

# How to push file to GitHub

Simple Version  
based on 洪志菁的講義

written by NZ



**#Zero**

**before start**

**download** git

**download** Visual Studio Code

**register** GitHub

<https://git-scm.com/downloads>

<https://code.visualstudio.com/download>

<https://github.com/>

**等到  
45分**

**#First**

**use Git Bash**

# dictionary 1

程式碼	意義
pwd	顯示當前資料夾位置
ls	列出當前位置中所有的資料夾
mkdir 資料夾名稱	創建新資料夾
cd 你要去的資料夾	移到要去的資料夾中
git init	初始化git
touch name.xx	新增.xx檔案

## step 1: 設定使用者的 Email 信箱以及使用者名稱

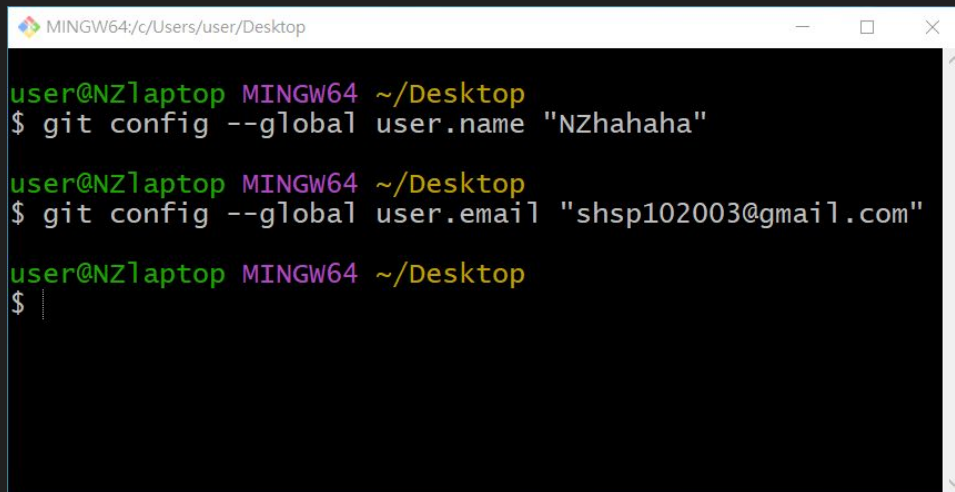
打開git bash

打入

git config --global user.name "你的名稱"

git config --global user.email "你的email"

(用註冊 github 的資訊)

A screenshot of a terminal window titled 'MINGW64: c:/Users/user/Desktop'. The terminal shows three lines of text: the first line is the prompt 'user@NZlaptop MINGW64 ~/Desktop' followed by the command '\$ git config --global user.name "NZhahaha"'; the second line is the prompt 'user@NZlaptop MINGW64 ~/Desktop' followed by the command '\$ git config --global user.email "shsp102003@gmail.com"'; and the third line is the prompt 'user@NZlaptop MINGW64 ~/Desktop' followed by a dollar sign and a cursor, indicating the command prompt is ready for input.

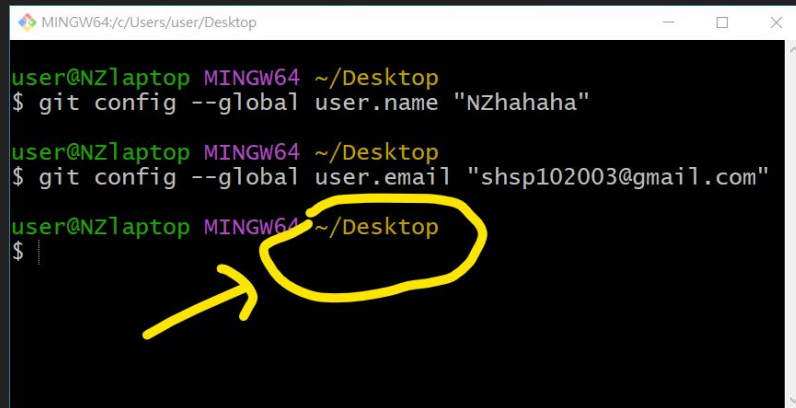
```
user@NZlaptop MINGW64 ~/Desktop
$ git config --global user.name "NZhahaha"

user@NZlaptop MINGW64 ~/Desktop
$ git config --global user.email "shsp102003@gmail.com"

user@NZlaptop MINGW64 ~/Desktop
$
```

## step 2: 確認自己在哪&創建資料夾&進入資料夾

從黃色圈圈處可以知道  
我們在桌面(Desktop)這個資料夾中



```
MINGW64/c/Users/user/Desktop
user@NZlaptop MINGW64 ~/Desktop
$ git config --global user.name "NZhahaha"

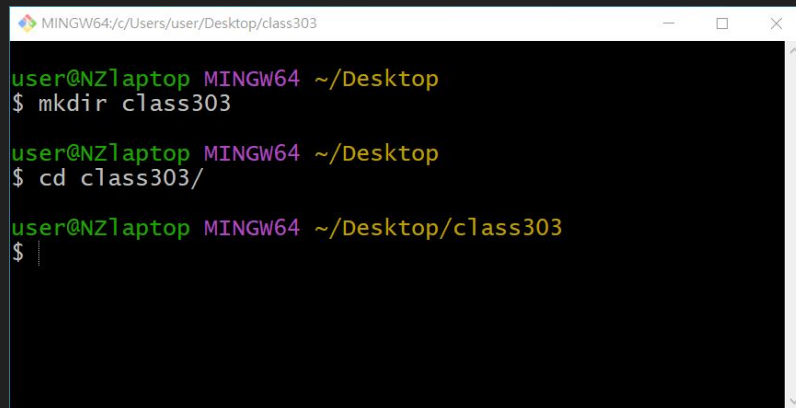
user@NZlaptop MINGW64 ~/Desktop
$ git config --global user.email "shsp102003@gmail.com"

user@NZlaptop MINGW64 ~/Desktop
$ |
```

A yellow circle is drawn around the text `~/Desktop` in the third prompt, with a yellow arrow pointing to it from the left.

