Git和GitHub

Git,GitHub是什麼

- Git?
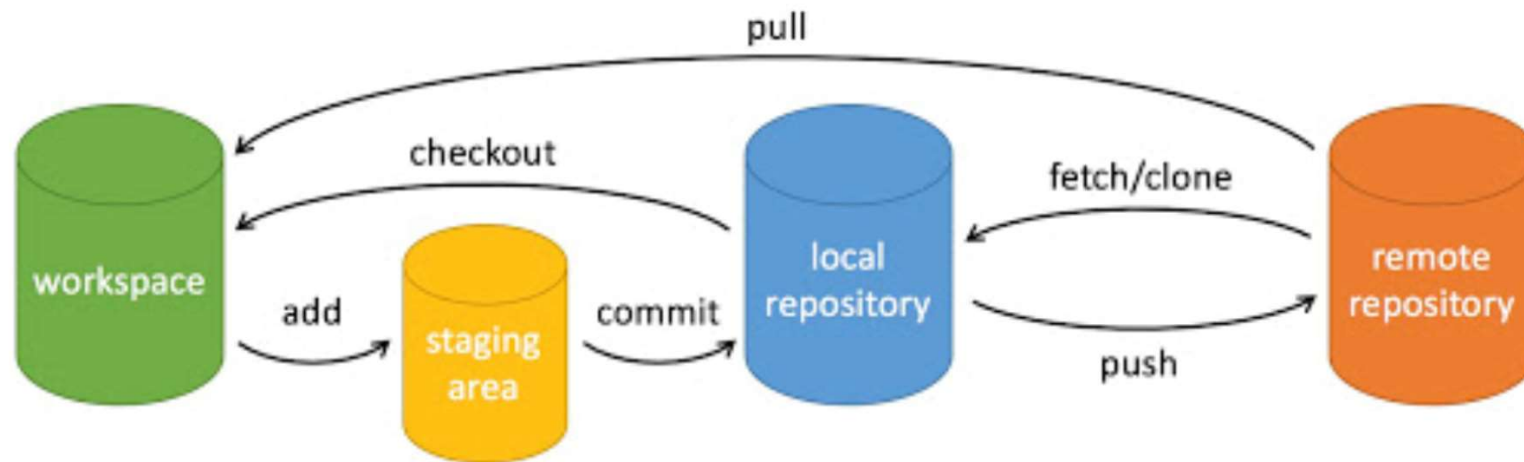
一種版本控制的工具，會記錄下程式開發的變化過程。控制文件在不同時候的內容，主要功能有儲存當前版本、恢復到特定版本、合併不同版本 ...

- GitHub?

Github是多人版本的Git平台。Github是基於Git，除了有**Git**的功能外，Github平台額外提供了雲端空間，容易看懂的網頁介面，讓成千上萬的開發人員在上面自己的Repository分享，溝通和合作寫程式完成產品。

