



# 版本控制與GITHUB實作

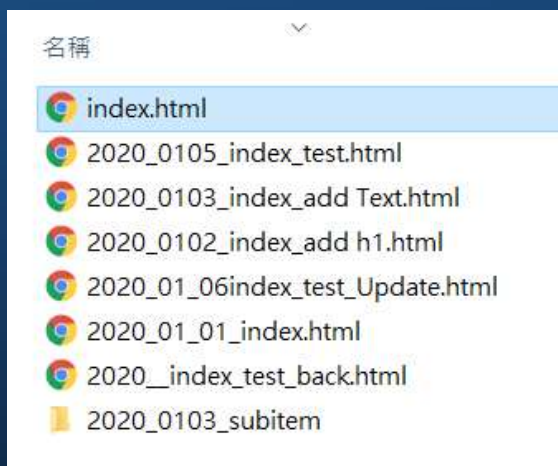
# AGENDA

- 版本控制
- GitHub 起手勢
- GitHub 動手做
- 利用GitHub做網頁

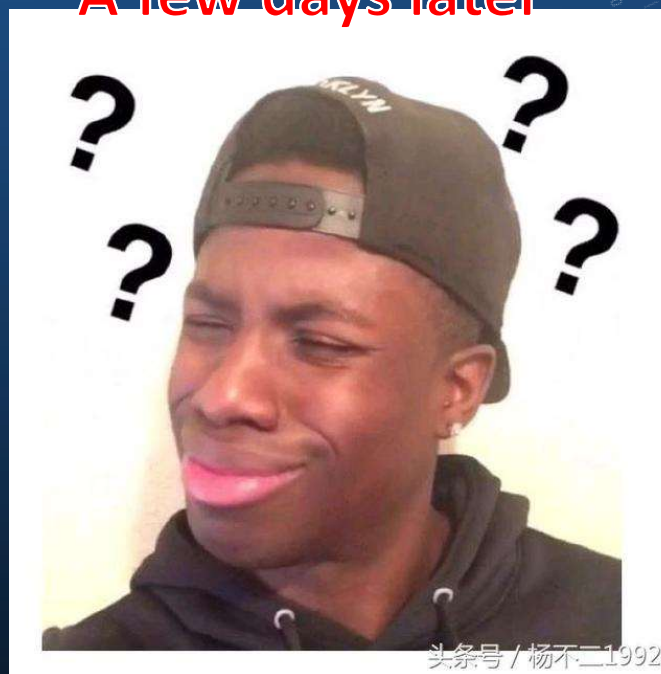
# VERSION CONTROL(版本控制)

