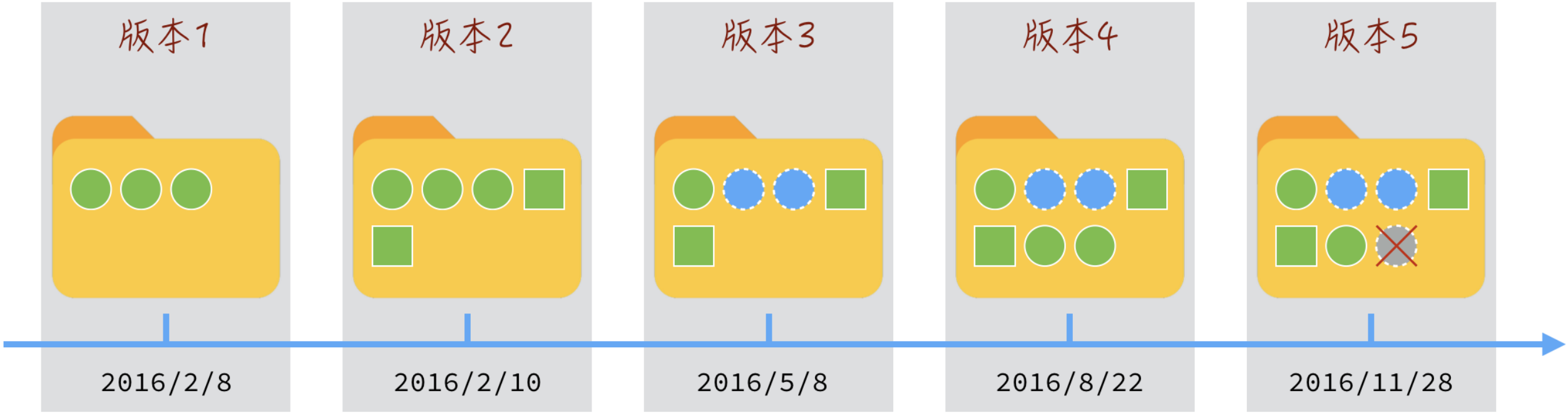
Git版本控制系统

田弘華

什麼是版本控制系統？

檔案內容管理的需要

- 我們常常新增、編輯、修改許多檔案。



檔案內容管理的需要

- **需要清楚的記錄每個檔案**
  - 是誰
  - 在什麼時候
  - 增加、修改或刪除檔案
  - 改變了什麼內容

安装Git

安裝與驗證

步驟一：到官方網站下載合適的版本

- 網址：https://git-scm.com/download/win

步驟二：一路按「下一步」按到底

步驟三：安裝完成後，找到「Git Bash」應用程式

- 在黑黑的視窗輸入下面指令，看看 Git 是不是有安裝起來

```
$ which git
$ git  --version
```

使用者設定

- **開始使用 Git，要做的第一件事（只要做一次就好），就是設定使用者的Email信箱和使用者名稱。**
- 請打開你的終端機視窗，輸入下面這兩行指令：

```
$ git config --global user.name "Eddie Kao"
$ git config --global user.email "eddie@5xruby.tw"
```

使用者設定驗證

- 輸入完成之後，檢視一下目前的設定：

```
$ git config --list
user.name=Eddie Kao
user.email=eddie@5xruby.tw
```

使用Git

本地準備：建立.git子目錄

全新的開始...第一次使用 Git，先建立一個新的目錄。

方法：用滑鼠建立子目錄 ＋ git init

～用指令建立子目錄

打開終端機視窗(CMD)，打入以下指令，# 符號是說明不用輸入：

```
$ cd C:\Users\bigfl\Desktop\tmp    #用cd指令切換至 /tmp 目錄
$ mkdir git-practice       #用 mkdir 指令建立 git-practice 子目錄
$ cd git-practice          #用cd指令切換至 git-practice 子目錄
$ git init                 # 初始化子目錄
# 這個指令會在這個子目錄裡建立一個 .git 目錄，讓 Git 對這個目錄進行版控
```

本地準備：建立.git子目錄

- 本來就有的目錄...

