

# Computer Programming

Section week 10

# Video time

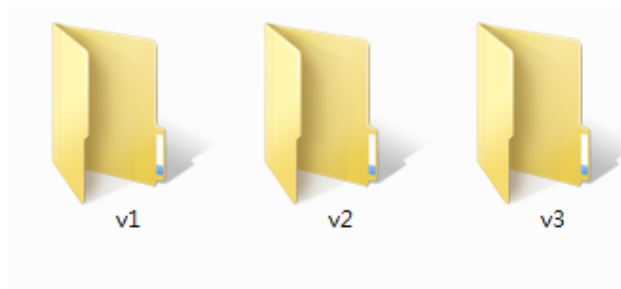
伽利略 - 自由落體實現 BBC

# Agenda

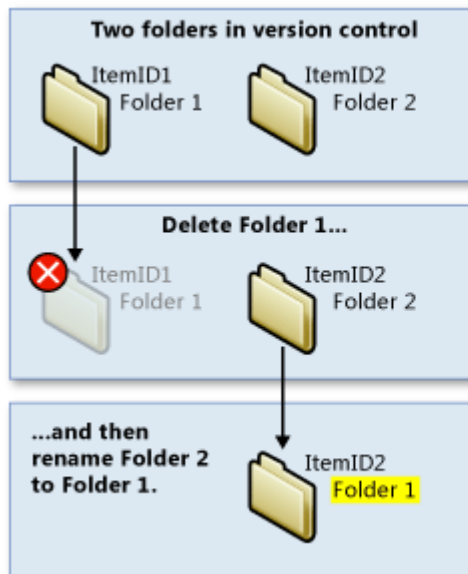
- 版本管理概念
- 介紹 GIT(基礎)
- GITHUB 是什麼
- 在 GITHUB 上建立自己的 repository
- GIT 指令介紹

# 版本管理概念

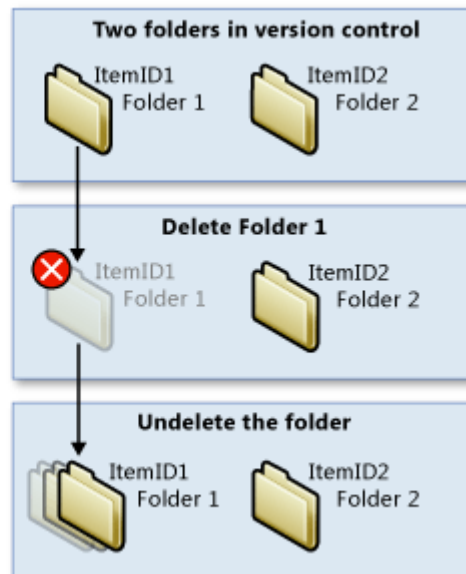
## 目錄管理法



Scenario A



Scenario B



# 介紹 GIT



**Git是一個分布式版本控制／軟體配置管理軟體，原是Linux核心開發者林納斯·托瓦茲(Linus Torvalds)為更好地管理Linux核心開發而設計。(refer [wiki](#))**

# GIT Original author



# 為什麼需要 GIT

- 檔案追蹤
- 區格差異(這次版本跟上次差在哪)
- 復原檔案(被覆蓋或是被弄壞了)
- 管控版本
- 多人協同

# Repository

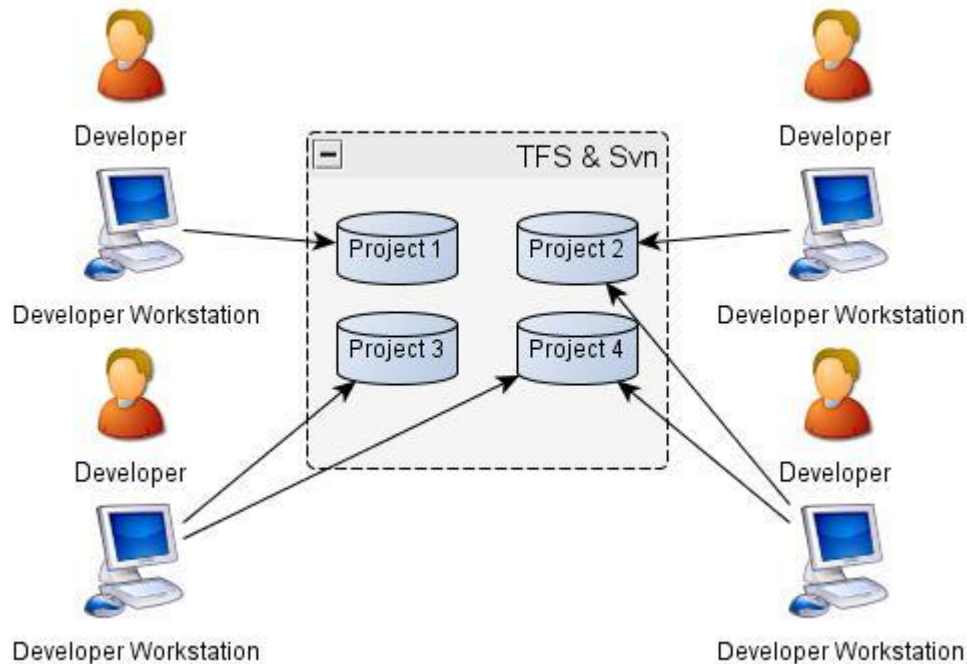
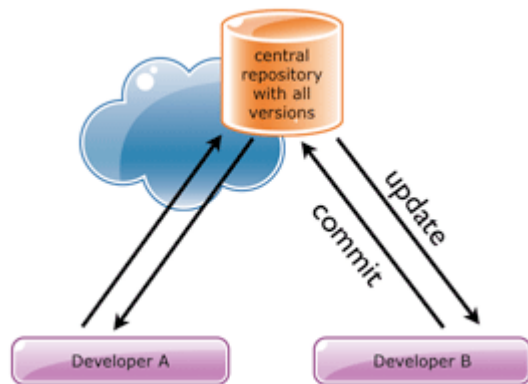