# 文件管理-BEFORE

- 每個檔名依照需求，存不同的檔名

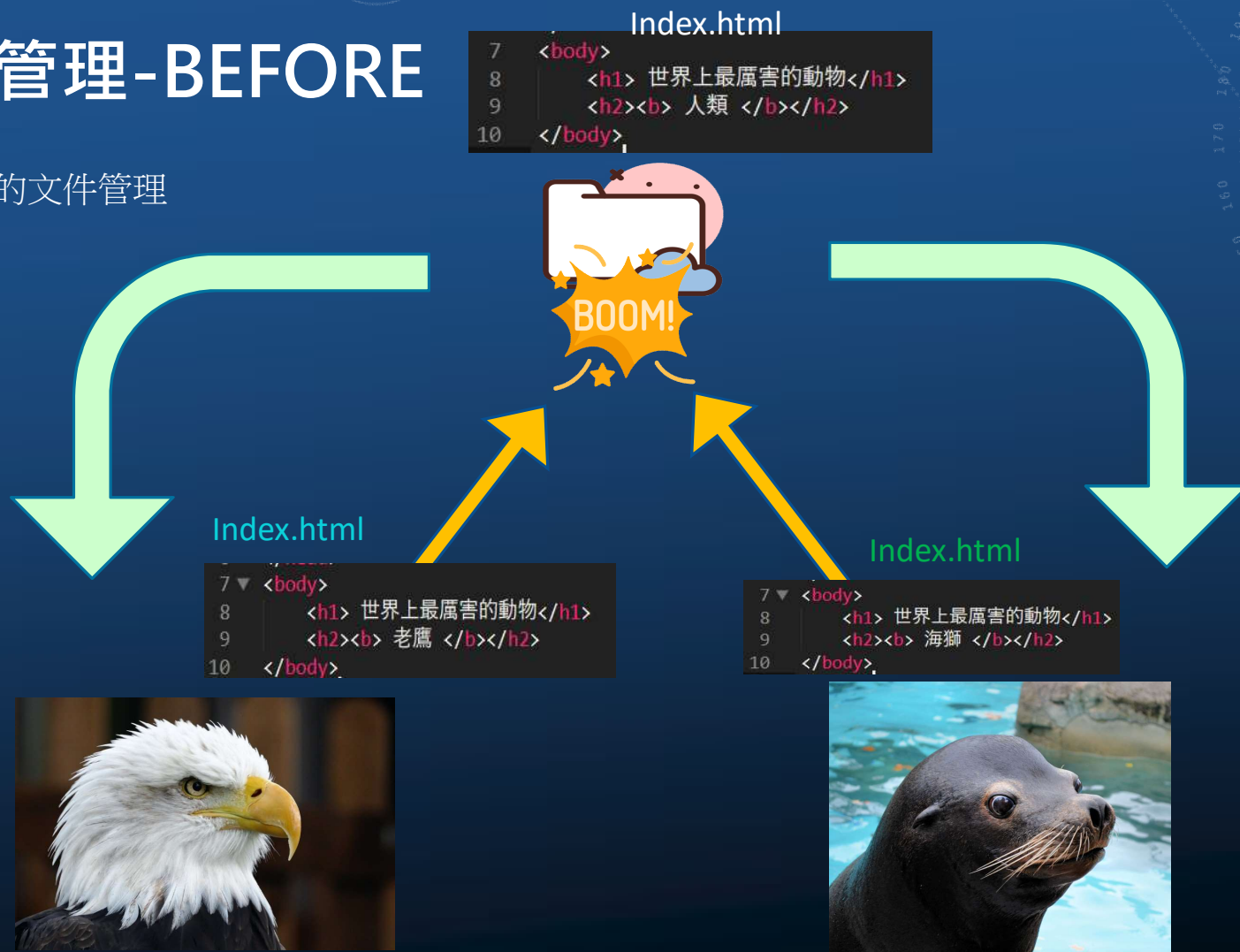


**A few days later**



# 文件管理-BEFORE

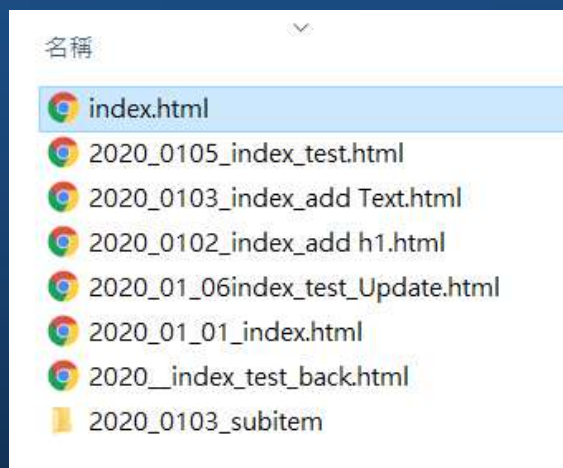
- 多人的文件管理



# 版本控制(VERSION CONTROL)

- 用於多人或多版本的開發
- 完整記錄整個程式的變更歷程(變更人員、內容、時間...等)
- 版本紀錄的優點
  - 程式修改歷程可追溯
  - 可追蹤整個檔案的歷史紀錄
  - 必較版本差異
  - 可多人同時協作
  - 當修改內容衝突可告知警示
  - 可依版本特性建立不同分支 或將分支進行合併

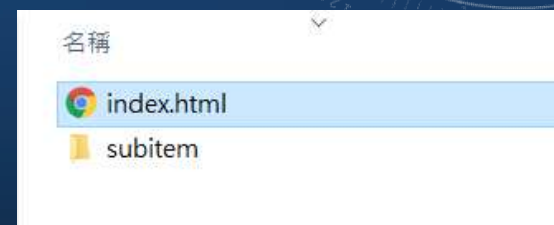
# 版本控制(VERSION CONTROL)



V1  
2020/1/1  
New index.html  
建立檔案  
修改人: 喵星人

V2  
2020/1/2  
修改 index.html  
新增標題資訊  
修改人: 喵星人

V3  
2020/1/3  
修改 index.html  
新增內文資訊  
新增subitem 資料夾



# GITHUB 起手勢

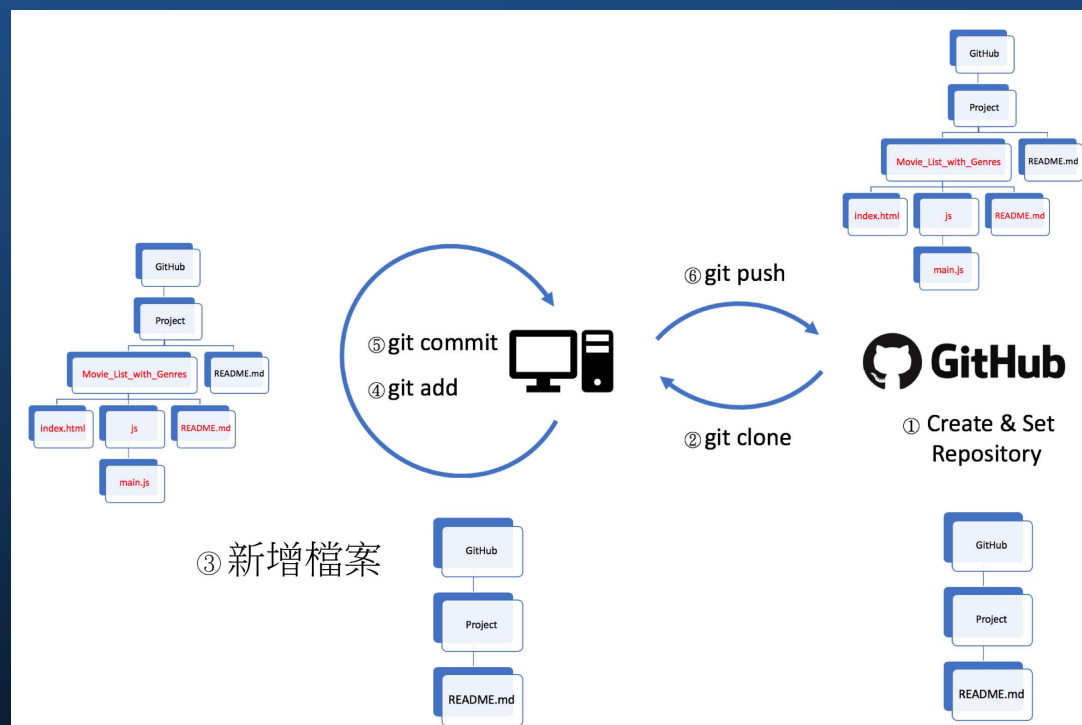


# 使用GIT做版本控制

- 是一種分散式的版本控制系統
- 可將檔案的修改紀錄儲存起來
- 可將檔案進行還原
- 當檔案修改有衝突時，提出警告避免檔案互相覆蓋

# GITHUB 基本操作流程

- GitHub的操作流程



參考來源: [【 DevOps 】透過 Visual Studio Code 進行 Clone 與 Push 操作 - Windows](#)