如果想針對本來就存在的目錄進行版控，其實只要到那個目錄下執行 **git init** 指令即可。

- 不想再被 **Git** 控制的目錄

Git 的版控很單純，全部就只是靠 **.git** 目錄在做事而已。所以不想被版控，或是只想給客戶不含版控紀錄的內容，移除 **.git** 目錄即可。

把檔案交給 Git 控管

- 在開始之前，可以先用 git status 指令查詢目錄現在的「狀態」。

建立檔案

- 在目錄裡透過系統指令建立一個名稱為 welcome.html 的檔案，裡面內容為 “hello, git”：

```
$ echo "hello, git" > welcome.html
```

～這個檔案尚未被加到 Git 版控系統裡，還沒開始正式被 Git「追蹤」，只是剛剛才加入目錄而已。

把檔案交給 Git 控管

- 讓 Git 開始「追蹤」

```
git add welcome.html   # git add 檔名
```

～這個檔案已經被安置到暫存區（Staging Area），稍後跟其它檔案一起被存到儲存庫裡。

～如果想要一口氣把全部的檔案加到暫存區，可直接使用 --all 參數：

```
git add --all
```

把檔案交給 Git 控管

- 把暫存區的內容提交到倉庫裡存檔

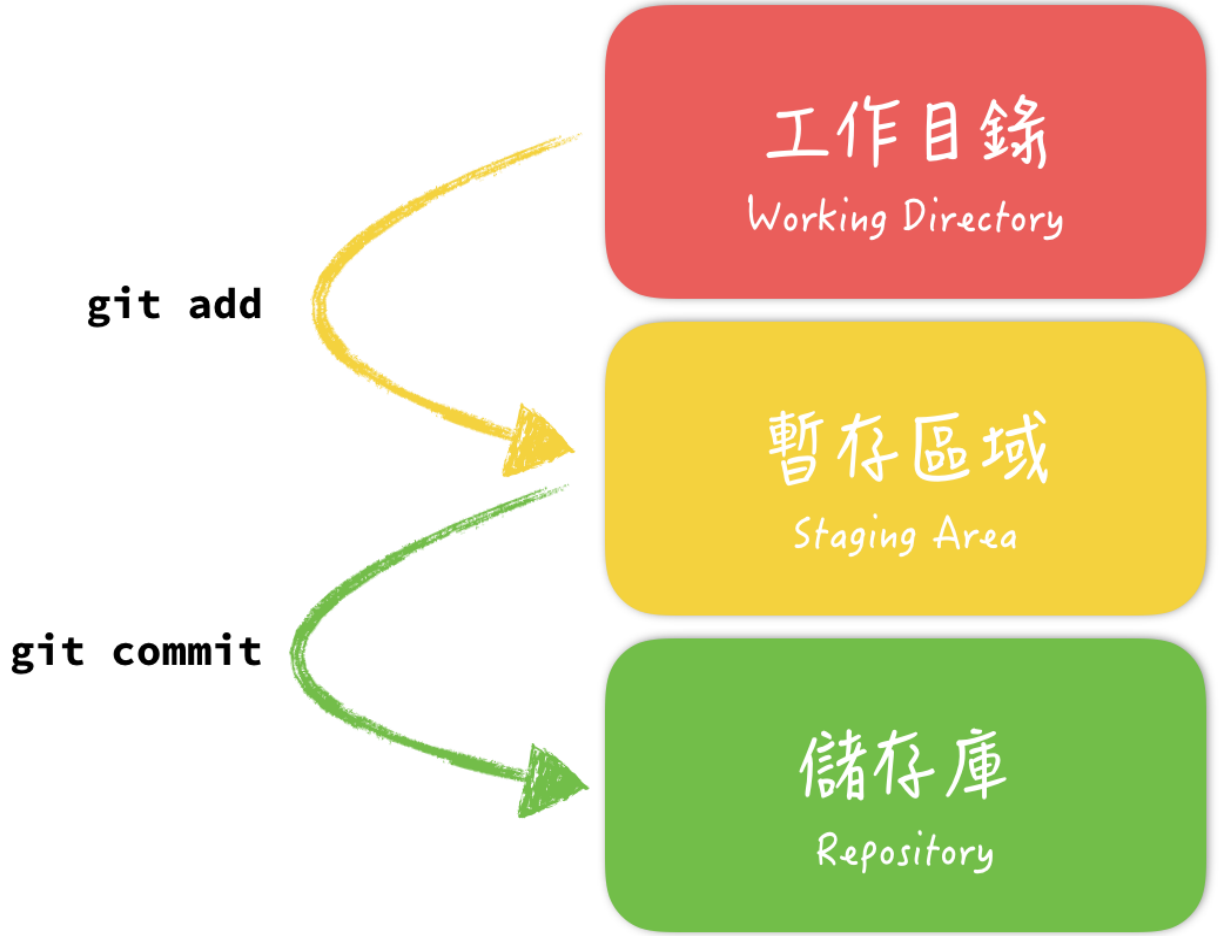
使用 git commit 指令，在後面加上 -m “init commit”。引號中加入簡單的中英文說明「你在這次的 Commit 做了什麼事」。

```
git commit -m "init commit“
```

