

了解 Git 的分支

「分支」是 Git 非常方便的功能之一,是能將歷程分開來 記錄的功能。將軟體分成不同的版本管理,或是將網站 分成不同的製作版本管理,就能輕鬆地管理相關的資料。 如果使用「Merge」這項功能,還能統整每個分支的修改 內容。請大家務必熟悉這項功能。





1

利用分支區分作業 - branch

在 Chapter 2 我們學到了利用 Git 記錄修訂歷程 (版本),以及如何還原錯誤。其實 Git 也可以分別記錄不同的修訂狀態,讓團隊分頭進行開發。能實現這項目的的功能稱為 「branch」。branch 的意思是「分支」,顧名思義,就是源自主幹的分枝。這到底是什麼 意思呢?接下來就讓我們慢慢了解吧!

該在何時使用分支功能?

前一節我們透過 Markdown 語法的文字使用了 Git,但 Git 原本是程式設計師為了開發軟體與網頁所開發的工具,除了能處理各種程式碼與檔案,也搭載了讓程式開發更具效率的功能。

在開發軟體或網頁時,常常會發生下列的情況。

- 1. 軟體開發完成與發表
- 2. 為了開發新版本而開發新功能
- 3. 在發表的版本找到錯誤,修正程式碼
- 4. 修正錯誤之後,無法於新版本套用

如果這時候沒有 Git 這項工具,就必須修正備份的舊版程式碼再發布程式,正在開發 的程式碼也得進行相同的修正,等於得修正兩次,但是這類修正很容易出錯,也不一定 每次都能看出哪個是新版的程式,哪個是舊版的程式,開發現場常因此陷入混亂。

如果有 Git 這套工具的話,就能透過「branch」功能,分成下列兩種狀態,再分頭進 行開發。



branch 功能可同時維持多個狀態,所以就讓我們趕快建立分支,體驗 branch 功能有 多麼好用吧!



接下來要啟動網站改版的專案,而這個網站會包含剛剛製作的頁面。由於必須一邊修 改網站,一邊保留原本的網站,所以要先建立分支,維持這兩個狀態。





	Branch
	🦆 New Branch 🦲 Delete Branches
Current Branch: renewal	
New Branch:	
Commit: 💽 Workin 🔿 Specifie	g copy parent
Checkor	It New Branch Create Branch Cancel

首先以滑鼠右鍵點選要新增分支的位置,也就是提交記錄。範例是在「新 增首頁了」的提交記錄按下滑鼠右鍵, 接著再點選「Branch」,新增分支, 建立新的狀態。

- 接著會顯示左圖的視窗。請在「New Branch」輸入「renewal」(①)。其他 的設定如圖所示,直接沿用即可(②)。 最後請點選「Create Branch」(③)。
 - 點選工具列的「Branch」,或是點選
 選單「Repository」→「Branch」,與 在提交記錄按下滑鼠右鍵新增分支的 方法有些不一樣。如果是透過工具列 或是選單的方式新增分支,視窗裡的 「Commit」選項會自動勾選 「Working copy parent」,這意思是 最新的提交記錄,所以會直接從最新 的提交記錄建立分支。如果不想從最 新的提交記錄就增分支,可在特定的 提交記錄按下滑鼠右鍵,或是點選 「Specified commit」,再指定要建 立分支的提交記錄。





利用分支功能分頭進行開發時,這些分支的內容通常會不同,此時可利用 Git 的功能 「合併」統整各分支的內容。接下來讓我們利用在前一節建立的 renewal 與上層的 master 分支進行開發作業,然後再試著統整這些開發內容。

試著在分支變更內容

首先要在上述的兩個分支新增變更。

Pile Edit V	'iew Repository	Actions To	ols Help			
git-practice		my-web		× + •		
+ ·	① () teh Pro	2 Iz	8	C.	S
	e Pusit Pe	All Branches	Show R	emote Branches	Date Order	' ag
File Status	L.	Graph	12 master	新増公司簡介了	Description	
History		0	o 🕩 renewal	新增首頁了		
v Le BRANCHES						
master						
O renewal	20					
> 🏷 TAGS						

	mpa	ny.htm l	C:\\my-web-cmd	↔ <h?< p=""></h?<>	1>新商品	Untitled-1	•
		1	<h1>新商品</h1>				
\mathcal{Q}							



 接著要新增新商品,所以請在 VSCode新增檔案,再追加下列的內 容,然後以「new-product.html」的 檔案名稱儲存。

			-
les	Stage All Stage Sele	SUCCESSION OF THE OWNER OF	
tahsu <baristah< th=""><th>su@gmail.com></th><th>© Commit options v</th><th>Į</th></baristah<>	su@gmail.com>	© Commit options v	Į