# GIT 管理倉庫



# 多人一起開發



# GITHUB 是什麼



**GitHub 是一個共享虛擬主機服務，用於存放使用Git版本控制的軟體代碼和內容計畫。它由GitHub公司（曾稱Logical Awesome）的開發者Chris Wanstrath、PJ Hyett和Tom Preston-Werner使用Ruby on Rails編寫而成。**

# GITHUB

**GitHub同時提供付費帳戶和為開源計畫提供的免費帳戶。根據在2009年的Git用戶調查，GitHub是最流行的Git存取站點。除了允許個人和組織建立和存取代碼庫以外，它也提供了一些方便社會化軟體開發的功能，包括允許用戶跟蹤其他用戶、組織、軟體庫的動態，對軟體代碼的改動和 bug 提出評論等。GitHub也提供了圖表功能，用於顯示開發者們怎樣在代碼庫上工作以及軟體的開發活躍程度。**

# GITHUB 上建立 Repo

[Explore](#) [Gist](#) [Blog](#) [Help](#)

mdetw



mdetw

Joined on 10 Nov 2014

0

Followers

0

Starred

0

Following

**Pro tip:** updating your profile with your name, location, and a profile picture helps other GitHub users get to know you.

[Edit profile](#)

Contributions

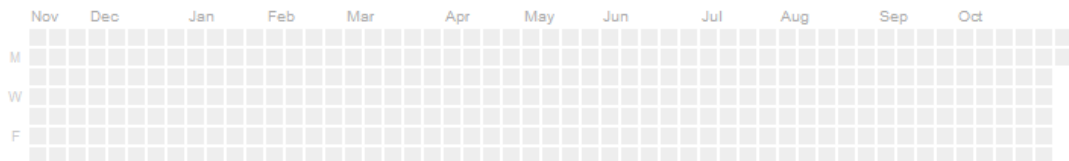


Repositories



Public activity

## Contributions

Summary of Pull Requests, issues opened, and commits. [Learn more.](#)

Less



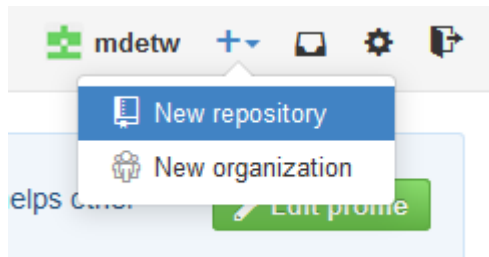
More

This is your **contribution graph**. When you make a commit to a repository, you'll get a for that day. Make more contributions and you'll get a darker green square. Over time, your chart might start looking [something like this](#).

We have a quick guide that will show you how to create your first repository. You'll also make a commit and **earn your first green square!**

[Read the Hello World guide](#)

# 新增 Repo





Owner



mdetw ▾

Repository name

/ |

Great repository names are short and memorable. Need inspiration? How about **tripping-octo-bear**.

Description (optional)



**Public**

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



**Private**

You choose who can see and commit to this repository.



**Initialize this repository with a README**

This will allow you to `git clone` the repository immediately. Skip this step if you have already run `git init` locally.

Add .gitignore: **None** ▾

Add a license: **None** ▾



Create repository



# GIT 個人設定

讓 GIT 知道你是誰, 也讓別人知道你是誰

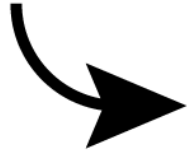
```
git config --global user.name "myname"
```

```
git config --global user.email "my@email.com"
```