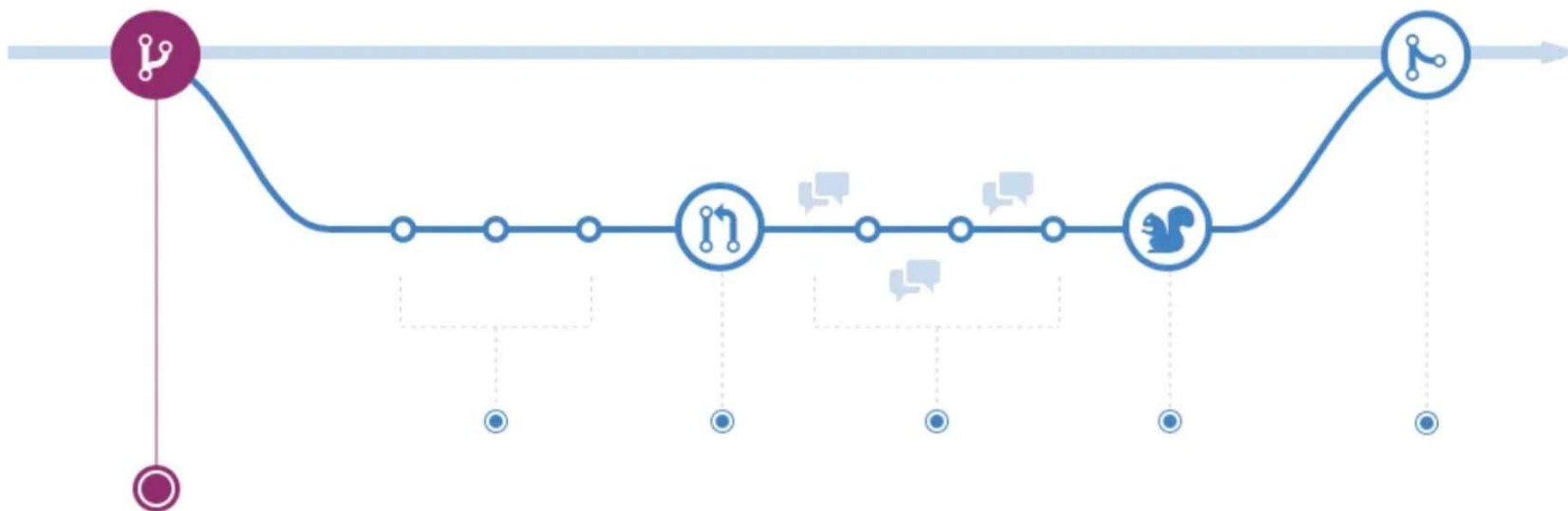# Git 中有四個區域

- 工作區 Working Directory
- 暫存區 Staging Area
- 本地儲存庫 Local Repository
- 遠端儲存庫 Remote Repository