# GITHUB操作說明

- Git init : 建立本地的 git 資料庫
- Git clone : 將遠端的資料複製一份到本地端
- **Git add(+ /加入)** : 將新增或修改的檔案加入更新列表中
- **Git commit (提交/認可)** : 確定提交的文件
- **Git Push(推送)** : 將更新丟到遠端的GitHub上

# 安裝GIT

- Git 下載連結 : <https://git-scm.com/>



```
命令提示字元
Microsoft Windows [版本 10.0.18363.720]
(c) 2019 Microsoft Corporation. 著作權所有，並保留一切權利。
C:\Users\jo>git --version
git version 2.21.0.windows.1
C:\Users\jo>
```

按下windows + s  
輸入指令“cmd”開啟命令提示字元  
輸入“git -version”

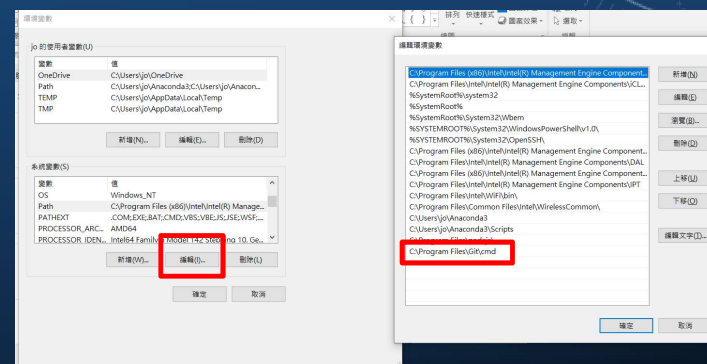
Mac 安裝請參考

<https://w3c.hexschool.com/git/fd6f6be>

# GIT 環境變數設定

- 安裝完Git後，需要對windows 環境設定環境變數，將Git的路徑加到環境變數中

C:\Program Files\Git\cmd



更多VS Code + Git 的環境設定，請參考

<https://medium.com/itsems-frontent/vs-code-github-%E8%A8%AD%E5%AE%9A-a7660e7e75ec>

# 安裝GIT

- 安裝完畢後，使用指令，確認是否設定成功
- 新增Git 使用資訊，在終端機下輸入以下指令

```
git --version
```

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

- 查看Git的設定，是否有把帳號與email設定正確

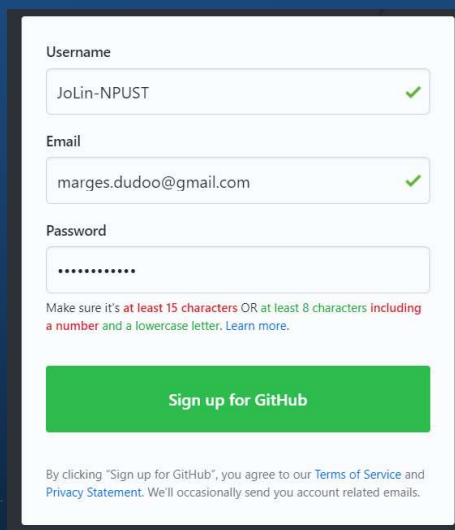
```
git config --list
```

```
C:\Users\jjo>git config --list
core.symlinks=false
core.autocrlf=true
core.fscache=true
color.diff=auto
color.status=auto
color.branch=auto
color.interactive=true
help.format=html
rebase.autosquash=true
http.sslbackend=openssl
http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bu
credential.helper=manager
core.editor='C:\Users\jjo\AppData\Local\Programs\Microsoft V
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
user.name=jolin-Dudoo
user.email=marges.dudu@gmail.com

C:\Users\jjo>
```

# 註冊 GITHUB

- Github 的官方網頁 : <https://github.com/>
- 請利用學校的mail 新增一個帳號
  - Username 請使用你的學號



Username

JoLin-NPUST ✓

Email

marges.dudoo@gmail.com ✓

Password

.....

Make sure it's **at least 15 characters** OR **at least 8 characters including a number and a lowercase letter**. [Learn more.](#)

**Sign up for GitHub**

By clicking "Sign up for GitHub", you agree to our [Terms of Service](#) and [Privacy Statement](#). We'll occasionally send you account related emails.



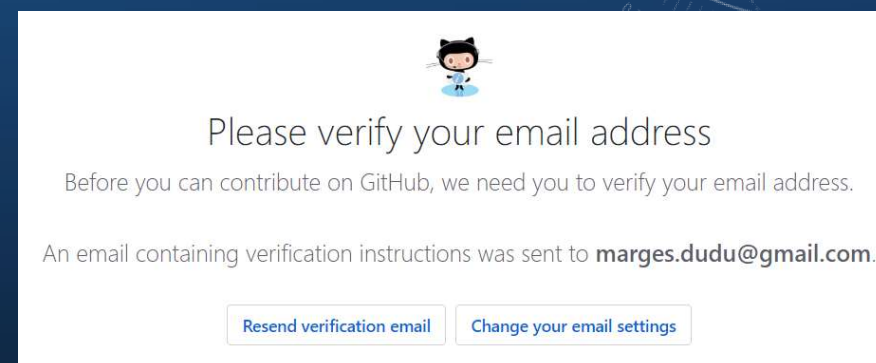
## Welcome to GitHub


Woohoo! You've joined millions of developers who are doing their best work on GitHub. Tell us what you're interested in. We'll help you get there.