# GIT 簡單指令 實做

- clone
- commit
- push

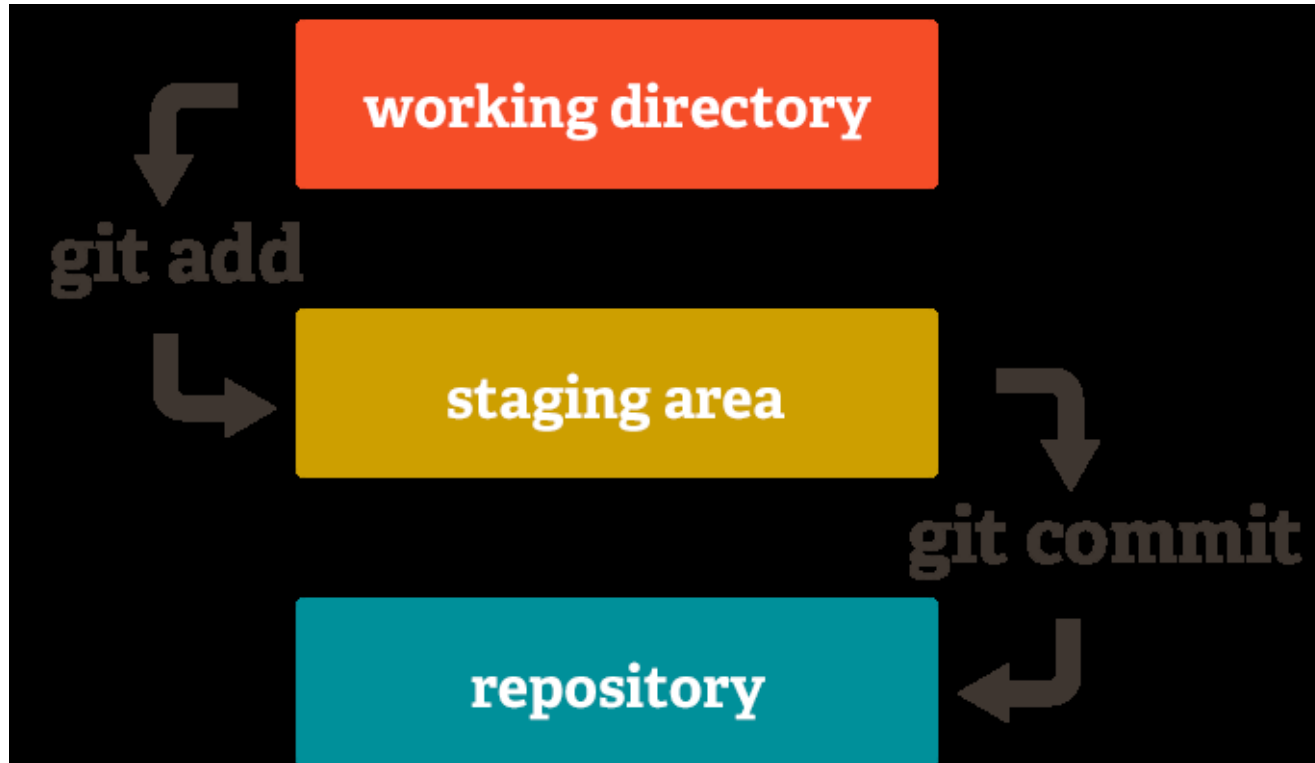
# GIT - clone



`git clone`



# **Git - commit**



# Git - push

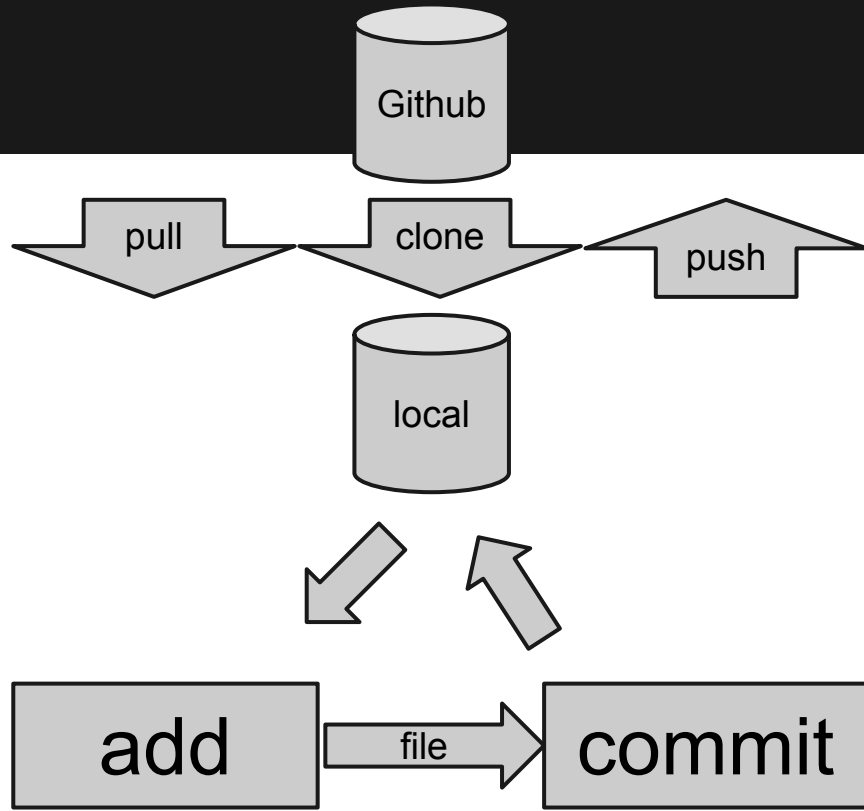
Push



# GIT - push



# 過程



# 結論

討論我們今天做了什麼？