GitHub協作開發
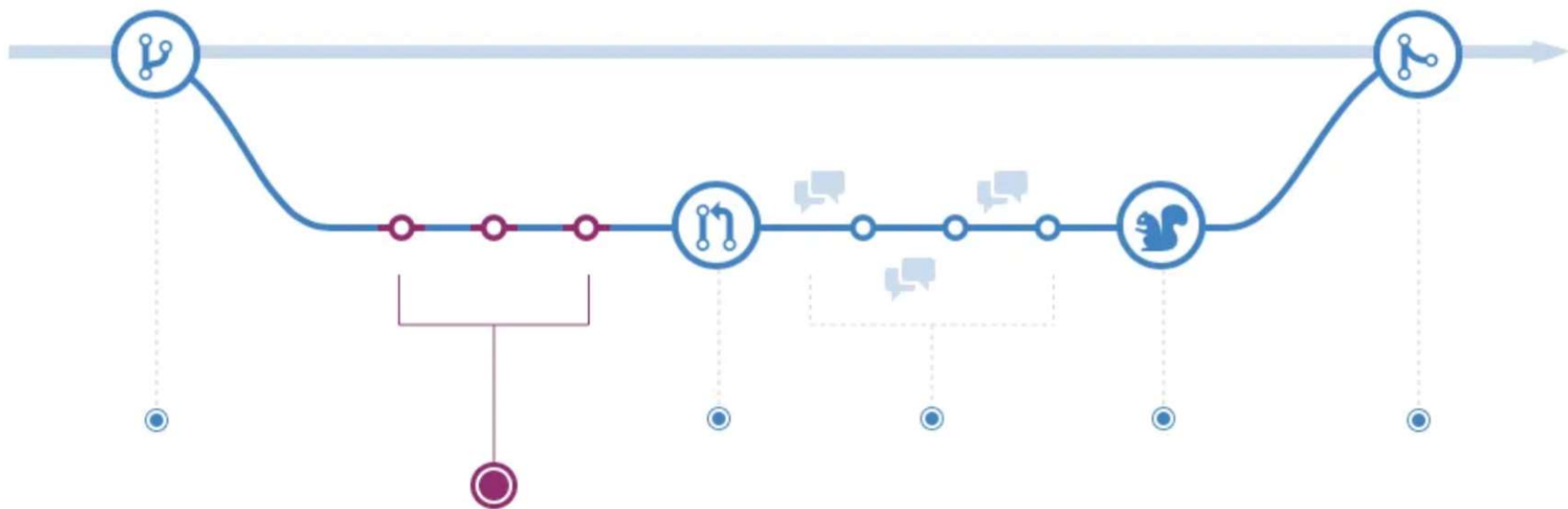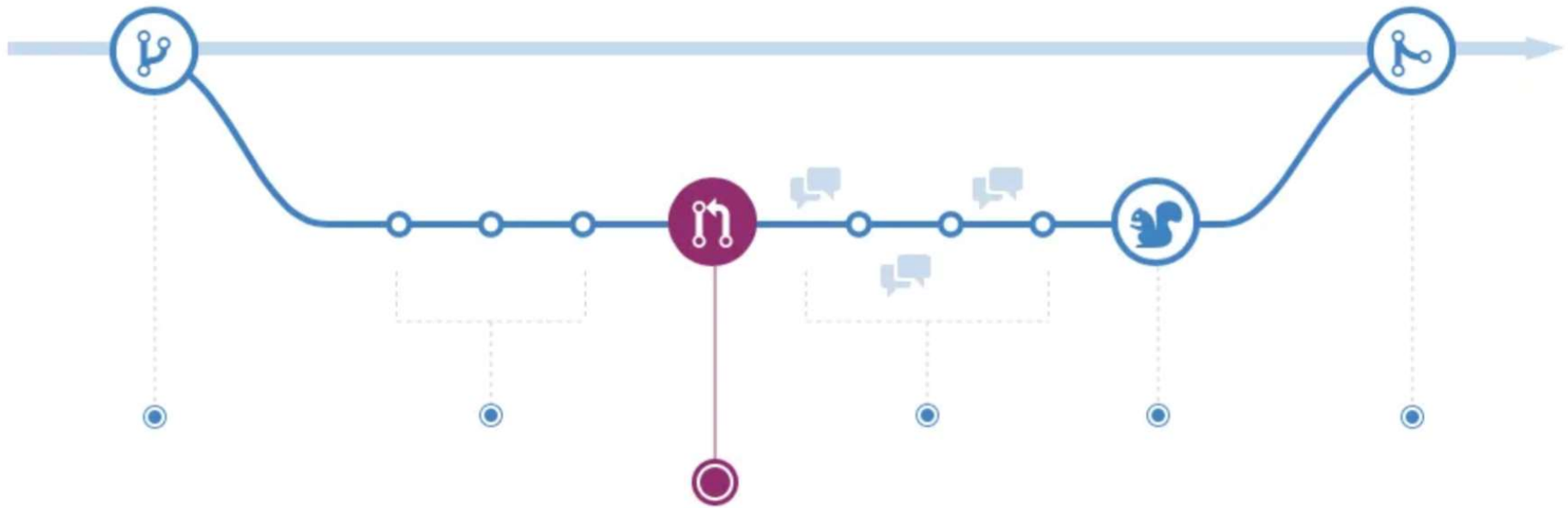
建立分支（**Branch**）