那現在要在桌面創一個新資料夾  
請打  
**mkdir 資料夾名稱**

然後要移到資料夾裡面  
**cd 資料夾名稱**



```
MINGW64/c/Users/user/Desktop/class303
user@NZlaptop MINGW64 ~/Desktop
$ mkdir class303

user@NZlaptop MINGW64 ~/Desktop
$ cd class303/

user@NZlaptop MINGW64 ~/Desktop/class303
$ |
```

## step 3: 初始化git & 創一個新文件檔(.md == markdown)

初始化git請打

**git init**

會出現右邊第三四行那串

接著創一個新文件檔(.md)

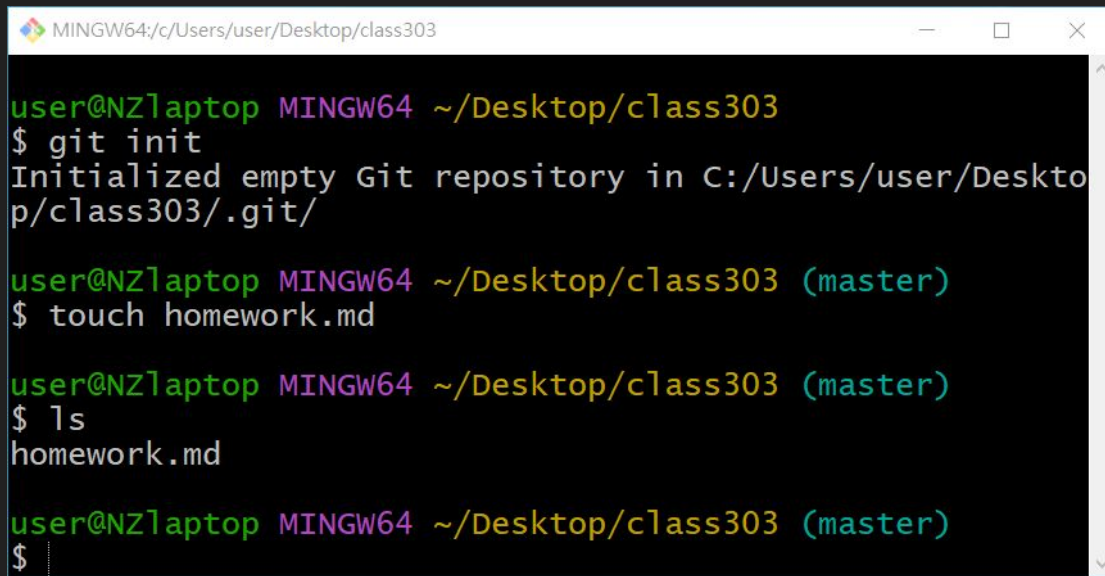
**touch** 想一個文件名稱.md

要確認有沒有成功

可以用

**ls**

列出資料夾中所有檔案

A screenshot of a terminal window titled 'MINGW64/c:/Users/user/Desktop/class303'. The terminal shows the following commands and output:  
1. Command: `git init`  
Output: `Initialized empty Git repository in C:/Users/user/Desktop/class303/.git/`  
2. Command: `touch homework.md`  
Output: (no output)  
3. Command: `ls`  
Output: `homework.md`  
The prompt is `user@NZlaptop MINGW64 ~/Desktop/class303 (master)`.

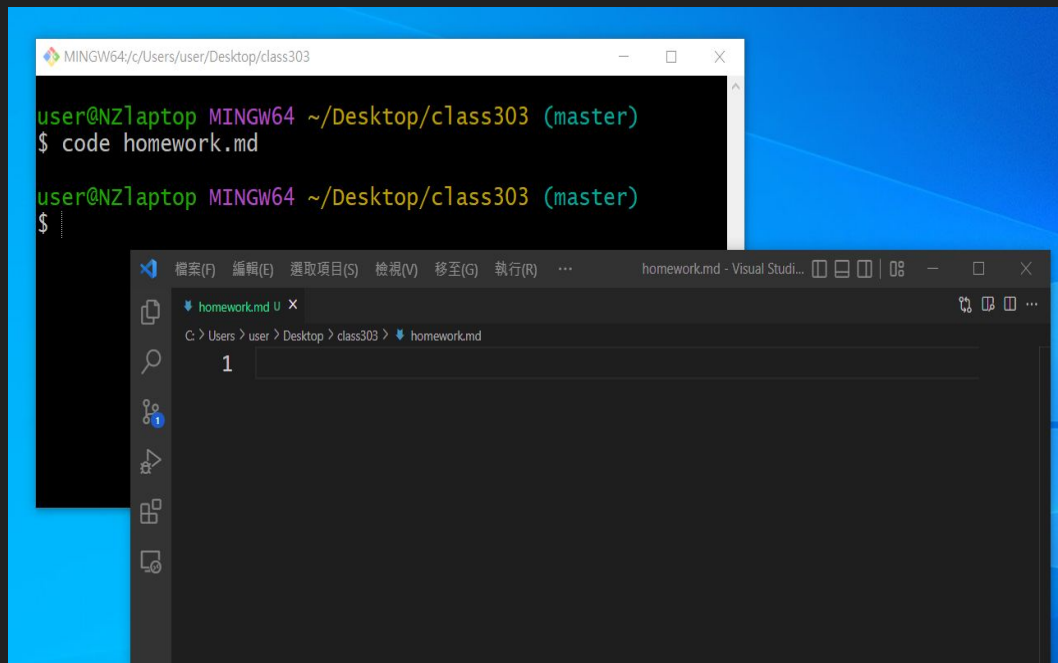


## step 4: 開始撰寫自我介紹

現在要打開剛剛的新文件檔  
請打

**code** 新文件檔名稱

如果跳出來的視窗有問問題  
能回答是就是  
(不然就自己讀英文XD)



