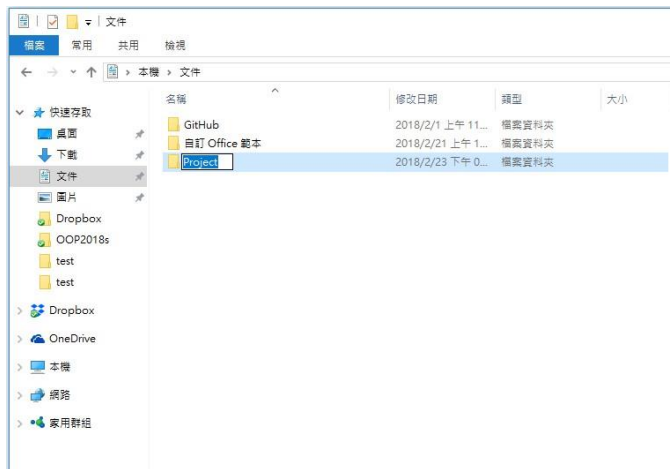
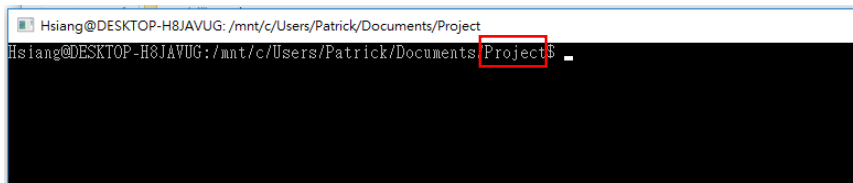


## 作業上傳教學

### 1. 在電腦上建立自己的專案資料夾



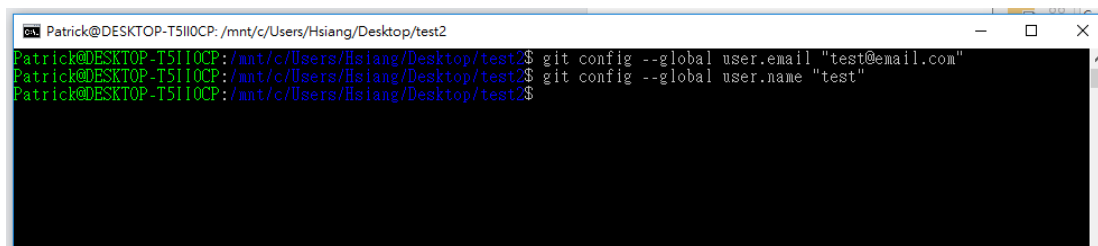
### 2. 在專案目錄下打開 終端機(ubuntu bash)



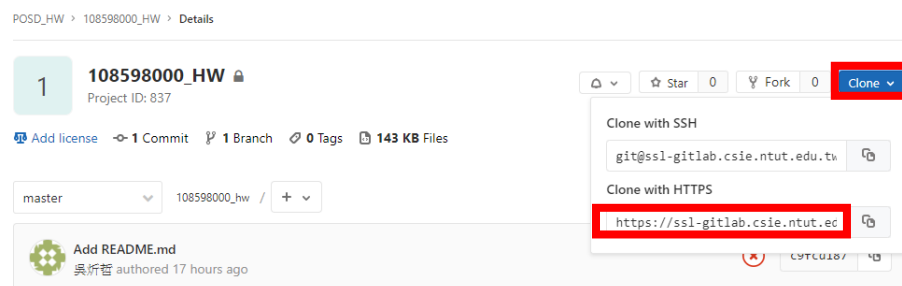
### 3. 設定 git 基本資料(僅系統第一次使用 git 時設定)。

```
git config --global user.email "${EMAIL}"
git config --global user.name "${NAME}"
```