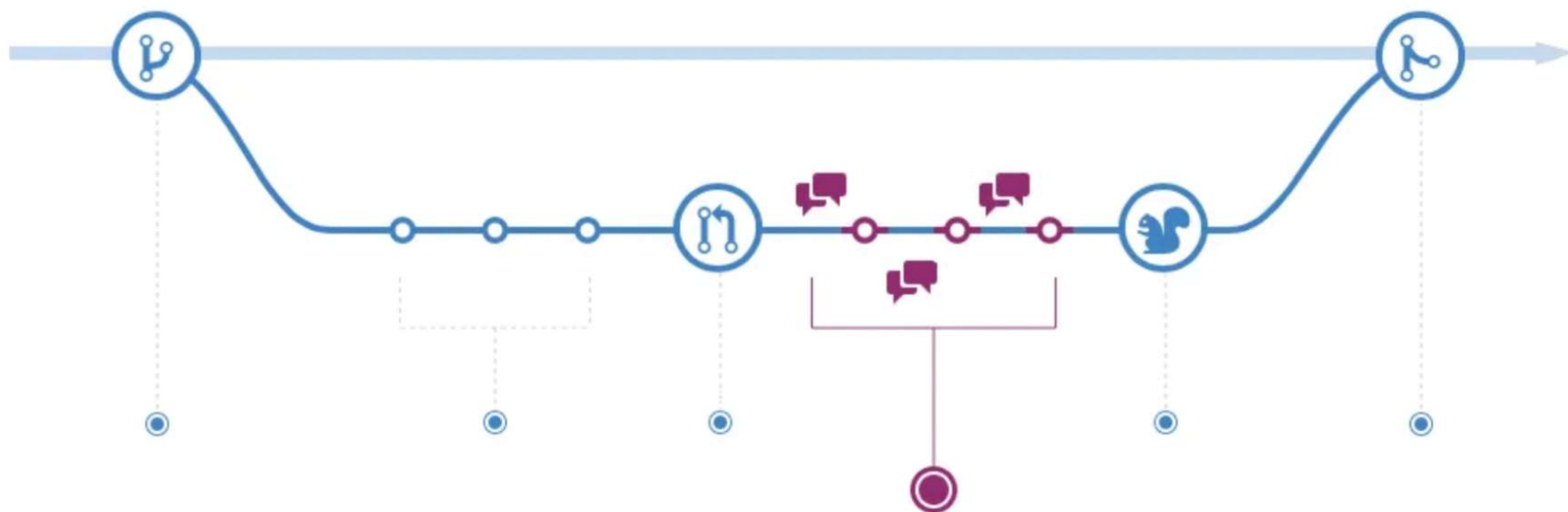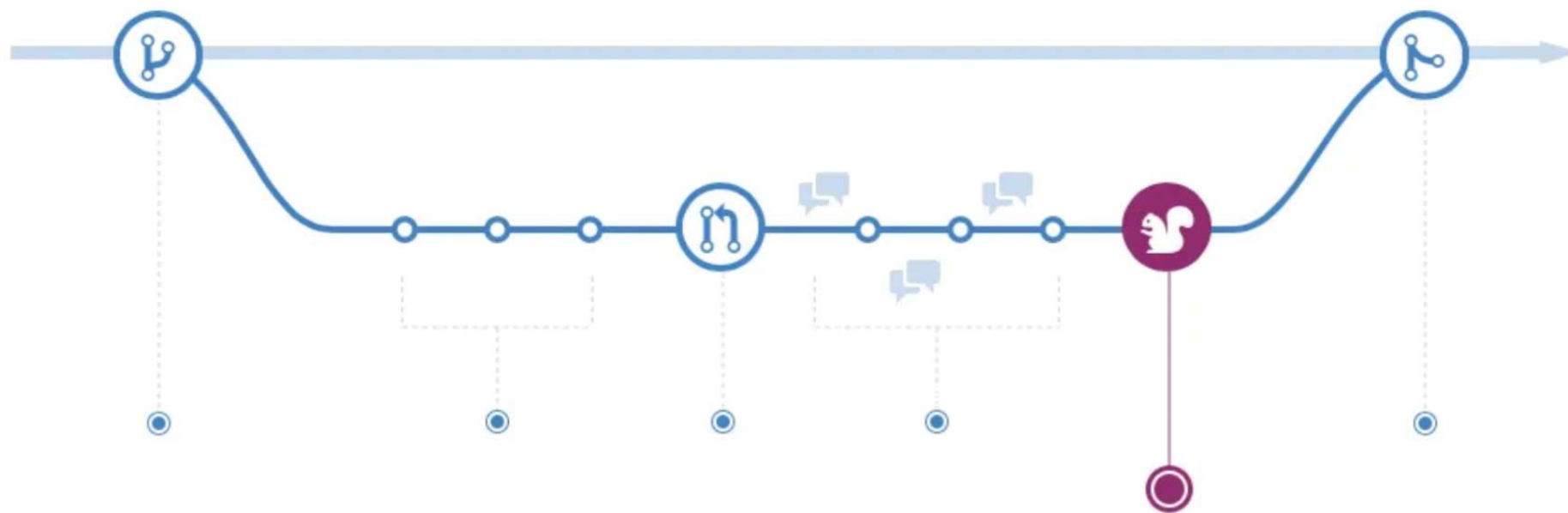# 