**#Second**

**use Visual Studio Code**

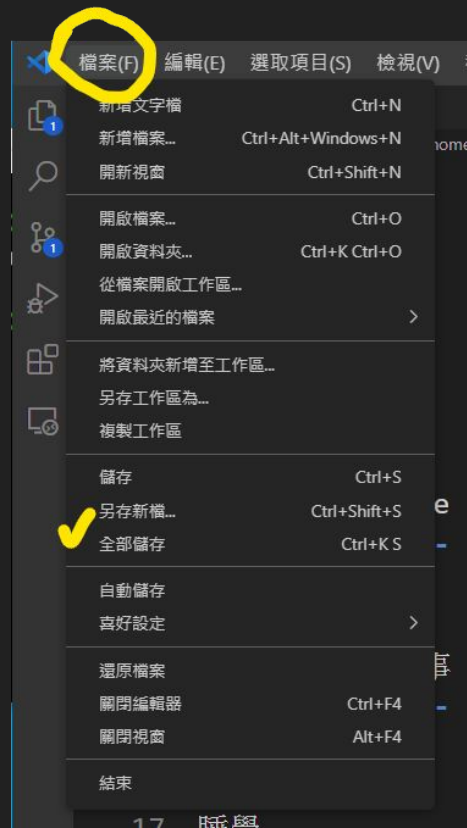
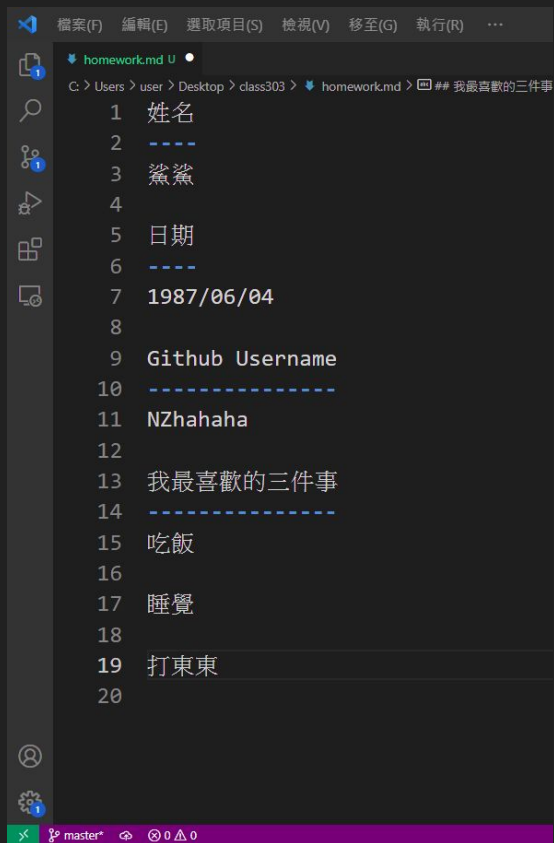
# step 1: 開始撰寫自我介紹

