

# Lab0

## Git - Branch Merge Conflict

本節目的：

- 創立新的分支(Branch) Develop 。
- 合併(Merge)分支 Develop → Master 。
- 了解如何解決分支合併時的衝突(Conflict) 。

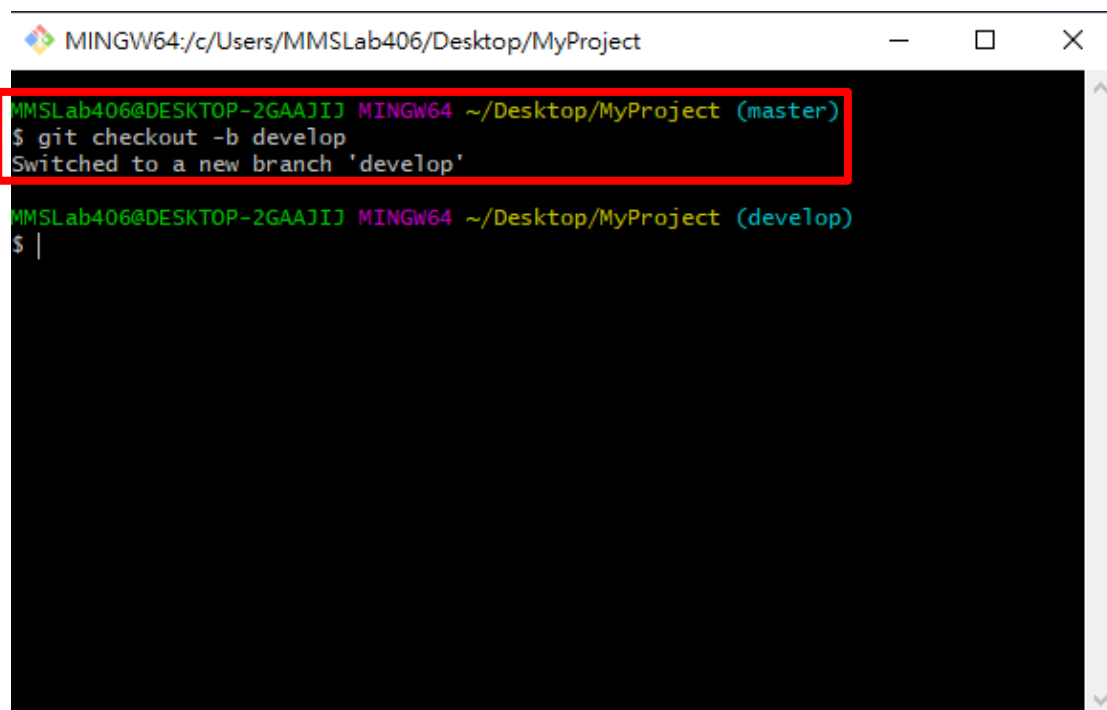
## 0.1 實戰演練

- 創立新的分支
- 合併(Merge)分支
- 解決分支合併時的衝突(Conflict)

### 0.1.1 創立新的分支

**Step1** 開啟 Lab0-1 的 MyProject 資料夾並開啟 Bash，輸入指令新增分支，並切換到新分支，如圖 0-1 所示。

```
$ git checkout -b develop
```



The screenshot shows a terminal window titled "MINGW64:/c/Users/MMSLab406/Desktop/MyProject". The prompt is "MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (master)". The user enters the command "\$ git checkout -b develop", and the terminal responds with "Switched to a new branch 'develop'". The prompt then changes to "MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (develop)". The first three lines of the terminal output are highlighted with a red rectangle.

```
MINGW64:/c/Users/MMSLab406/Desktop/MyProject
MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (master)
$ git checkout -b develop
Switched to a new branch 'develop'
MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (develop)
$ |
```