新增提交（**Commits**）
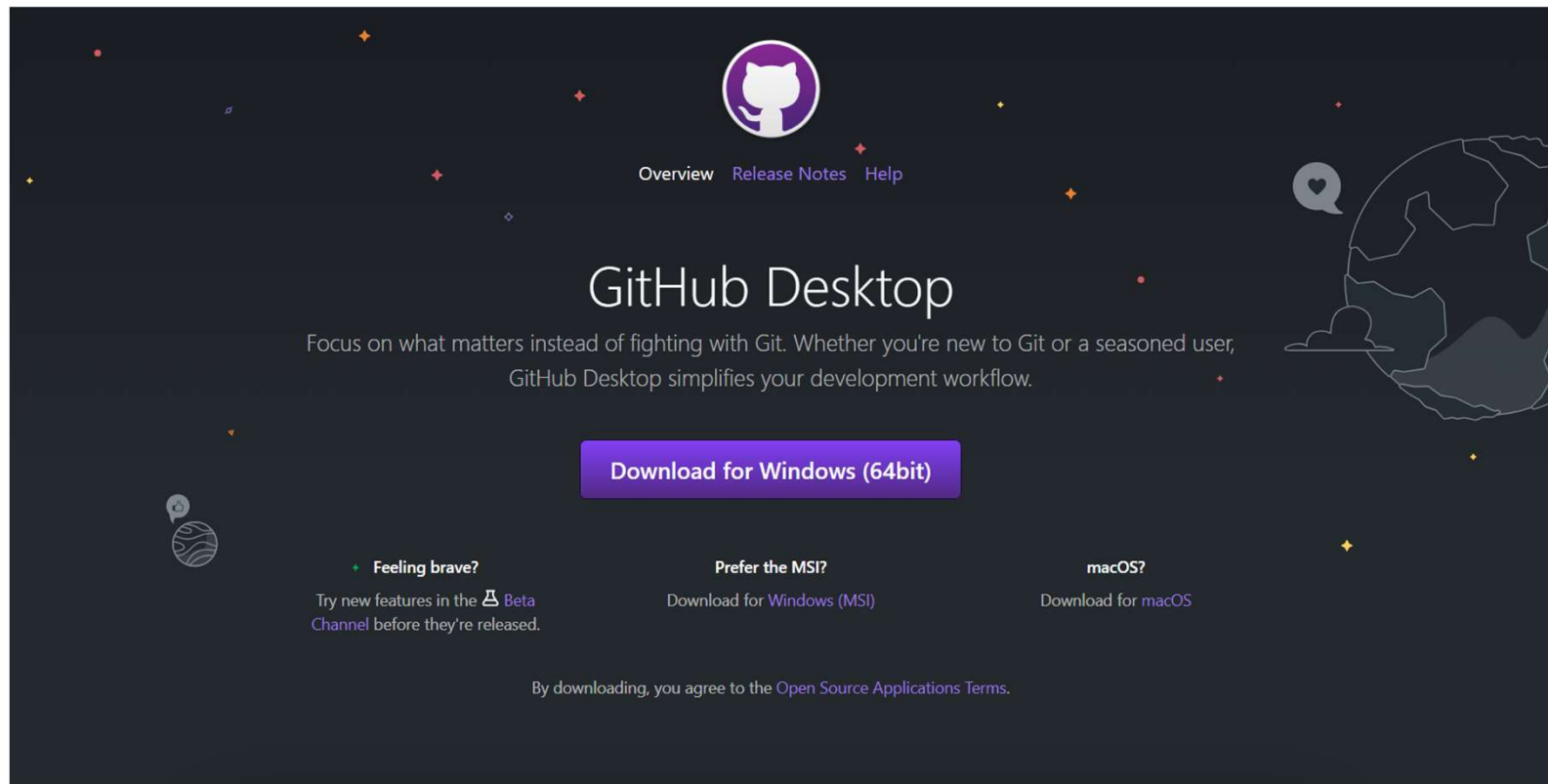
開啟 **Pull Request**