輸 入下 方 的 提 交 訊 息 , 然 彳 檔案 。

輸入提交訊息

1 新增新商品了

製作 HTML 檔案

<h1>新商品 </h1>



P File	Edit View	Reposito	ry Action	s Tools	Help		
git-prac	tice		my-we	eb		× + -	
+	€	1	٢	l	រៃ	8	
Commit	Pull	Push	Fetch	Branch	Merge	Stash	Di
wo	RKSPACE		All Bran	ches 🗸	Show Re	mote Branches	Date
			Grap	1			Des
File	Status		O,	0	🕼 renewal	新增新商品了	
Hist			•	\mathcal{V}	master	f増公司簡介了	
Sea	rch		¥.,	新増	首頁了		
			0			2	

確認 History(歷程)就會發現提交記錄的左側出現了左圖的樹狀圖(1)。 此外,每個提交記錄的左側都顯示了 分支名稱(2)。第一行是「renewal」 分支的開頭提交記錄,第二行是「master」分支的開頭提交記錄。

如此一來便可從樹狀圖知道是在哪個 分支提交檔案。

變更 master 分支

我們剛剛已經進行了改版的作業,假設發現目前在「master」分支發表的網站找到問題。我們希望將標題的顏色從「紅色」改成「藍色」。

4

💡 File Ed	it View	Reposito	ry Actions	Tools	Help		
git-practice	9		my-web)		× + -	
+	€	1	٩	j	រំន	8	
Commit	Pull	Push	Fetch	Branch	Merge	Stash	Di
WORK	SPACE		All Branch	ies ~	Show Ren	note Branches	Date
File Cha			Graph				Des
File Sta	itus		•	4	renewal ^采	听 增新商品了	
History	/		9	0	master	新增公司簡介了	7
Search			•	新増	首頁了		
✓ ູ່ ≯ BRANC	HES						
 master 	r						
renewa	al						
index.md		o.txt 📀	index.html C	\my-web	M × 🗠 cor	npany.html C:\\n	iy-web
C:>Users>baris 1 € <h1 sty<="" td=""><td>> OneDrive /le="colo</td><td>≥ Deskton ≥ r: ■blue"</td><td>my-web≥⇔i > tomosta<td>ndex.html 1></td><td>>@h1</td><td>]</td><td></td></td></h1>	> OneDrive /le="colo	≥ Deskton ≥ r: ■blue"	my-web≥⇔i > tomosta <td>ndex.html 1></td> <td>>@h1</td> <td>]</td> <td></td>	ndex.html 1>	>@h1]	
		I					

第一步先中斷在 renewal 分支的作 業,以及切換到 master 分支。請從 左側的分支一覽表雙點「master」分 支,切換到 master 分支。

 由於已經切換至 master 分支,請在 此時切換到 VSCode,再如下修改 「index.html」的內容,然後儲存 檔案。

^г ir	idex.html」的內容
1	<h1 style="color: red">tomosta</h1>
	ŧ.
改	寫成下列的內容
1	<h1 style="color: blue">tomosta</h1>



File Edit View Reposite olt-practice	ry Actions Tools Help	
(→ [■]) (→ (→) Commit Pull Push	P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P P	Cit-flow Remote Terminal Explorer Settings
WORKSPACE	Pending files, sorted by file status 👻 🗮 👻	Search Q
File Status	Staged files Unstage All Unstage Selected	
History Factor Image: second seco	D holeshad 📼	
	United for Board Store Internet	Select a life to view the diff

Pile Edit V	'iew Reposit	ory Action	s Tools	Help			
git-practice		my-we	b		× + •		
+ 4	\bigcirc	٢	b	ls.	8	G	\bigcirc
Commit Pul	Push	Fetch	Branch	Merge	Stash	Discard	Tag
WORKSPAC	E	All Brand Graph	thes ~ [☑ Show Rei naster	mote Branches 多正文字颜色了	Date Order Description	~
History Search		ţĴ	● ↓ 新増· 新増·	renewal 公司簡介了 首頁了	新增新商品了		
v أن BRANCHES master							
O renewal	Ņ	1					