What kind of work do you do, mainly?

[Software Engineer](#)  
I write code

[Student](#)  
I go to school





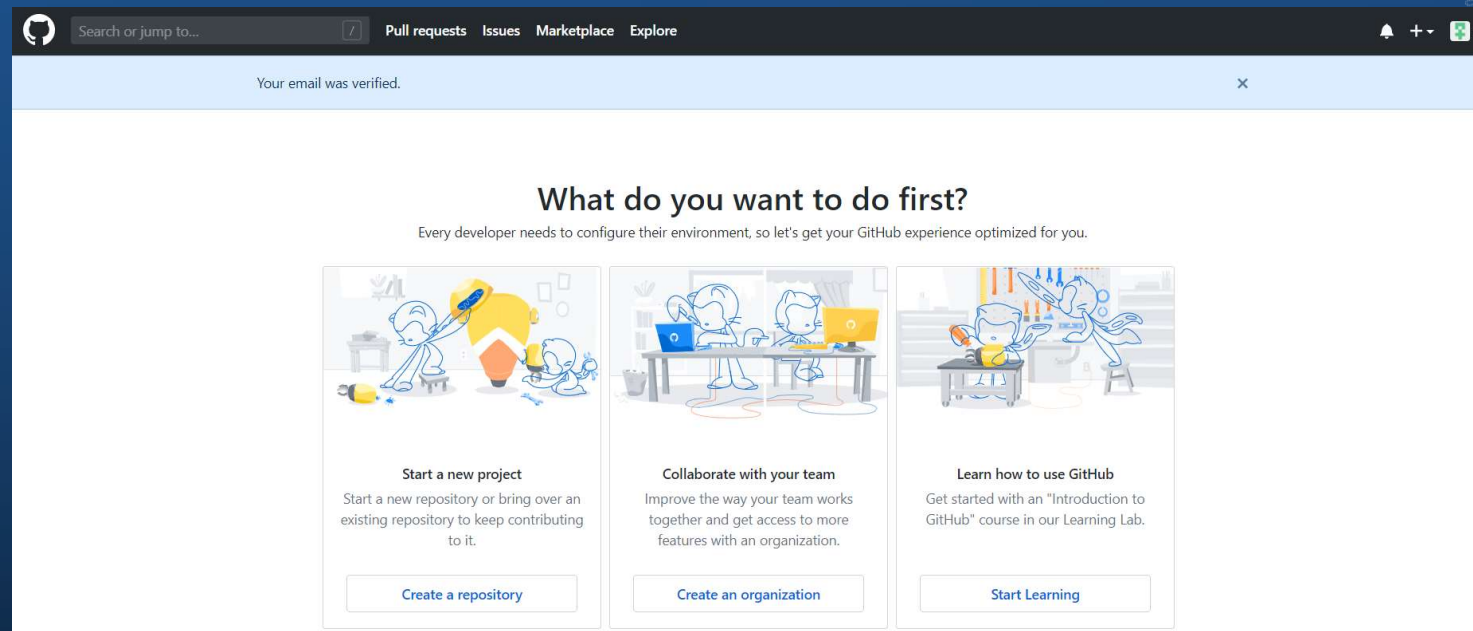
## Please verify your email address

Before you can contribute on GitHub, we need you to verify your email address.

An email containing verification instructions was sent to **marges.dudu@gmail.com**.

[Resend verification email](#) [Change your email settings](#)

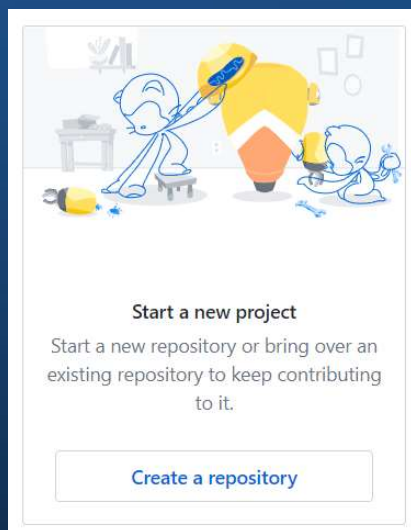
# 註冊 GITHUB





# 新增遠端資料庫

# 建立專案



設定專案的名稱

請設定成 public

Owner: jolin-Dudoo / Repository name: HelloGit

Great repository names are short and memorable. Need inspiration? How about **fictional-chainsaw**?

Description (optional): Git Hub 測試用網頁

☒ Public: Anyone can see this repository. You choose who can commit.