討論與審查你的程式碼

部署（**Deploy**）**(暫時不會用到)**
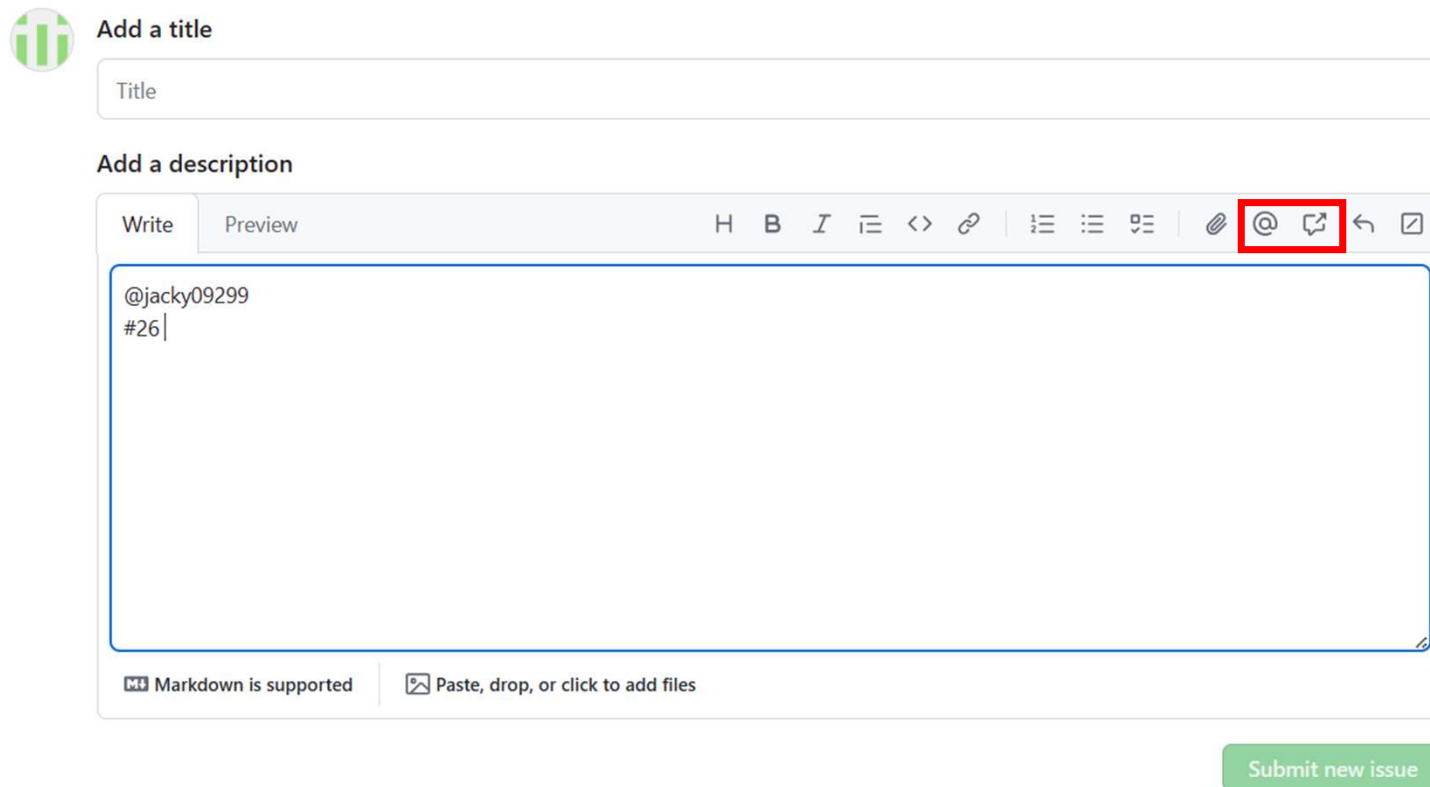
合併（**Merge**）
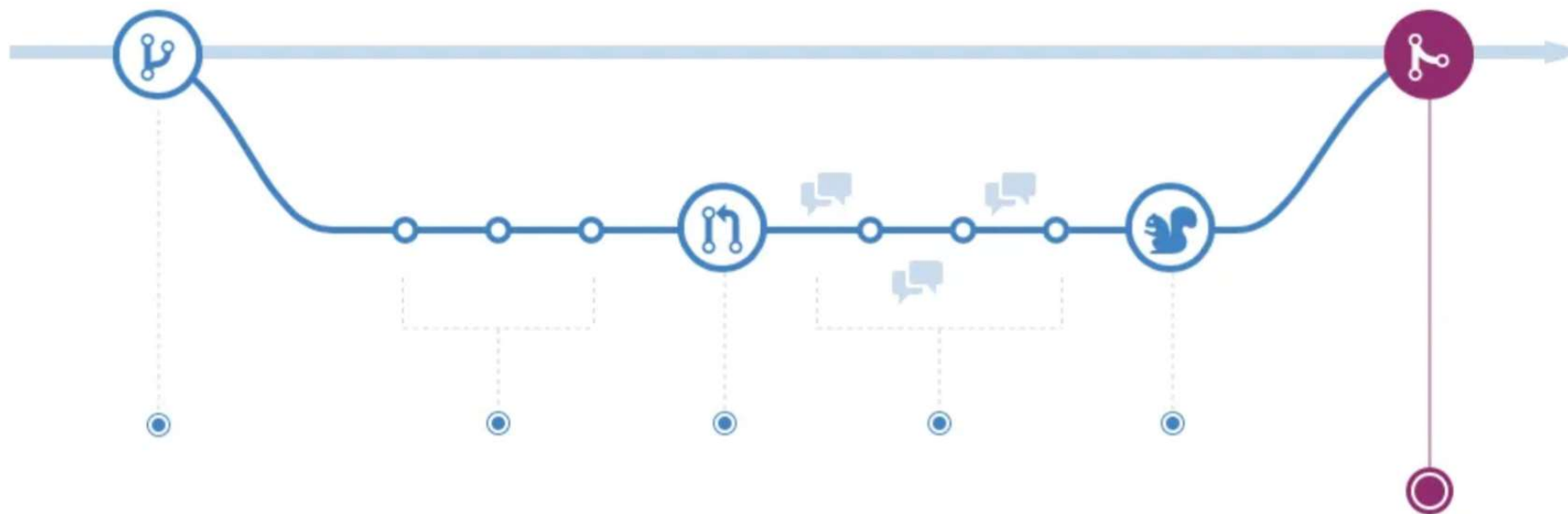
下載GitHub Desktop