圖 0-1 新增分支並切換

**Step2** 用記事本開啟 README，並修改內容後儲存，如圖 0-2 所示。

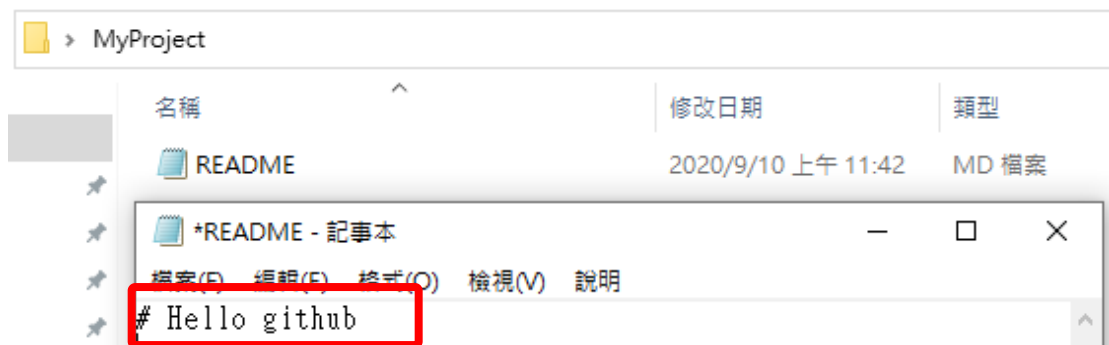


圖 0-2 修改 README 檔

**Step3** 檢查資料狀態，有一個修改的檔案，如圖 0-3 所示。

```
$ git status
```

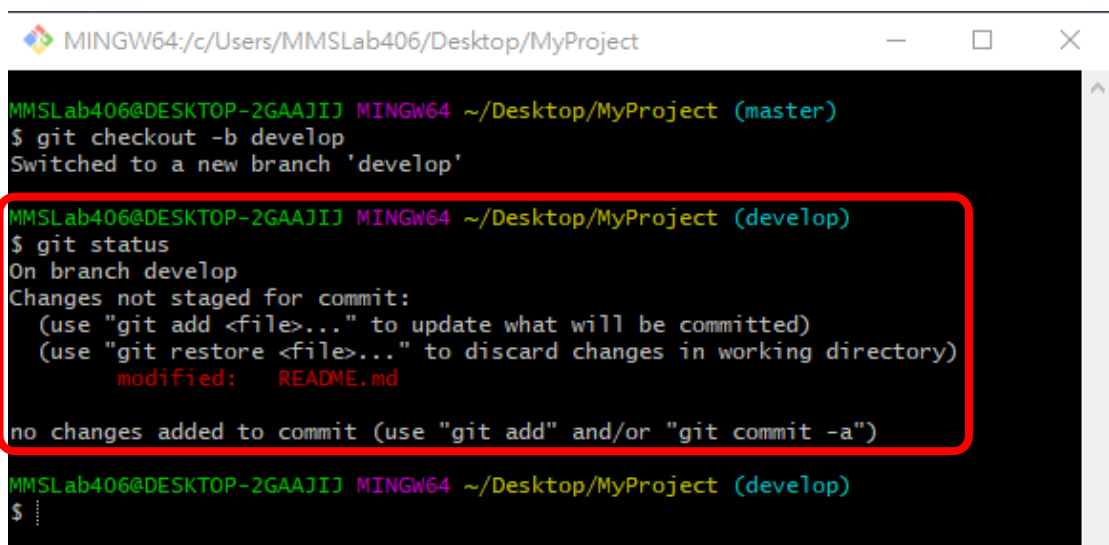
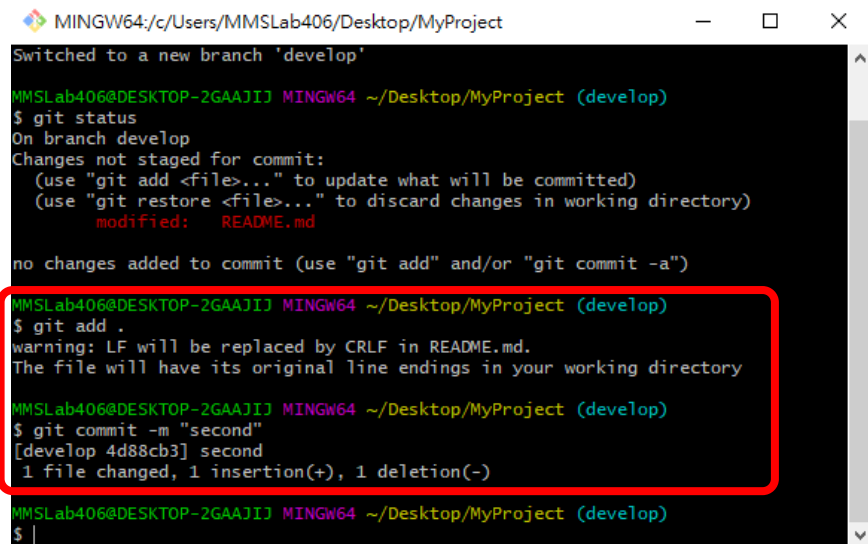