把文字貼上並修改

打完後按左上角進行儲存

英文介面: file>>save all

中文介面: 檔案>>全部儲存



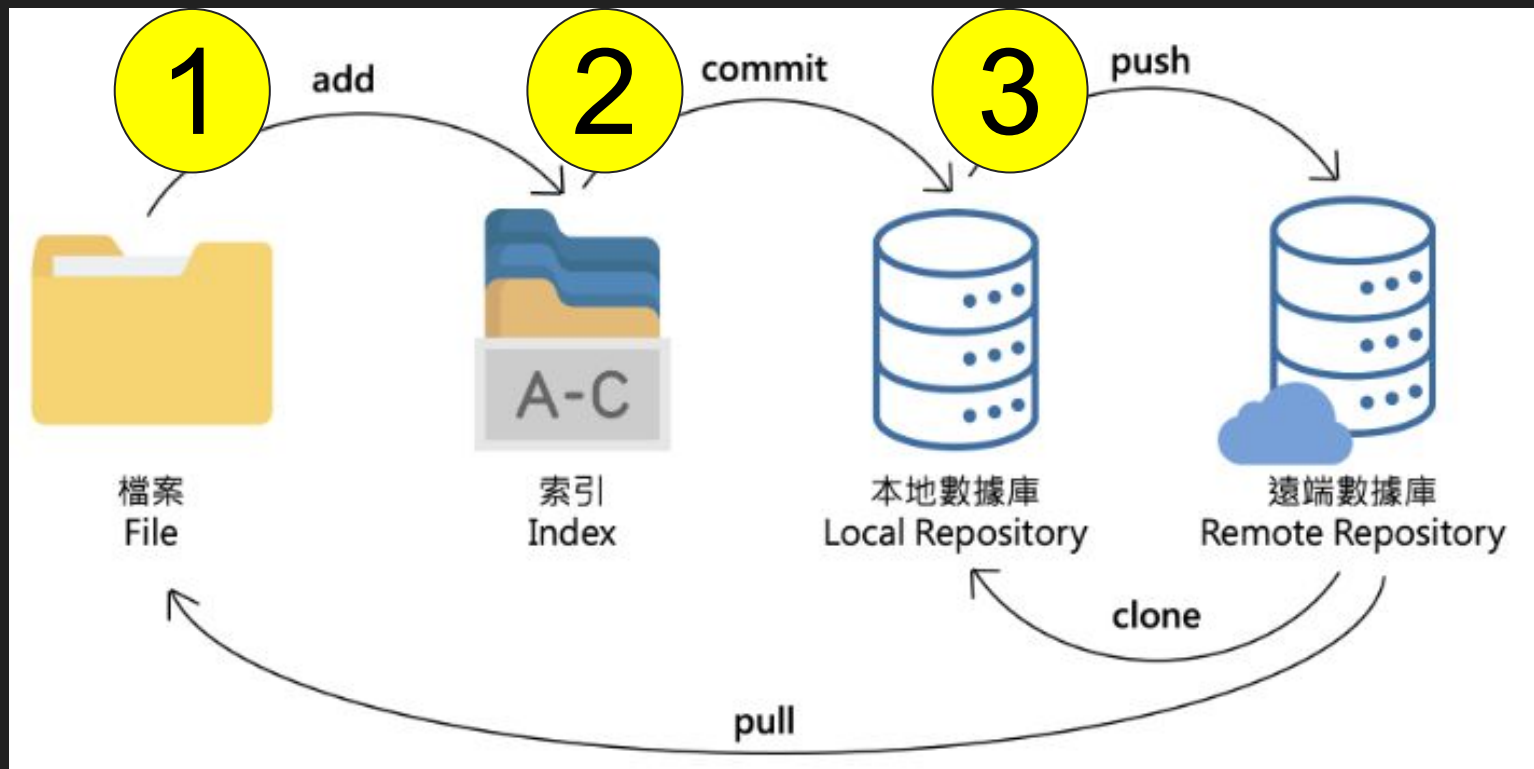
**#Third**

**use Git Bash**

# dictionary 2

程式碼	意義
git add 檔案名稱	把檔案加入暫存區
git status	檢查暫存區狀況
git commit -m "給暫存區名字"	儲存新版本&備註暫存區
git log	檢查本地數據庫狀況

# know-how

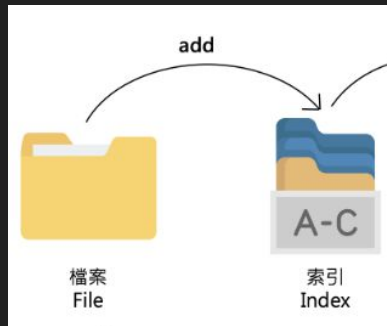


## step 1: 把文件檔上傳暫存區

將檔案傳至暫存區請打  
**git add 檔案名稱**

確認有無成功  
**git status**

可看到有綠色new file



```
MINGW64:/c/Users/user/Desktop/class303

user@NZlaptop MINGW64 ~/Desktop/class303 (master)
$ git add homework.md

user@NZlaptop MINGW64 ~/Desktop/class303 (master)
$ git status
On branch master

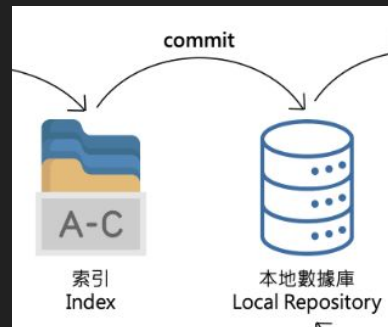
No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   homework.md

user@NZlaptop MINGW64 ~/Desktop/class303 (master)
$
```

## step 2: 把暫存區備註名稱並上傳本地儲存庫

將暫存區打包成一個版本並傳至本地儲存庫請打  
**git commit -m "備註版本名稱"**



確認有無成功

**git log**

可看到作者、時間、  
版本備註名稱

```
MINGW64/c/Users/user/Desktop/class303
user@NZlaptop MINGW64 ~/Desktop/class303 (master)
$ git commit -m "this is first version"
[master (root-commit) 3689d6b] this is first version
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 homework.md

user@NZlaptop MINGW64 ~/Desktop/class303 (master)
$ git log
commit 3689d6b0f174999c2e66d3a2ad32bf81347bd658 (HEAD ->
master)
Author: NZhahaha <shsp102003@gmail.com>
Date: Thu Sep 15 23:04:19 2022 +0800

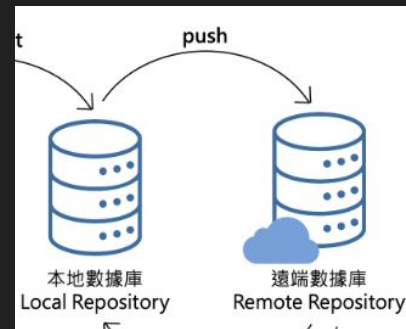
    this is first version

user@NZlaptop MINGW64 ~/Desktop/class303 (master)
$
```