～完成存檔（或備份）了！

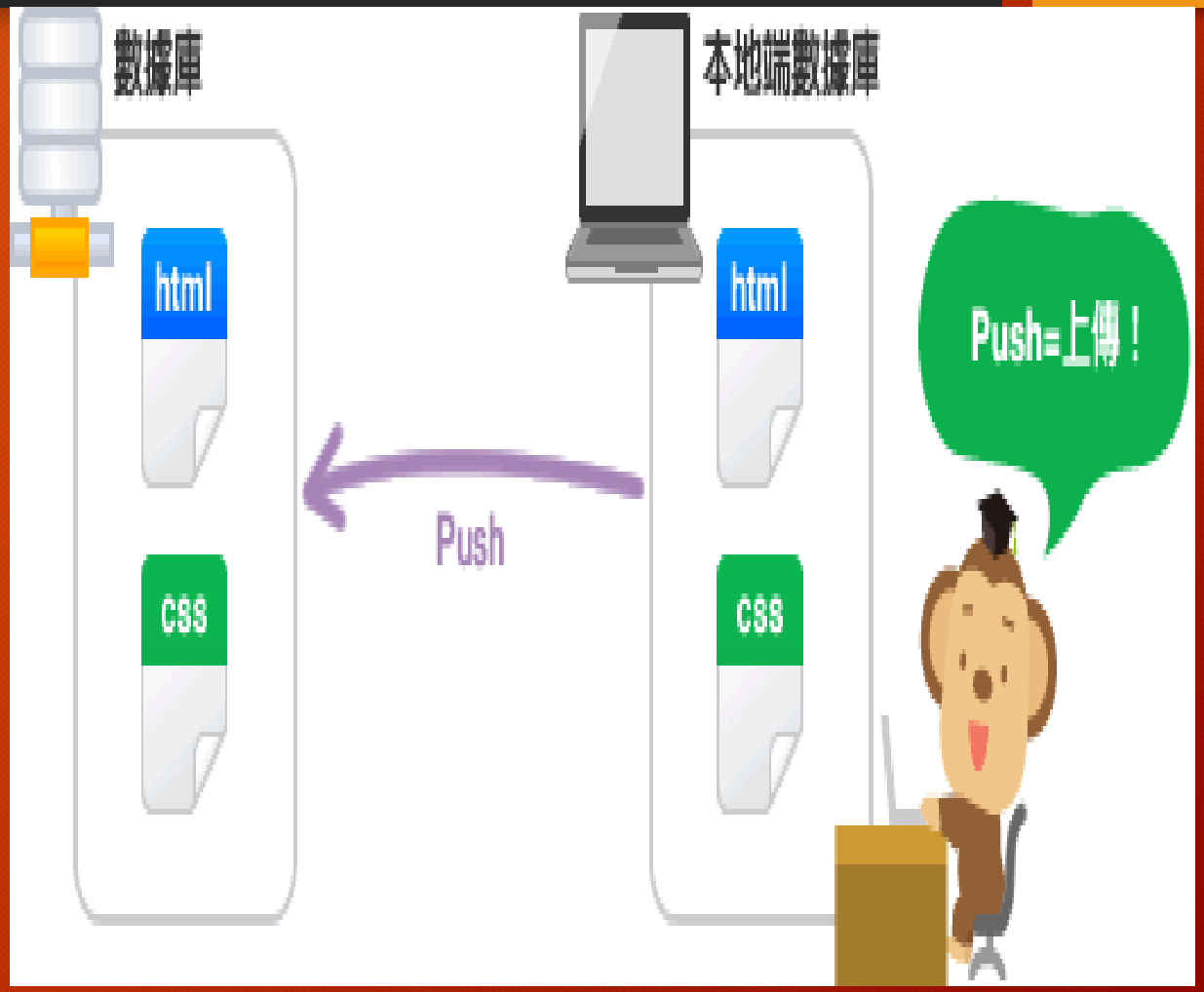
工作區、暫存區與儲存庫

- Git 分成「工作目錄（Working Directory）」、「暫存區（Staging Area）」以及「儲存庫（Repository）」三個區塊，透過不同的 Git 指令，可以把檔案移往不同的區域。
- git add 指令把檔案從工作目錄移至暫存區（或索引）。
- git commit 指令把暫存區的內容移至儲存庫。