把`${EMAIL}` 及 `${NAME}` 換成個人的 email 及 使用者名稱



### 4. 前往 Gitlab (<https://ssl-gitlab.csie.ntut.edu.tw>) 並找到自己的專案連結 點選clone後 選擇 HTTPS 複製URL



5. 輸入指令: `git clone ${{Your URL}}`

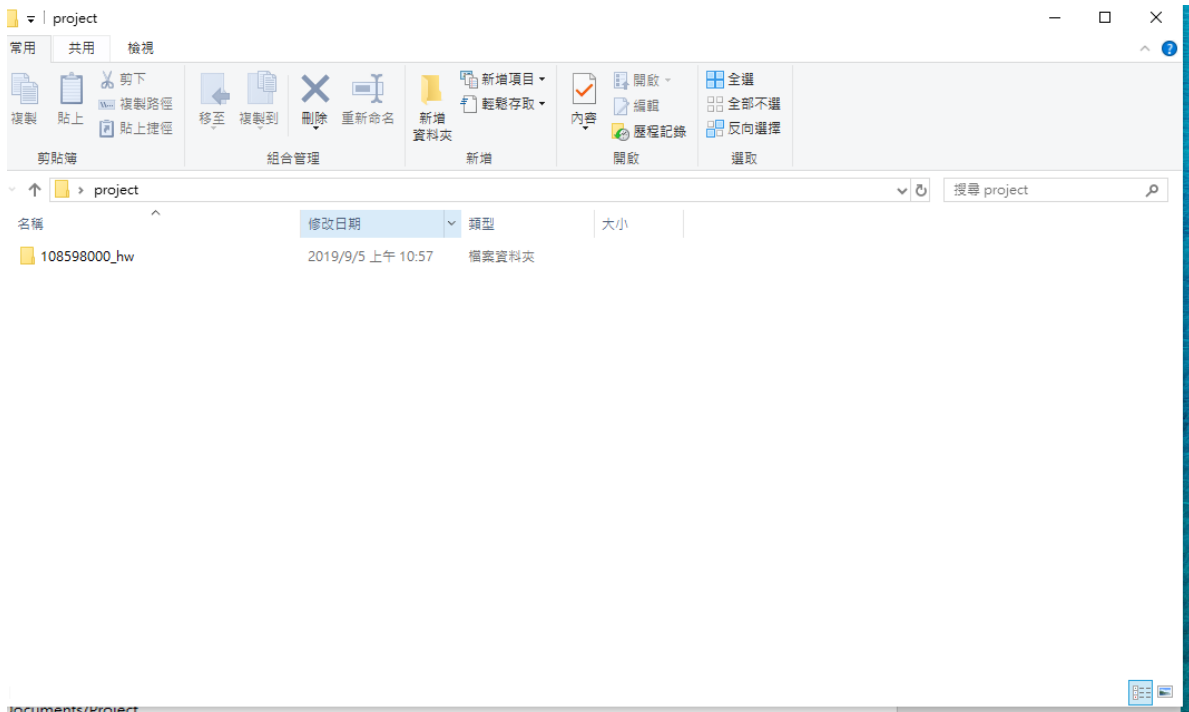
- `${{Your URL}}` 用上面複製的專案連結取代

```
C:\Users\Lab1321\Desktop\project>bash
lab1321@DESKTOP-2VQUNMM:/mnt/c/Users/Lab1321/Desktop/project$ git clone https://ssl-gitlab.csie.ntut.edu.tw/posd_hw/108598000_hw.git
```

Enter之後輸入你的帳號密碼

```
lab1321@DESKTOP-2VQUNMM:/mnt/c/Users/Lab1321/Desktop/project$ git clone https://ssl-gitlab.csie.ntut.edu.tw/posd_hw/108598000_hw.git
Cloning into '108598000_hw'...
Username for 'https://ssl-gitlab.csie.ntut.edu.tw': 108598000
Password for 'https://108598000@ssl-gitlab.csie.ntut.edu.tw':
```

完成之後



你就可以看到你的作業專案

6. 確認專案當前的狀態。

輸入指令: `cd ${{學號}}_hw`

`git status`

```
lab1321@DESKTOP-2VQUNMM:/mnt/c/Users/Lab1321/Desktop/project/108598000_hw$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
```