## step 3: 把本地儲存庫傳至GitHub

將暫存區打包成一個版本並傳至本地儲存庫請打  
**git commit -m "備註版本名稱"**



確認有無成功

**git log**

可看到作者、時間、  
版本備註名稱

(也可用

**git status**

確認暫存區是空的)

```
MINGW64/c/Users/user/Desktop/class303
user@NZlaptop MINGW64 ~/Desktop/class303 (master)
$ git commit -m "this is first version"
[master (root-commit) 3689d6b] this is first version
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 homework.md

user@NZlaptop MINGW64 ~/Desktop/class303 (master)
$ git log
commit 3689d6b0f174999c2e66d3a2ad32bf81347bd658 (HEAD ->
master)
Author: NZhahaha <shsp102003@gmail.com>
Date: Thu Sep 15 23:04:19 2022 +0800

    this is first version

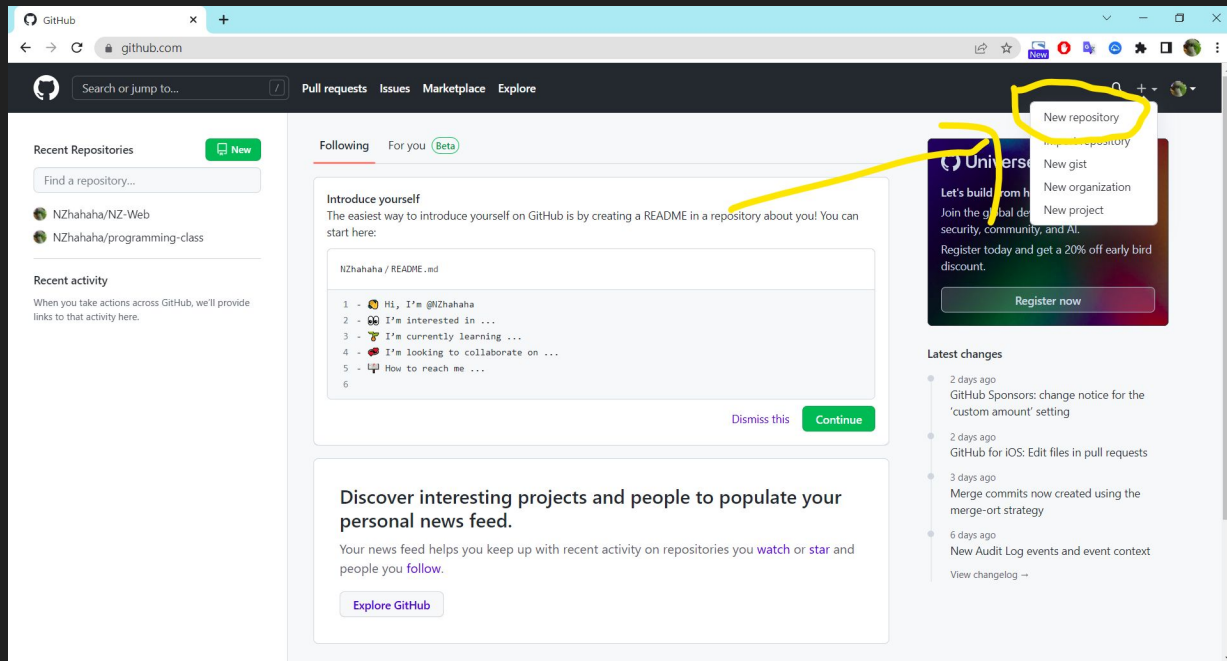
user@NZlaptop MINGW64 ~/Desktop/class303 (master)
$
```

**#Four**

**use GitHub**

# step 1: 在GitHub建立一個遠端儲存庫

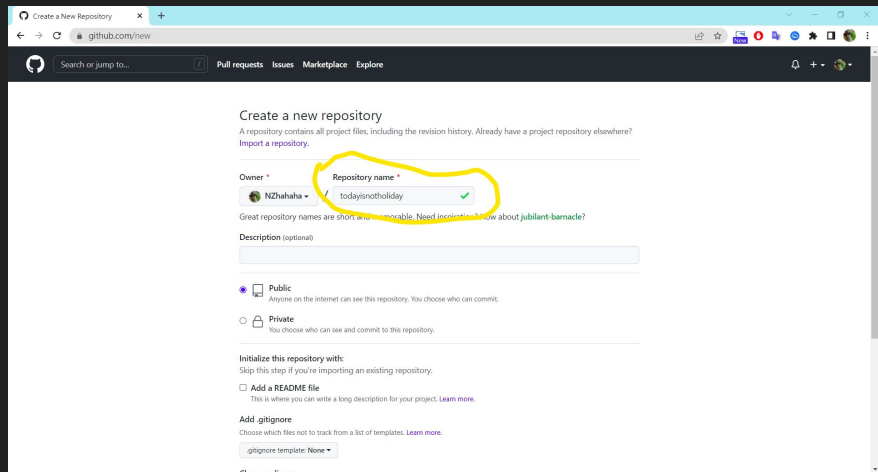
到GitHub網站  
點選右上角+  
點New repository



## step 2: 設定遠端資料庫

在Repository name  
命名你遠端資料庫名稱

往下滑按建立資料庫  
creat repository



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner \*  Repository name \*

Great repository names are short and memorable. Need inspiration? [How about jubilant-barnacle?](#)

Description (optional)

☒ Public  
Anyone on the internet can see this repository. You choose who can commit.