圖 0-3 檢查資料狀態

**Step4** 將檔案加入提交名單並提交，如圖 0-4 所示。

```
$ git add .
```

```
$ git commit -m "second"
```



```
MINGW64:/c:/Users/MMSLab406/Desktop/MyProject
Switched to a new branch 'develop'

MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (develop)
$ git status
On branch develop
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")

MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (develop)
$ git add .
warning: LF will be replaced by CRLF in README.md.
The file will have its original line endings in your working directory

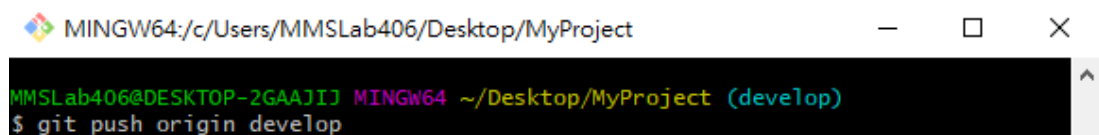
MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (develop)
$ git commit -m "second"
[develop 4d88cb3] second
1 file changed, 1 insertion(+), 1 deletion(-)

MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (develop)
$ |
```

圖 0-4 加入並提交

**Step5** 提交到遠端資料庫 ( GitHub ) 上，如圖 0-5 所示。

```
$ git push origin develop
```



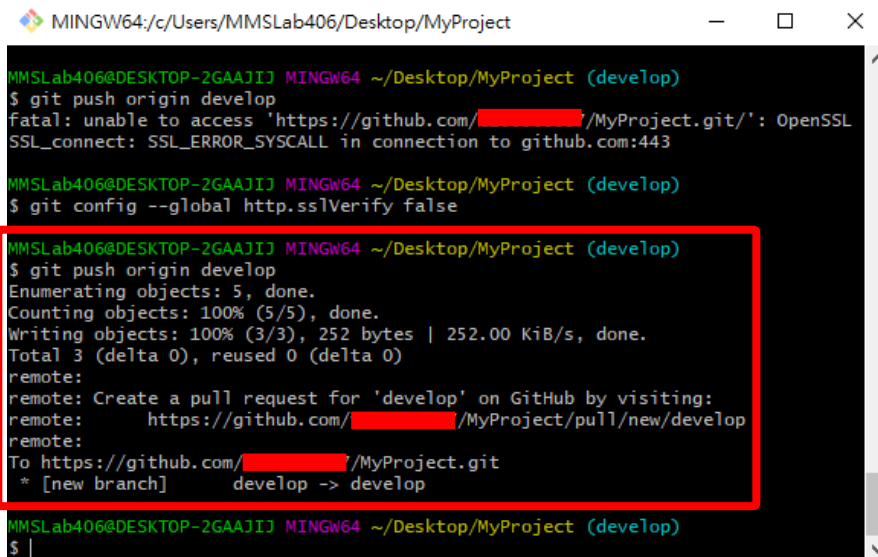
```
MINGW64:/c:/Users/MMSLab406/Desktop/MyProject

MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (develop)
$ git push origin develop
```