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

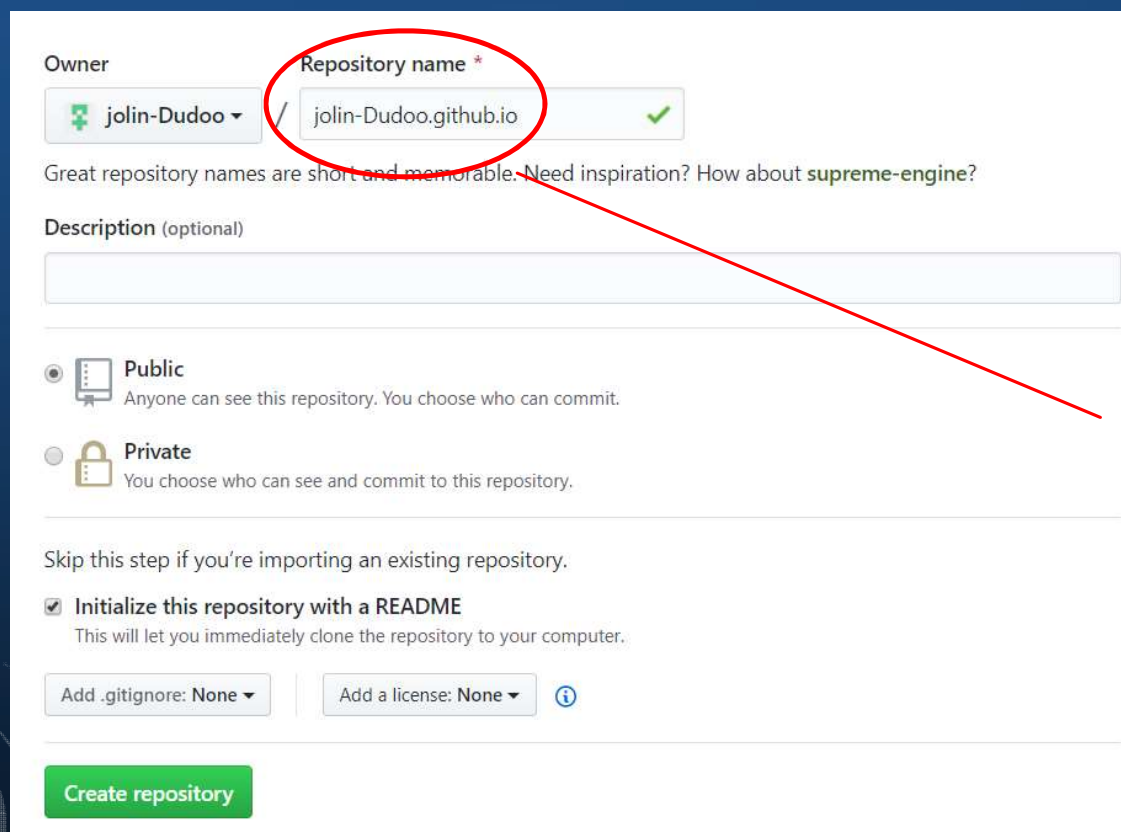
Skip this step if you're importing an existing repository.

☒ Initialize this repository with a README: This will let you immediately clone the repository to your computer.

Add .gitignore: None / Add a license: None

Create repository

# 用GITHUB架站



Owner: jolin-Dudoo

Repository name \*: jolin-Dudoo.github.io ✓

Great repository names are short and memorable. Need inspiration? How about **supreme-engine**?

Description (optional)

☐ Public  
Anyone can see this repository. You choose who can commit.

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

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

☒ Initialize this repository with a README  
This will let you immediately clone the repository to your computer.

Add .gitignore: None Add a license: None ⓘ

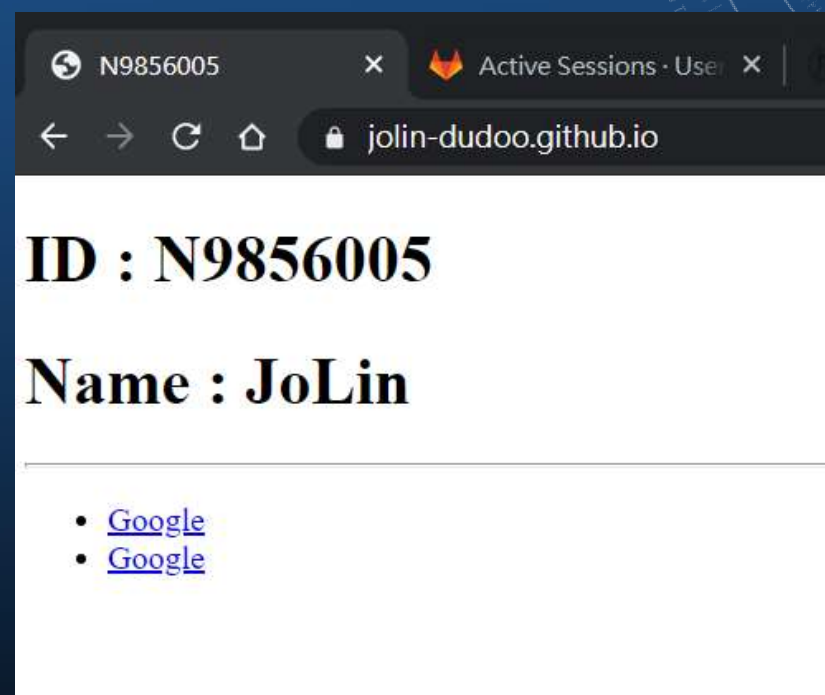
Create repository

新增Repository時採用  
**UserName.github.io**  
例如:N9856005.github.io

# 用GITHUB架站

- 上傳完畢後，只要在網址列輸入網址就可以連到網頁



Username.github.io



# 下載專案到LOCAL

- 專案建立完成，下方會有個Quick setup網址
- 當中所顯示的path就是目前專案的Path
- path右方的按鈕按下去，就可以複製網址

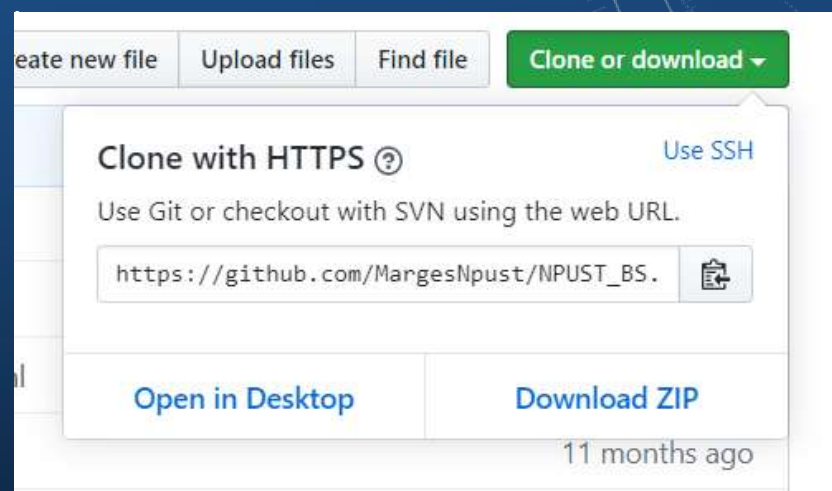
## Quick setup — if you've done this kind of thing before