☐ Private  
You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

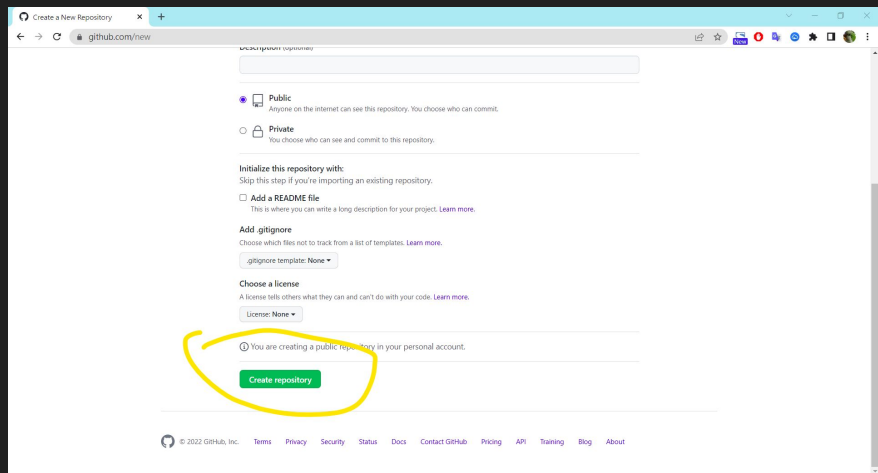
☐ Add a README file  
This is where you can write a long description for your project. [Learn more.](#)

Add gitignore

Choose which files not to track from a list of templates. [Learn more.](#)

gitignore template:

Choose a license



☒ Public  
Anyone on the internet can see this repository. You choose who can commit.

☐ Private  
You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

☐ Add a README file  
This is where you can write a long description for your project. [Learn more.](#)

Add gitignore

Choose which files not to track from a list of templates. [Learn more.](#)

gitignore template:

Choose a license

A license tells others what they can and can't do with your code. [Learn more.](#)

License:

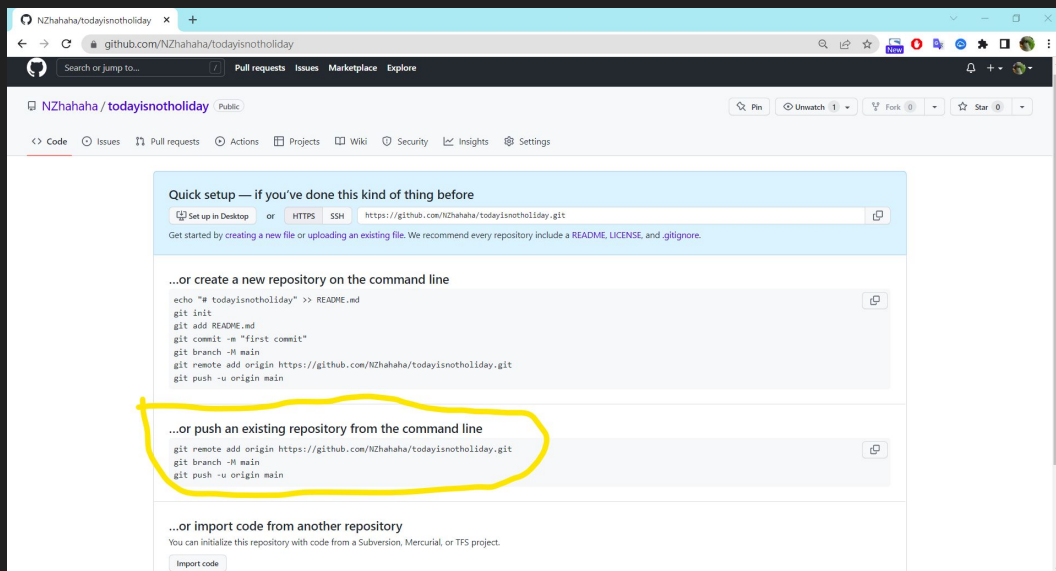
① You are creating a public repository in your personal account.

© 2022 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Docs](#) [Contact GitHub](#) [Pricing](#) [API](#) [Training](#) [Blog](#) [About](#)

