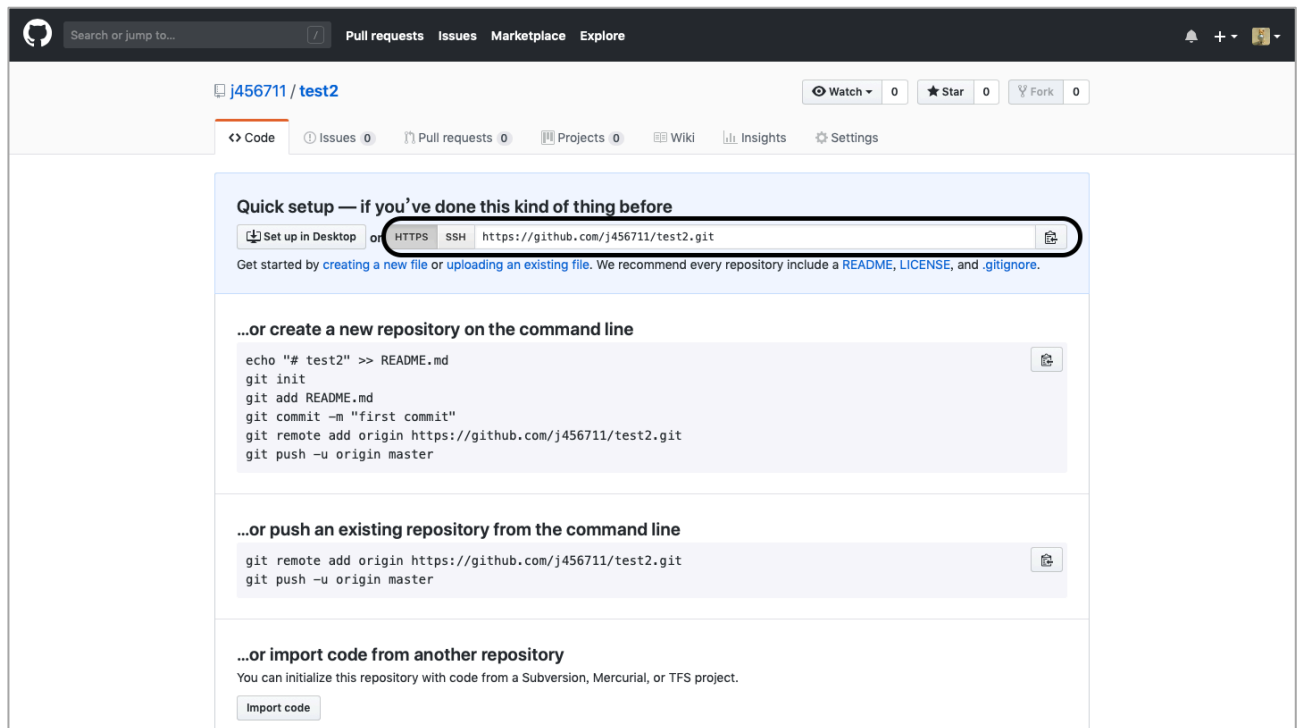


2. 進入到Create a new repository的頁面後，第一步填入專案名稱test2，第二步將專案設為公開，第三步按下Create repository。

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。



3. 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。



4. 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。

```
j456711 ~ $ cd test2
j456711 test2 $ git init
Initialized empty Git repository in /Users/j456711/test2/.git/
```

5. 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。

```
j456711 (master #) test2 $ git remote add origin https://github.com/j456711/test2.git
j456711 (master #) test2 $ git remote -v
origin https://github.com/j456711/test2.git (fetch)
origin https://github.com/j456711/test2.git (push)
```

6. 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。

```
j456711 (master #) test2 $ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)

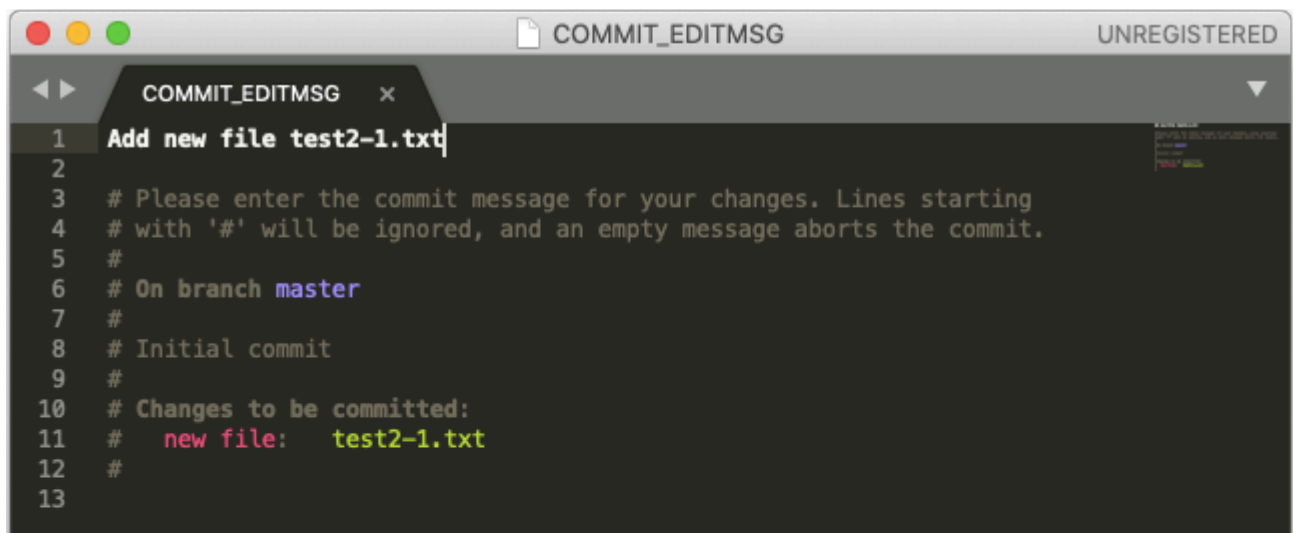
    test2-1.txt

nothing added to commit but untracked files present (use "git add" to track)
j456711 (master #) test2 $ git add test2-1.txt
```

7. 接著，利用`git commit`來將檔案加入到repository。

```
j456711 (master +) test2 $ git commit
```

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



```
COMMIT_EDITMSG x
1 Add new file test2-1.txt
2
3 # Please enter the commit message for your changes. Lines starting
4 # with '#' will be ignored, and an empty message aborts the commit.
5 #
6 # On branch master
7 #
8 # Initial commit
9 #
10 # Changes to be committed:
11 #   new file:   test2-1.txt
12 #
13
```

終端機會跑出以下結果：

```
[master (root-commit) cf2f979] Add new file test2-1.txt
1 file changed, 1 insertion(+)
create mode 100644 test2-1.txt
```

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。

```
j456711 (master) test2 $ git log
commit cf2f979a04801c2cd057227e721ec657103b9d45 (HEAD -> master)
Author: Jill Yeh <j456711282@gmail.com>
Date: Thu Jan 10 23:01:08 2019 +0800

    Add new file test2-1.txt
```

8. 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。

```
j456711 (master) test2 $ git push origin master
Counting objects: 3, done.
Writing objects: 100% (3/3), 231 bytes | 231.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/j456711/test2.git
* [new branch]      master -> master
```

9. 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。