 Set up in Desktop or **HTTPS** **SSH** `https://github.com/MargesNpust/MargesNpust.github.io.git` 

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

# 下載專案到LOCAL

- 若你的專案之前已經上傳過資料，在專案頁面的右上角會有一個 Clone or download的按鈕，直接點選也能看到網址



# VS CODE+GIT HUB

- 新增一個資料夾，要用來放置Git Hub 的資料夾，例如: c:\GitHub\_Test
- 由VS Code開啟資料夾
- 在VS Code 上方 終端機 > 新增終端機，開啟一個終端機頁面
- 輸入指令

**git clone[剛剛複製的github網址]**

例如: `git clone https://github.com/jolin-Dudoo/HelloGit.git`

```
PS C:\20200501_Npust\GitHub_Test> git clone https://github.com/jolin-Dudoo/HelloGit.git
Cloning into 'HelloGit'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
PS C:\20200501_Npust\GitHub_Test> 
```

# VS CODE+GIT HUB



目前要做的網址是 有個子目錄(HelloWorld) ,  
下方有個index.html的檔案

Jolin-Dudoo.github.io

Jolin-Dudoo.github.io/HelloWorld/index.html



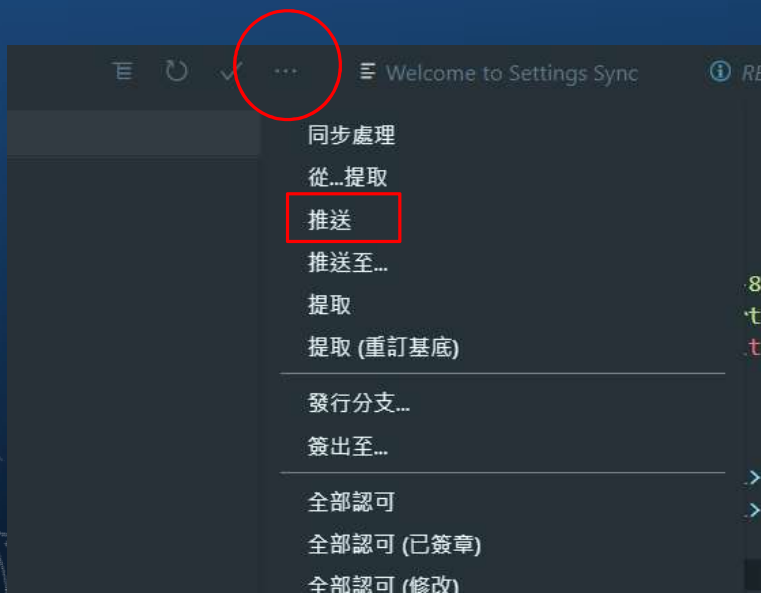
# 新增檔案到遠端

- 在剛建立連結的資料夾下，分別新增一個資料夾以及一個檔案
- 左方“原始控制檔”的tab，會發現有個變更
- 按下“+” > 檔案的狀態變成暫存的變更



# 新增檔案到遠端

- 在上方的訊息欄位，輸入此次變更的內容，例如:新建立檔案
- 按下上方的勾勾，則該檔案會變成認可
- 點選上方的 ...，點選”推送”