圖 0-5 提交到遠端資料庫

**Step6** 檔案成功推至遠端的 GitHub，如圖 0-6 所示，若遇到 unable to access 的問題，請輸入以下指令，再執行一次 Step5。

```
$ git config --global http.sslVerify false
```



```
MINGW64; c:/Users/MMSLab406/Desktop/MyProject
MMSLab406@DESKTOP-2GAAJ1J MINGW64 ~/Desktop/MyProject (develop)
$ git push origin develop
fatal: unable to access 'https://github.com/[redacted]/MyProject.git/': OpenSSL
SSL_connect: SSL_ERROR_SYSCALL in connection to github.com:443

MMSLab406@DESKTOP-2GAAJ1J MINGW64 ~/Desktop/MyProject (develop)
$ git config --global http.sslVerify false

MMSLab406@DESKTOP-2GAAJ1J MINGW64 ~/Desktop/MyProject (develop)
$ git push origin develop
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Writing objects: 100% (3/3), 252 bytes | 252.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
remote:
remote: Create a pull request for 'develop' on GitHub by visiting:
remote:   https://github.com/[redacted]/MyProject/pull/new/develop
remote:
To https://github.com/[redacted]/MyProject.git
 * [new branch]      develop -> develop

MMSLab406@DESKTOP-2GAAJ1J MINGW64 ~/Desktop/MyProject (develop)
$ |
```

圖 0-6 檔案上傳成功

**Step7** 回瀏覽器，F5 重新整理 GitHub 的頁面，並切換到 develop 分支，可以看到剛剛上傳的檔案，如圖 0-7 所示。

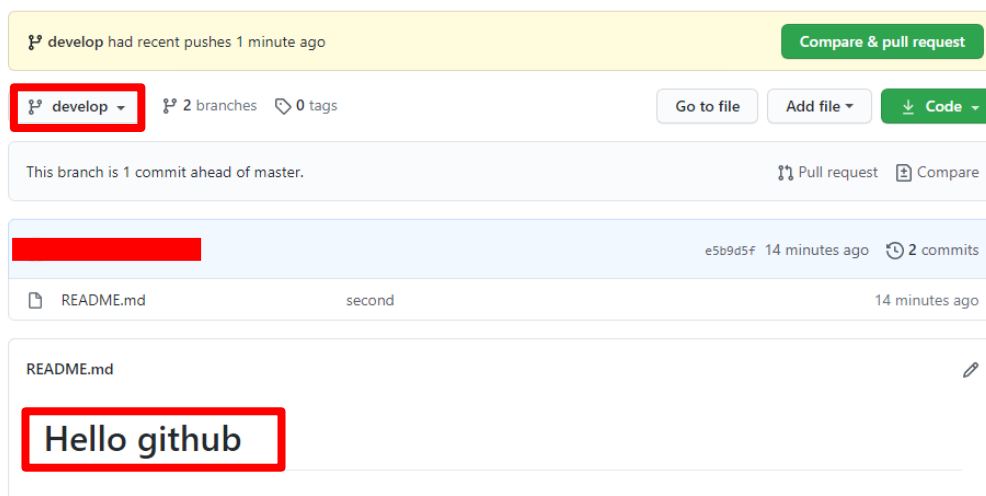


圖 0-7 查看 GitHub 上的專案

## 0.1.2 合併(Merge)分支

Step1 切換分支從 develop 到 master，如圖 0-8 所示。

```
MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (develop)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (master)
$ |
```