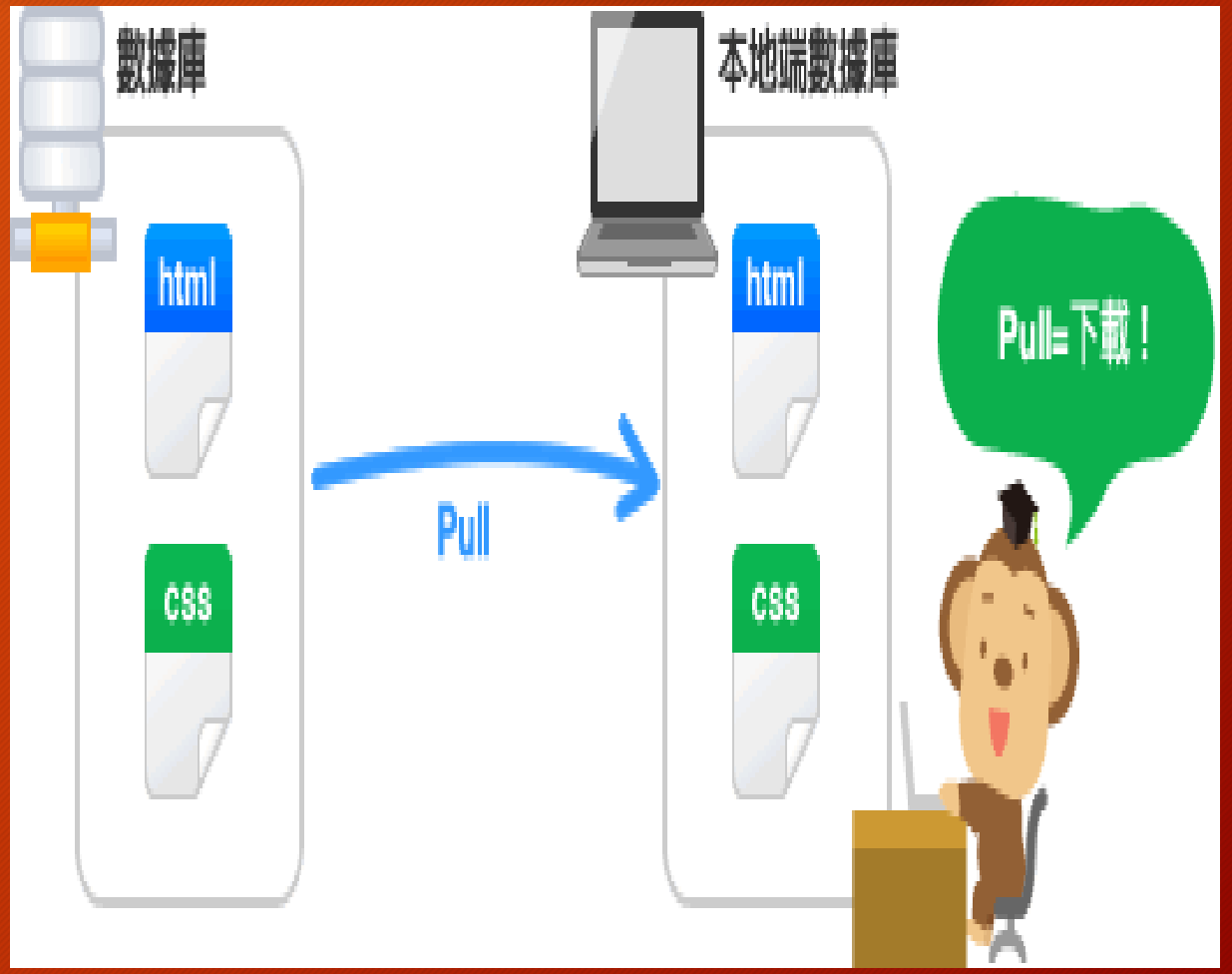
Push到遠端數據庫

- 在Git執行Push(推送)操作。執行Push之後，本地端的修改歷史會被上傳到遠端數據庫。
- 遠端數據庫的修改歷史就會和本地端數據庫的修改歷史保持同步。



從遠端數據庫執行Pull

- 欲同步遠端數據庫以更新本地端數據庫，請使用Pull(拉取)。
- 執行pull之後，會從遠端數據庫下載最新的修改歷史，將其同步到自己的本地端數據庫。



Git and Github

Owner

Repository name *

HungHuaTien / 課程名稱

Great repository names are s Your new repository will be created as - How about **jubilant-octo-spoon**?

Description (optional)

內容描述

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

指令摘要

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

Set up in Desktop or HTTPS SSH https://github.com/HungHuaTien/-.git

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

...or create a new repository on the command line

```
echo "# -" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/HungHuaTien/-.git
git push -u origin master
```