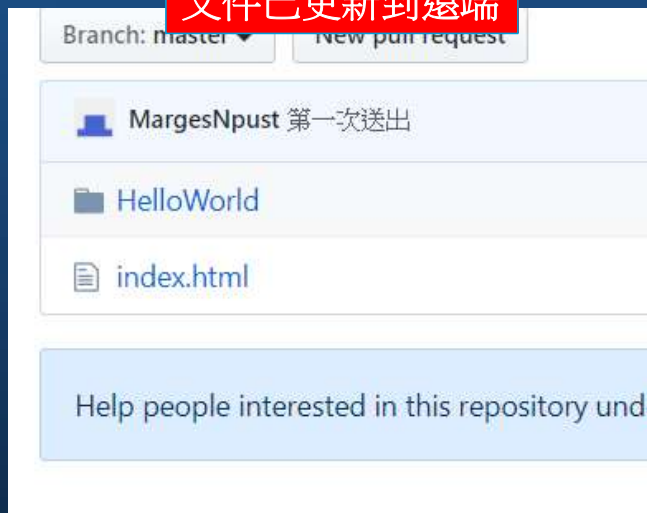
# 新增檔案到遠端

- 點選上方的勾勾輸入 註解文字



# 連線GITHUB網頁

文件已更新到遠端

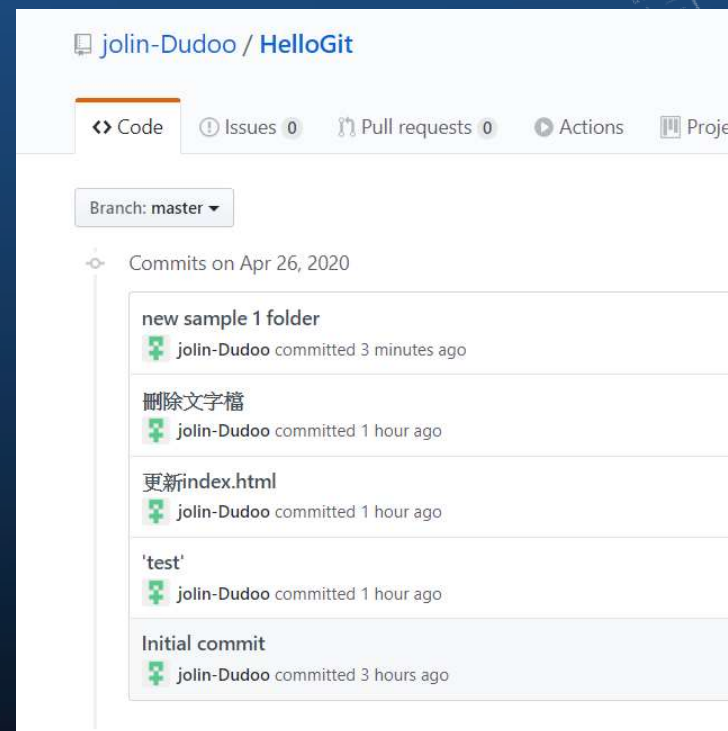
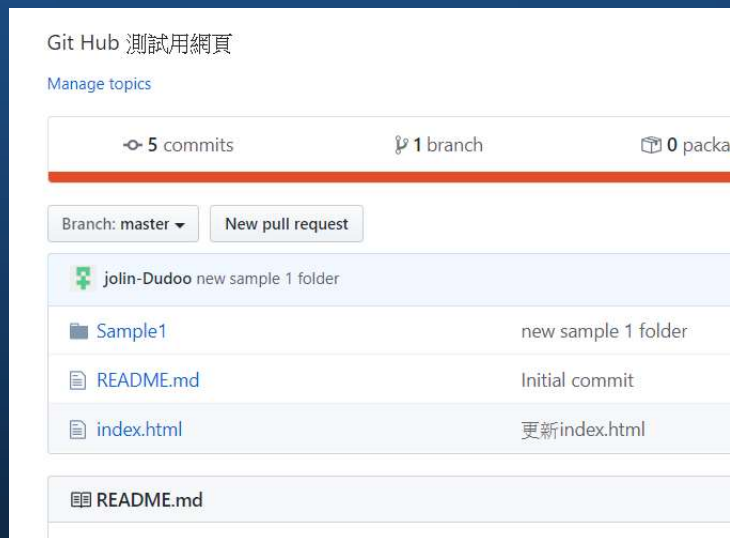


利用網址 就可以連到該網頁了



# 確認COMMITTS

- 在Git Hub 上，切換到code \ commits 就可以看到剛剛推送的資訊



## 補充說明

- 如果你在推送時，有出現permission錯誤，請用管理員身分，開啟終端機，下達指令

```
git config --system --unset credential.helper
```

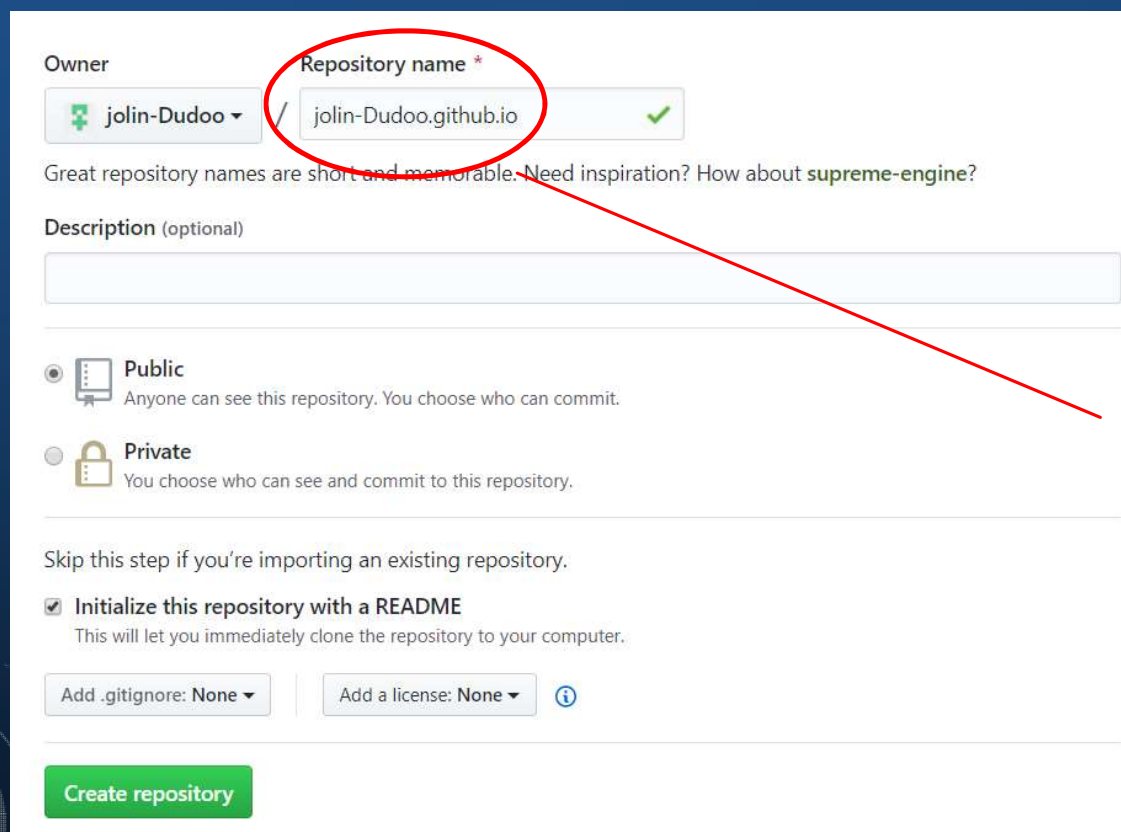
- 若你想要將帳號密碼記住，請在終端機輸入

```
git config --global credential.helper store
```

<https://blog.csdn.net/sz85850597/article/details/84939368>

# 利用GITHUB做網頁

# 用GITHUB架站



Owner: jolin-Dudoo

Repository name \*  
jolin-Dudoo.github.io ✓

Great repository names are short and memorable. Need inspiration? How about **supreme-engine**?