 如此一來文字就變成藍色的了。利用 Edge 這類網頁瀏覽器開啟「index. html」即可確認結果。接下來要在這 個狀態之下提交檔案。提交訊息 如下。

輸入提交訊息

1 修正文字顏色了

到此,在 master 分支的作業就告一 段落。

4. 雙點「renewal」分支,切換至 renewal分支,繼續進行改版作業。

讓 master 分支與 renewal 分支合併

這次改版作業已經完成,但現在的 renewal 分支與 master 分支各有不同之處。

若從新增頁面的 renewal 分支打開 「index.html」 會發現標題還是紅色的,如果不合併的話,剛剛修正的部分將打回原形。



因此,各分支的開發作業若是已經完成,可進行「合併」作業,讓 renewal 分支的修改內容套用至 master 分支。





www.gotop.com.tw

GitHub 的基本操作

Git 的變更歷程可放在遠端儲存庫與其他團隊成員分享, 或是能讓多台電腦同時進行作業。最具代表性的遠端儲 存庫之一就是 GitHub,這一章要教大家取得 GitHub 的 帳號,還要介紹這個遠端儲存庫的使用方法。只要學會 使用 GitHub 的方法,Git 的世界就會變得更加寬廣。





到目前為止,我們都是獨自使用 Git。就算只是單獨使用 Git, Git 也能讓我們備份檔案,還能隨時還原錯誤的操作,應用的範圍非常廣泛。

但老實說,只在自己的電腦使用 Git 實在太可惜,因為 Git 在與「遠端儲存庫」這種網路儲存庫搭配使用之後,才能發揮真正的威力。所以接下來要帶著大家一起了解遠端儲存庫。

遠端儲存庫的功能

Section

遠端儲存庫具有下列的功能。

- 可多人同時編輯同一個程式
- 可讓其他人修改的部分同步套用至自己的檔案
- •可以請別人確認自己修改的內容,或是請別人找出問題以及修正問題
- •可以管理軟體的開發進度

只要使用 Git 與遠端儲存庫就能解決開發過程常遇到的問題,所以若是要與許多人一 起開發軟體,絕對少不了 Git 與遠端儲存庫的幫忙。

直到前一章為止,在自己電腦建立的儲存庫稱為「**本地端儲存庫**」。本書也將這種儲 存庫稱為本地端儲存庫。

使用遠端儲存庫的方法

如果想使用遠端儲存庫,可透過下列這幾種方法。

- 自行建立伺服器,再設定為 Git 伺服器
- •使用 GitHub 這種儲存庫租借服務(SaaS)
- •使用專案管理系統內建的 Git 功能

Git 是以「開源碼」這種方式開發的軟體,所以只要有伺服器電腦,連遠端儲存庫都能 自行建立(有興趣的讀者可搜尋開源碼的相關內容)。此外,也有「GitLab」這種建立 Git 伺服器的軟體,所以能輕鬆建置伺服器。



Section

3 在 Sourcetree 設定 GitHub 帳號

在 GitHub 建立的遠端儲存庫可於之前建立的本地端儲存庫「新增」,資料可透過同步 更新的方式共用。

因此,必須在 Sourcetree 設定 GitHub 的帳號。這部分稍微有點複雜,請大家一步步 跟著操作吧!

設定 OAuth

相關的細節會於 p.139 的專欄說明,但本書與 GitHub 連線的方式採用 SSH 這種方式。 SSH 是較為安全的連線方式,但是需要將「公開金鑰」新增至 GitHub 才能使用,所以設 定有些複雜,但只要完成第一次,後續就不需要再次設定,還請大家跟著說明,一步步 完成設定。第一步是設定 Sourcetree 與 GitHub 網站連動的 OAuth(參考 p.139 的專欄)。 讓我們開始設定吧!

🖶 Windows 的環境





Options								×
General	() Updates	- + Diff	Git	୍ଷର Mercurial	Custom Actions	(2) Authentication	(D) Network	
Accou	ints b - Https Sa	wed Passv	vords			I	Add	
20	itHub - htt	ps://api.g	ithub.com	/baristahsı	u6611	~	0	
	ſ	lost ——— Hosting Se	rvice: GitH	ub	2/	~	-2	
	P	referred Pro	tocol: HTTF	vyginiub.com vs	lv	-	-8	
		redentials - Authentica Userni	tion: OAuth	1		~		
			Need h	Refres elp logging i	sh OAuth Token into your account?			
					OK	Cancel		

P Edit Hosting Acc	ount	×
_ Host		
Hosting Service:	GitHub	\sim
Host URL:	https://github.com/	
Preferred Protocol:	SSH	\sim
Credentials	•	
Authentication:	OAuth	~
Username:	eclipze4book	
[Refresh OAuth Token	
P	leed help logging into your account?	
Authentication C	ĸ	
	OK Car	ncel

Delete repositories Ability to delete any adminable repository	~
Gists Read and write access	~
((j)) Notifications Read access	~
Repositories Public and private	~
Personal user data Full access	\sim
Cancel Authorize atlassian	
Authorizing will redirect to http://localhost:34106	
Not owned or operated by GitHub GitHub	ĸ

 點選右上角的「Add」之後(●),會 如圖開啟「Edit Hosting Account」視 窗。請如圖在「Hosting Service」選 擇「GitHub」(❷),再於「Preferred Protocol」選擇「SSH」(❸)。



5.