協作練習

- 1.clone 儲存庫Practice
- 2..新增一個branch，publish分支，添加一個文件"姓名.txt"，完成commit、push
- 3.將branch hello merge進你的分支，發起pull request，讓main分支合併你的的分支，並指定你左邊那位為reveiwer
- 4.審查你右邊那位的pull request，並合併進main

# GitHub 發issue

- 發一個issue，標注一個人和你剛剛發的pull request

Git Download

- https://git-scm.com/downloads
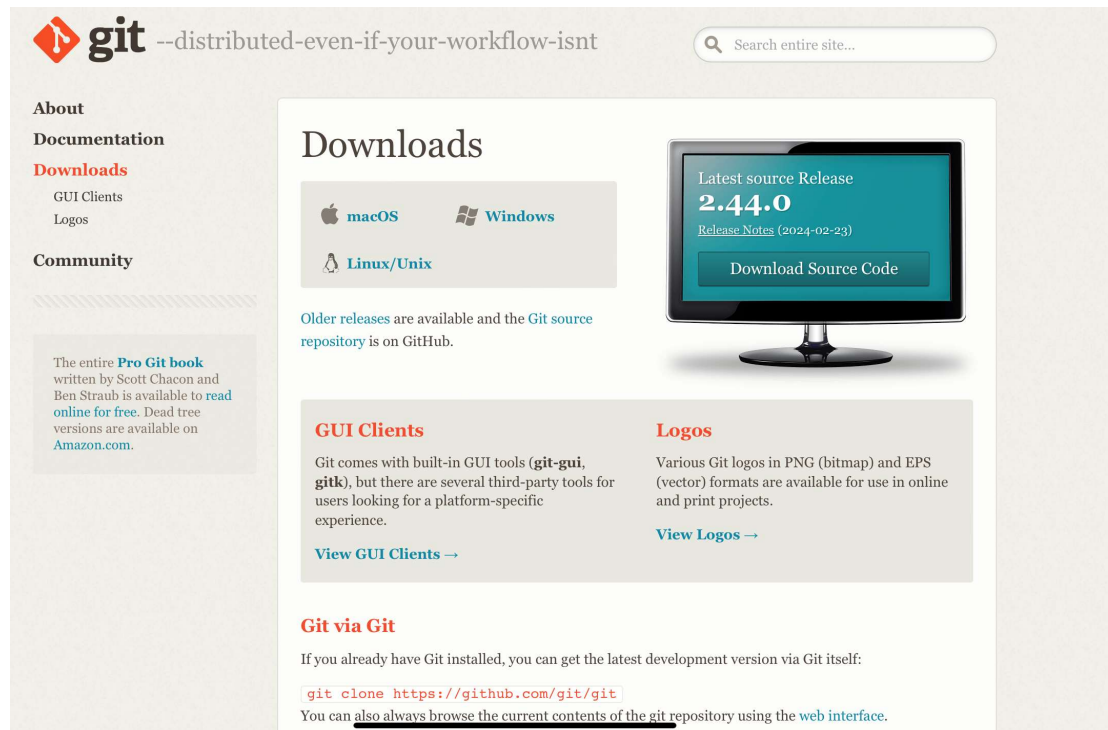
設定ssh key