圖 0-8 切換分支

Step2 打開資料夾確認 README 目前的資料，如圖 0-9 所示。

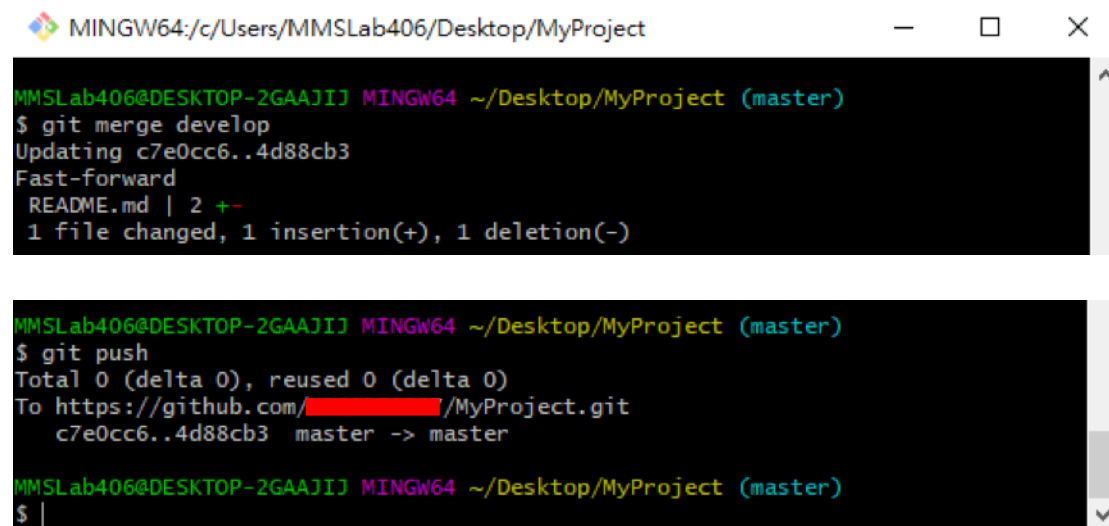


圖 0-9 確認目前資料

**Step3** 合併分支、提交版本、上傳到遠端資料庫，如圖 0-10 所示。

```
$ git merge develop
```

```
$ git push
```



The screenshot shows a Windows terminal window titled "MINGW64:/c/Users/MMSLab406/Desktop/MyProject". The terminal output is as follows:

```
MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (master)
$ git merge develop
Updating c7e0cc6..4d88cb3
Fast-forward
 README.md | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)

MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (master)
$ git push
Total 0 (delta 0), reused 0 (delta 0)
To https://github.com/[redacted]/MyProject.git
 c7e0cc6..4d88cb3 master -> master

MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (master)
$ |
```