## step 3: 找到GitHub幫你生成的指令

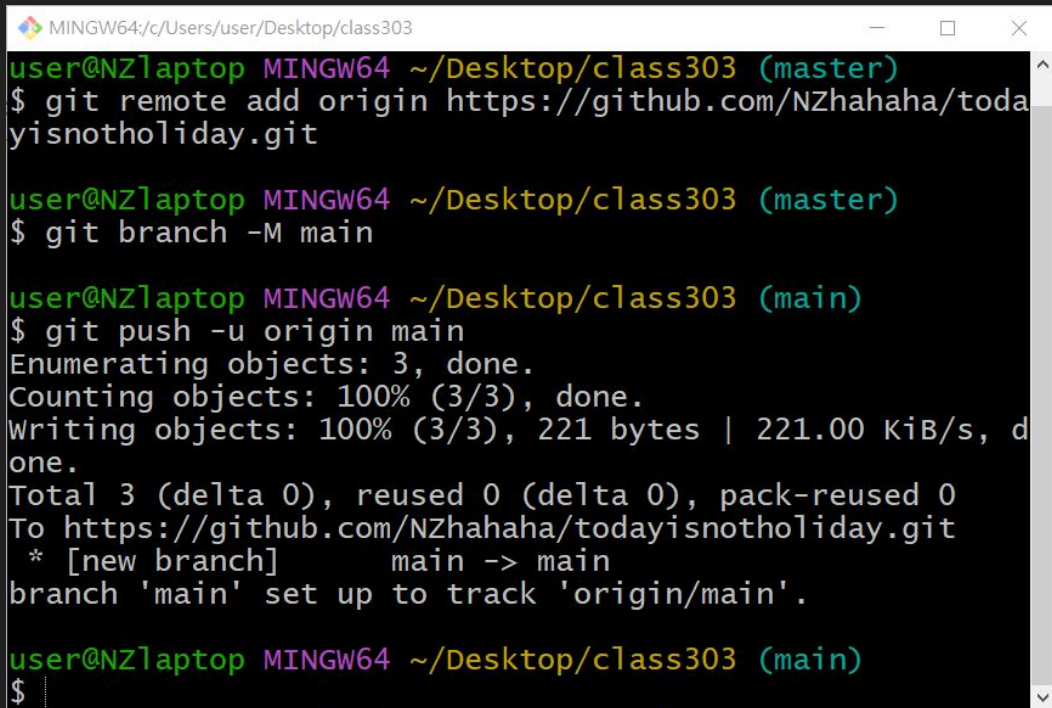
因為我們已經有本機端資料庫  
所以選擇只有3行指令的

**...or push an existing repository from the command line**



## step 4: 開始上傳

將剛剛3行程式碼丟到git bash



```
MINGW64: c:/Users/user/Desktop/class303
user@NZlaptop MINGW64 ~/Desktop/class303 (master)
$ git remote add origin https://github.com/NZhahaha/todayisnotholiday.git

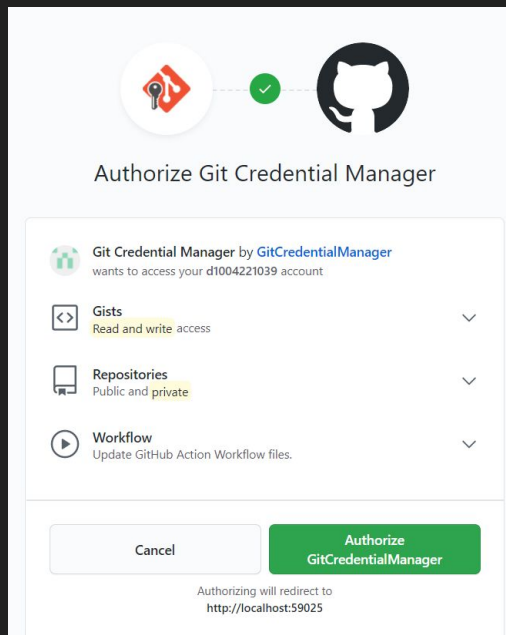
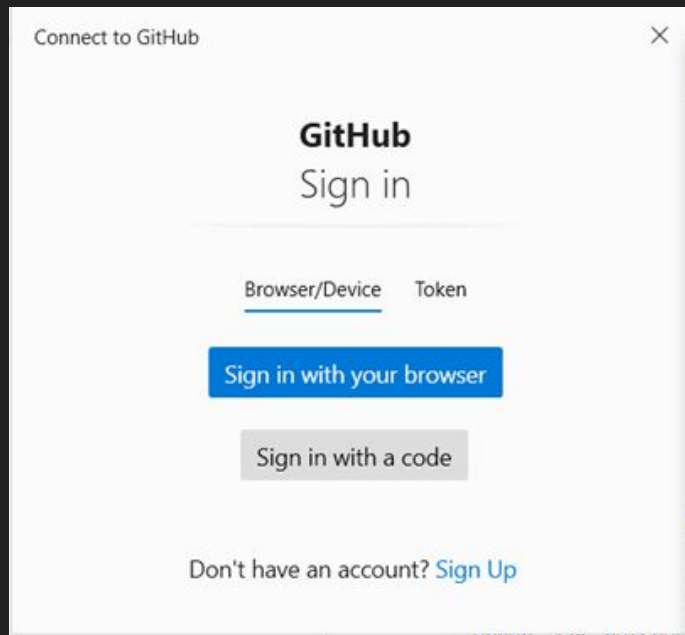
user@NZlaptop MINGW64 ~/Desktop/class303 (master)
$ git branch -M main

user@NZlaptop MINGW64 ~/Desktop/class303 (main)
$ git push -u origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 221 bytes | 221.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/NZhahaha/todayisnotholiday.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

user@NZlaptop MINGW64 ~/Desktop/class303 (main)
$
```

## step 5: 驗證身分(我驗證過所以找網路上資料)

沒出意外可以這樣用  
藍色(用瀏覽器登入)>>綠色(授權)



## step 5-1: 驗證身分(意外發生 by27)

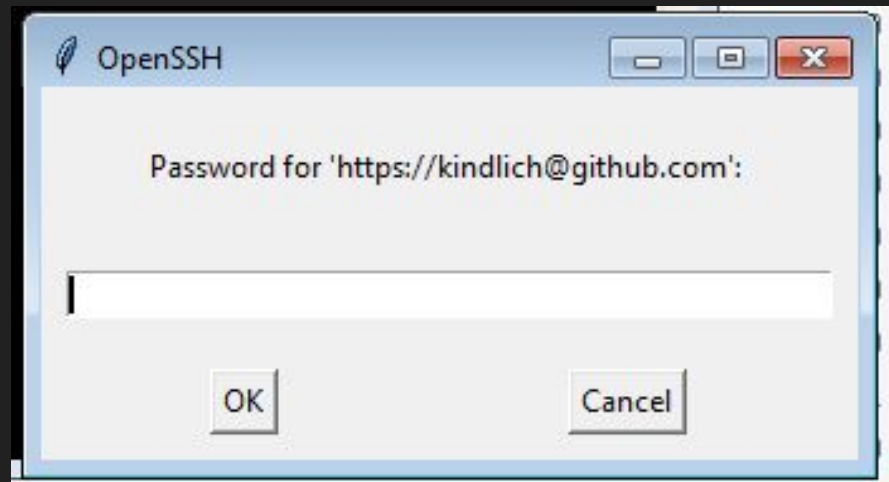
假如他是叫你打帳號密碼  
然後又跑出右邊這個東西  
恭喜你要去尋找大祕寶了(灑花  
這裡的password不是你想的那個  
但我懶得打那麼多有遇到再call我