- 打開cmd
- 輸入ssh-keygen
- 進入settings -> SSH and GPG keys -> New SSH key
- 進入.ssh資料夾，打開xxx.pub，複製裡面的文字
- 貼到Key方塊中，Title自訂

Add new SSH Key

Title

Key type

Authentication Key ⇕

Key

Begins with 'ssh-rsa', 'ecdsa-sha2-nistp256', 'ecdsa-sha2-nistp384', 'ecdsa-sha2-nistp521', 'ssh-ed25519', 'sk-ecdsa-sha2-nistp256@openssh.com', or 'sk-ssh-ed25519@openssh.com'

Add SSH key

jacky09299   ✕

⊞ Your projects
⊛ Your Copilot
⊟ Your organizations
⊕ Your enterprises
☆ Your stars
♡ Your sponsors
⌨ Your gists

⤒ Upgrade
⊕ Try Enterprise
⚗ Feature preview
⚙ Settings
⊞ GitHub Docs
⚊ GitHub Support

Sign out
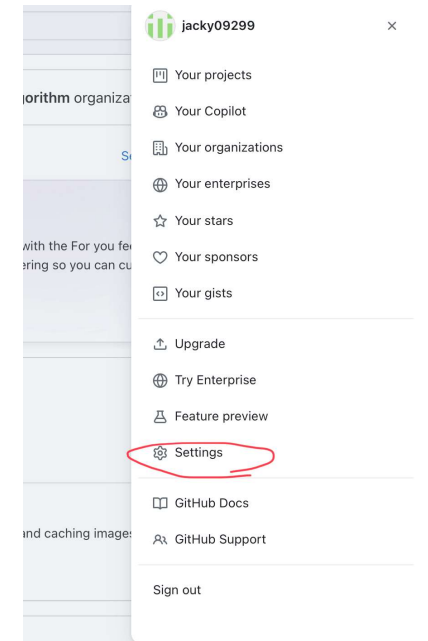
🔑 SSH and GPG keys

New SSH key

- 打開git bash
- git config --global user.name [Your Name]
- git config --global user.email [you@example.com]
- cd [路徑] ex: cd C:/Users/Jacky/Documents
- mkdir git_test
- cd git_test

Git練習

- git clone git@github.com:WingSyncAlgorithm/Practice.git
- cd Practice
- git checkout [branch-name1] 切換到自己建的分支(以下簡稱分支1)
- git branch [branch-name2] 新建一個分支(以下簡稱分支2)
- vim hello_world.py
- 按i，接著就可以編輯文件，打print("hello world")
- 按esc，輸入「:wq」，就生成了一個文件
- dir 檢查是否生成了一個hello_world.py
- git add hello_world.py
- git commit -m "create a python file"
- git push [remote-name] [branch-name2]　　　註:remote-name預設會是origin
- git checkout [branch-name1] 回到分支1
- git merge [branch-name2] 將分支2合併進分支1