圖 0-10 合併、提交、上傳

**Step4** 打開資料夾確認 README 目前的資料，會發現已經變成 develop 分支的

資料，這就是將 develop 合併到 master，如圖 0-11 所示。



圖 0-11

### 0.1.3 解決分支合併時的衝突(Conflict)

**Step1** 打開資料夾修改 README 的資料並儲存，如圖 0-12 所示。

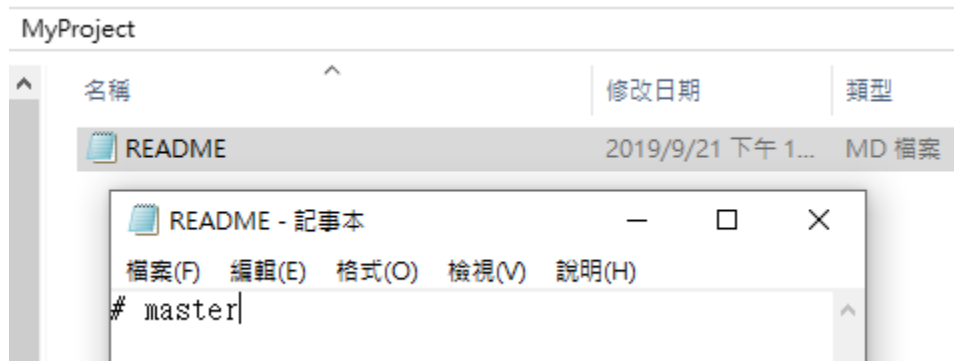


圖 0-12 修改資料

**Step2** 檢查狀態、加入提交名單、提交，如圖 0-13 所示。

```
$ git status  
  
$ git add .  
  
$ git commit -m "master"
```

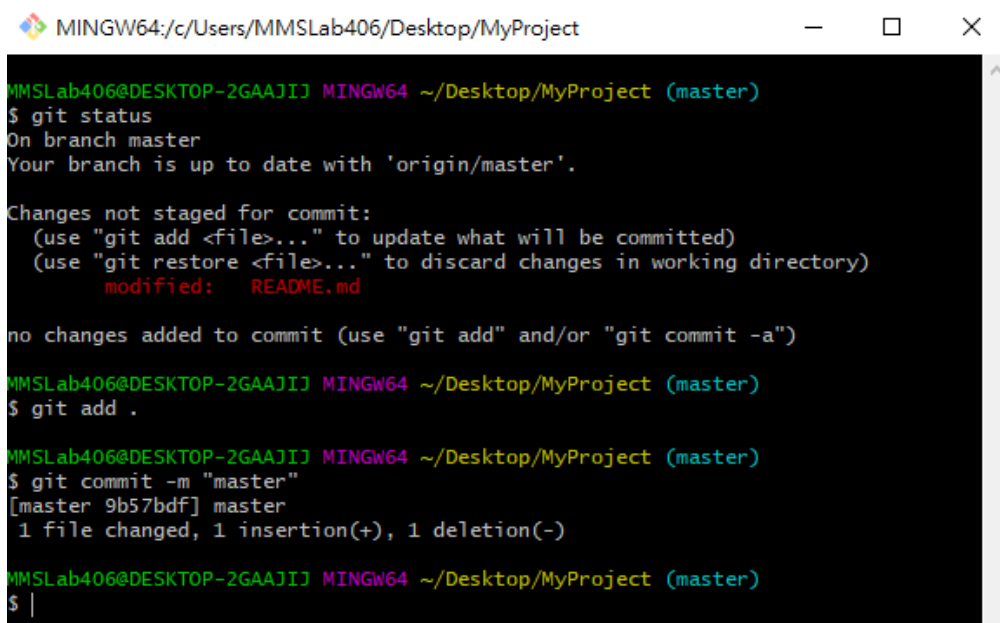
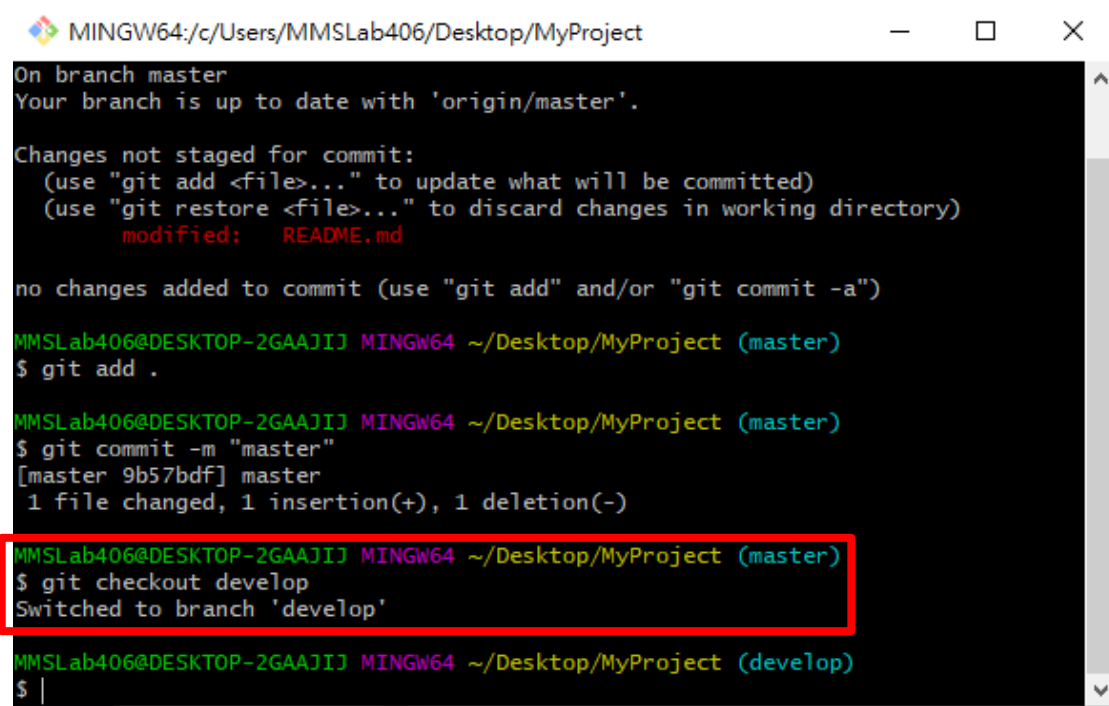