Description (optional)

☒ **Public**  
Anyone can see this repository. You choose who can commit.

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

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

☒ **Initialize this repository with a README**  
This will let you immediately clone the repository to your computer.

Add .gitignore: None ▾ | Add a license: None ▾ ⓘ

**Create repository**

新增Repository時採用  
**Username.github.io**



# 用GITHUB架站

- 上傳完畢後，只要在網址列輸入網址就可以連到網頁

Useename.github.io



# 參考網頁



31 JULY 2019 / DEVOPS

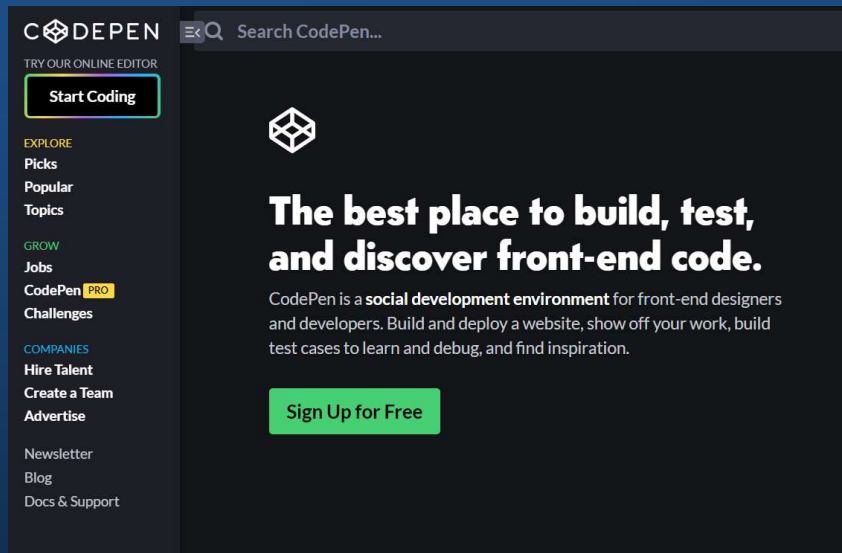
## 【 DevOps 】透過 Visual Studio Code 進行 Clone 與 Push 操作 - Windows

## 使用 GitHub 免費製作個人網站

GitHub 除了提供免費的 Git Server 外，如果推上去的分支的名字剛好叫做 `gh-pages` 虛擬主機要便宜得多，也安全得多。不過也有一些限制：

# 增強功力的網站

Code Pen



# 好用工具網-HACKMD

 HackMD

最新消息 | 收費方案 | 最近瀏覽 | 登入

## 用 Markdown 即時協作知識庫

用即時協作的 Markdown 編輯器書寫 團隊 文件，讓你快速協作、輕鬆紀錄想法、隨時共享知識。

[立即註冊](#)

### HackMD 使用教學

[回到使用教學頁面](#)

#### 快速上手

- [快速入門教學](#)
- [完整功能介紹](#)

#### 內容排版

- [基本排版](#)
- [在筆記中貼入程式碼](#)
- [在筆記中嵌入另一篇筆記](#)
- [如何撰寫數學式及圖表](#)
- [Emoji 完整表](#)
- [自訂文字顏色](#)
- [如何建立表格](#)

 變更於 4 個月前

 擁有這篇筆記

[訂閱](#)

## 用 HackMD 做簡報

[< 回到使用教學](#)

HackMD 整合了 reveal.js，讓我們可以用 Markdown 輕鬆寫簡報，讓我們開始吧！

### 製作簡報與分頁

只要 HackMD 有支援的 Markdown 語法就能放進簡報裡面，無論是流程圖、內嵌 Youtube 影片或是 Latex 語法都行！

我們用「三個連字號」(---) 組成的「分隔線」來區別簡報的分頁，記得在前後各加上一個空行：

```
# 簡報標題一
## 簡報標題二
---
- 清單
- 清單
```

# GITHUB作業

- 請從往GitHub上下載作業檔  
[https://github.com/MargesNpust/GitHub\\_HomeWork.git](https://github.com/MargesNpust/GitHub_HomeWork.git)
- 請撰寫JavaScript ,讓網頁呈現如下的效果
- 請將作業GitHub連結，傳到學習平台上，進行作業繳交  
範例網址: [https://github.com/UserName/GitHub\\_HomeWork.git](https://github.com/UserName/GitHub_HomeWork.git)

MIS快餐點餐價格表

	5號餐 8元	4號餐 11元	3號餐 14元	2號餐 17元	1號餐 20元
3	3*8=24	3*11=33	3*14=42	3*17=51	3*20=60
4	4*8=32	4*11=44	4*14=56	4*17=68	4*20=80
5	5*8=40	5*11=55	5*14=70	5*17=85	5*20=100
6	6*8=48	6*11=66	6*14=84	6*17=102	6*20=120
7	7*8=56	7*11=77	7*14=98	7*17=119	7*20=140