或者關鍵字給你

Personal access tokens

給你參考

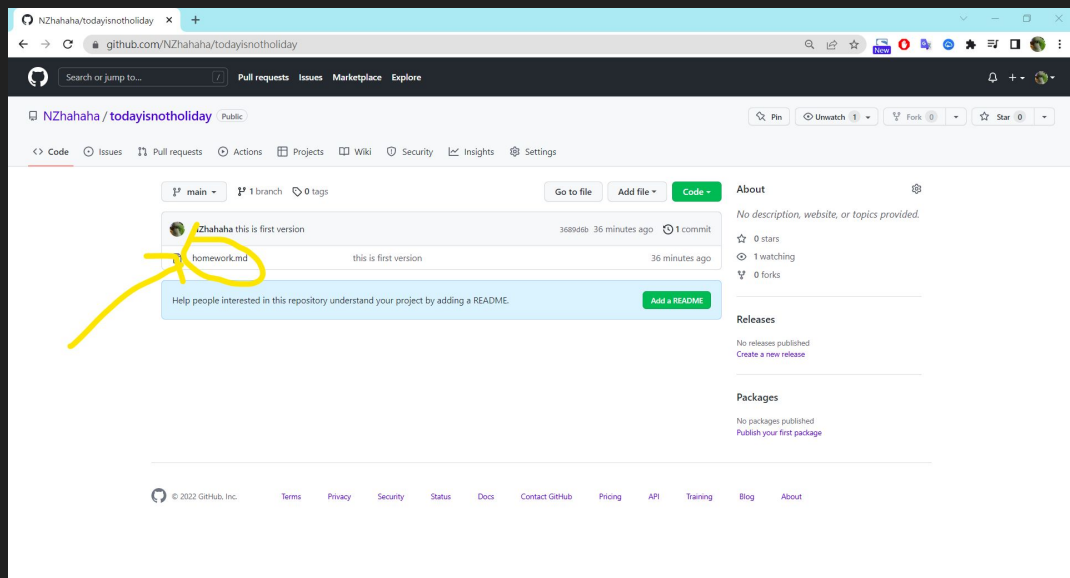
<https://andy6804tw.github.io/2021/08/14/token-authentication-requirements-for-git-operations/>





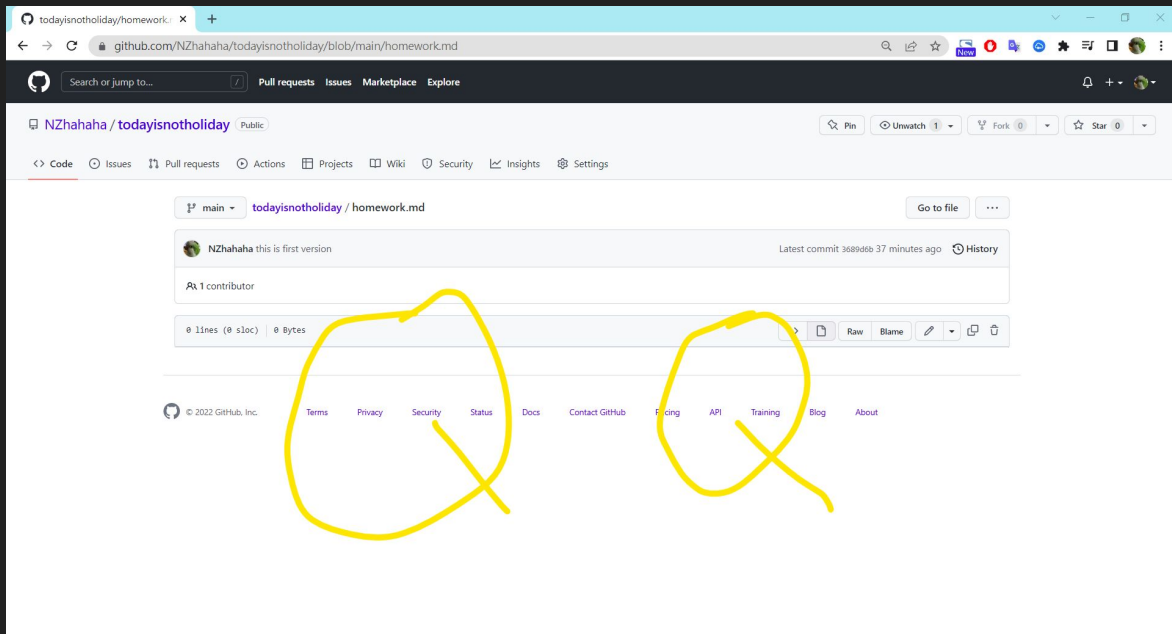
# step 6: 大功告成

重新整理網頁會出現以下畫面  
然後點黃色框框就會出現你打的文件檔



## step 6-QWQ: 然後我就翻車了

你各位要記得按Visual Studio Code的存檔阿~~  
不然QQ那部份就跟我一樣會是空白的了(但我要睡了就不重做XD





本著作由蘇恩立製作，以[創用CC 姓名標示-非商業性-相同方式分享 3.0 台灣 授權條款](https://creativecommons.org/licenses/by-nc-sa/3.0/tw/)釋出。

# Congratulations ~~~

Simple version  
based on 洪志菁的講義

written by NZ