因為目前沒做任何變更 所以會看到上圖

7. 以下操作皆要在你的專案資料夾`${{學號}}_HW`下進行

8. 在專案內有新增檔案或修改檔案時, 要將 新增/變更 的檔案加入準備 commit 的佇列。輸入指令: `git add .`

```
Hsiang@DESKTOP-H8JAVUG: /mnt/c/Users/Patrick/Documents/Project
Hsiang@DESKTOP-H8JAVUG:/mnt/c/Users/Patrick/Documents/Project$ git add .
```

- 如有移除的檔案則變為 `git rm ${{已刪除的檔案}}`

```
Hsiang@DESKTOP-H8JAVUG: /mnt/c/Users/Patrick/Documents/Project
Hsiang@DESKTOP-H8JAVUG:/mnt/c/Users/Patrick/Documents/Project$ git rm test1.cpp
rm 'test1.cpp'
Hsiang@DESKTOP-H8JAVUG:/mnt/c/Users/Patrick/Documents/Project$ _
```

9. 再次確認檔案的變更狀態。輸入指令: `git status`

```
Hsiang@DESKTOP-H8JAVUG: /mnt/c/Users/Patrick/Documents/Project
Hsiang@DESKTOP-H8JAVUG:/mnt/c/Users/Patrick/Documents/Project$ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        renamed:   test1.cpp -> test2.cpp

Hsiang@DESKTOP-H8JAVUG:/mnt/c/Users/Patrick/Documents/Project$
```

10. 確認無誤後，將這次變更 `commit` 到本地端版控中。輸入指令: `git commit`

– m “\${{Your message}}”

```
Hsiang@DESKTOP-H8JAVUG: /mnt/c/Users/Patrick/Documents/Project
Hsiang@DESKTOP-H8JAVUG:/mnt/c/Users/Patrick/Documents/Project$ git commit -m "Commit testing"
[master 0deded9] Commit testing
1 file changed, 0 insertions(+), 0 deletions(-)
rename test1.cpp => test2.cpp (100%)
Hsiang@DESKTOP-H8JAVUG:/mnt/c/Users/Patrick/Documents/Project$ _
```

11. 將本地端的版本紀錄上傳至 GitLab。輸入指令: `git push origin master`  
輸入你的 Gitlab 的 Username 及 Password

\* Password 在輸入時不會顯示出任何字

```
Hsiang@DESKTOP-H8JAVUG: /mnt/c/Users/Patrick/Documents/Project
Hsiang@DESKTOP-H8JAVUG:/mnt/c/Users/Patrick/Documents/Project$ git push origin master
Username for 'http://140.124.181.136:81': 106820000
Password for 'http://106820000@140.124.181.136:81': _
```

## 12. 回到 GitLab 確認自己的作業

框內為新增的檔案

POSD\_HW > 108598000\_HW > Details

1 **108598000\_HW** Project ID: 837

Star 0 Fork 0 Clone

Add license 1 Commit 1 Branch 0 Tags 143 KB Files

master 108598000\_hw / History Find file Web IDE

test 吳祈哲 authored 1 minute ago ad45135c

README Add CHANGELOG Add CONTRIBUTING Enable Auto DevOps Add Kubernetes cluster

Set up CI/CD

Name	Last commit	Last update
README.md	Add README.md	18 hours ago
test.cpp	test	1 minute ago

README.md

test

## 13. 上傳到 Jenkins 查看自己的作業狀態

會有兩個專案，通過的話會亮藍色，沒通過會亮紅色

Jenkins

108598000 | 登出

開啟自動更新頁面

posd2019f\_HW posd2019f\_HW\_TA 全部

S	W	名稱	上次成功時間	上次失敗時間	上次建置花費時間
		posd2019f_108598000_HW	18 時 - #7	3 分 3 秒 - #12	2 秒
		posd2019f_108598000_HW_TA	18 時 - #7	無	2.1 秒

顯示: S M L

圖例: RSS 提醒-全部 RSS 提醒-成功 RSS 提醒-最近發生建置

建置佇列

你列中沒有建置作業。

建置執行歷史狀態

- 1 建置
- 2 建置