接著在底下的「Authentication」選 擇「OAuth」(①),然後點選「Refresh OAuth Token」(②)。







以團隊的方式 使用 Git 吧

假設團隊成員是透過 GitHub 使用 Git,每位成員就必須 將自己的工作內容整理得簡單易懂,方便其他的成員參 考。Git內建了許多這類整理工作內容的功能。之前我 們已經學了許多相關的功能,如果再學會本章介紹的功 能,就能更輕鬆、更靈活地使用 Git 了。



Stash 隱藏



Amend 修正



Rebase 重訂基底



guasi 擠壓



Cherrypick 撷取優點



Patch 縫補

www.gotop.com.tw

Section 5

1 利用 stash 功能暫時迴避 – stash

當你在某個分支作業時,突然需要處理另一項緊急作業。此時正在使用的分支還有沒 提交的內容。如果在這種狀態切換分支,還沒提交的內容就會套用在切換後的分支,這 麼一來,兩個分支的作業就會混在一起,後續將會衍生不少問題,但是,提交做到一半 的內容,會讓提交歷程變得凌亂難以理解。

此時可使用讓做到一半的內容暫時迴避的「**Stash**」功能。Stash 有「隱藏」的意思, 是一種跳過提交,暫時保管作業內容的功能。

前置作業

接下來要在新的儲存庫練習。

C:\Users\baris\OneD	rive\Desktop\git	-advanced	Browse
git-advanced			
Git 🗸			
Create Repository C	n Account:	-0	
Account			
GitHub: eclipze4book		~	
Dwner			
eclipze4book		\sim	

$\bigstar \equiv \qquad \leftarrow \Rightarrow$					
Ω,	contact.html C:\\my-web	↔ inde.html U ×			
	C: > Users > baris > OneDrive > I 1 <h1>tomosta</h1>	Desktop > git-advanced >			

- 在 Sourcetree 點選「File」→「Clone/ New」,再點選「Create」開啟建立儲存庫的畫面之後,建立「gitadvanced」這個儲存庫。此外,在 Windows的環境建立新的儲存庫時, 可連同遠端儲存庫一併新增(①)。請 試著如圖設定。勾選「Is Private」之後,就能建立私有儲存庫(②)。 macOS環境則需要開啟網頁瀏覽器 另外新增。
- 2. 武著在這個儲存庫新增檔案。先在 VSCode輸入下列的內容。

新增「index.html」

<h1>tomosta</h1>



2 利用 amend 新增提交記錄 – commit --amend

stash 可讓正在修改的內容暫時迴避。這次要學的是執行提交作業之後,修改與新增 內容的方法。修改到一半不小心提交,或是提交之後才發現有疏漏時,都可以使用 「amend」新增提交內容。讓我們試用這個功能吧!

提交

Section

5

第一步,要先提交欲追加內容的檔案。

X :	=	$\leftarrow \rightarrow$	○授尋
φ	檔案總管 ····	↔ contact.html C:\\my-web	☉ index.html M × 🔍
	∨ 已開啟的編輯器	C: > Users > baris > OneDrive > I	Desktop > git-advanced > Q
Q	↔ contact.html C:\	1 <h1 style="color:</th><th>red"> tomosta</h1>	
	× ⇔ index.html M	2	
90	• contact.html C:\	3 	
6 2	∨ 沒有開啟的資料夾	4 <11>公司間介 11 5	
à	您尚未開啟資料夾。	6	

1	這次要如下修改「index.html」(刪除		
1	前一節追加的「進行改版中」的訊息)。		
修改 [「] index.html」			
1	<h1 style=" color:red">tomosta</h1>		
2			
3			
4	公司簡介 		
5			

以
 是建立列表的標籤。



先故意提交一次,提交訊息為「新增
 內容列表了」。





「**Rebase (重訂基底)**」是合併功能之一,可讓提交記錄變得更加簡潔。首先,我們 來了解合併與重訂基底的差異。

事前準備

要試用 Rebase 這項功能必須在多個分支準備不同的提交記錄。接下來會介紹一連串的步驟,請大家一邊複習之前學過的內容,一邊完成事前準備。



www.gotop.com.tw