圖 0-13 檢查、加入、提交



**Step3** 切換分支到 develop，如圖 0-14 所示。

```
$ git checkout develop
```



```
MINGW64:/c/Users/MMSLab406/Desktop/MyProject
On branch master
Your branch is up to date with 'origin/master'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")

MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (master)
$ git add .

MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (master)
$ git commit -m "master"
[master 9b57bdf] master
1 file changed, 1 insertion(+), 1 deletion(-)

MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (master)
$ git checkout develop
Switched to branch 'develop'

MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (develop)
$ |
```

圖 0-14 切換分支

**Step4** 打開資料夾修改 README 的資料並儲存，如圖 0-15 所示。

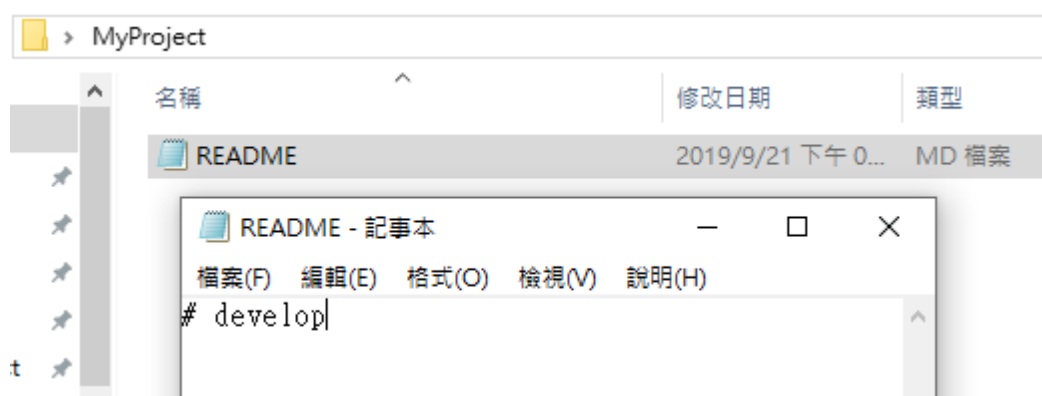


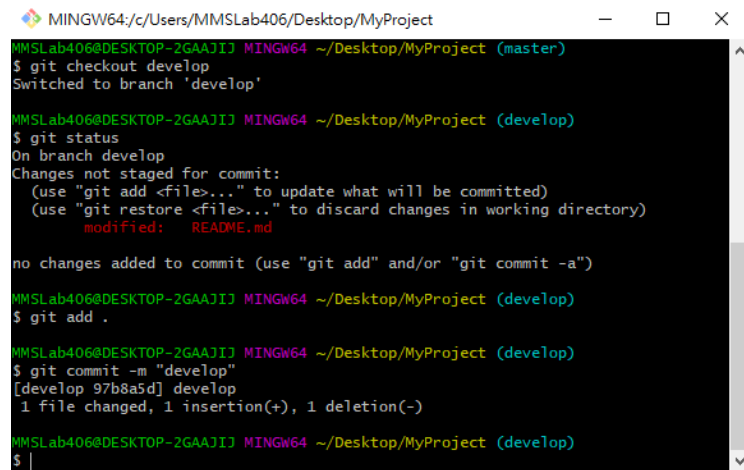
圖 0-15 修改資料

**Step5** 檢查狀態、加入提交名單、提交，如圖 0-16 所示。

```
$ git status
```

```
$ git add .
```

```
$ git commit -m "develop"
```



```
MINGW64/c:/Users/MMSLab406/Desktop/MyProject
MMSLab406@DESKTOP-2GAAJ1J MINGW64 ~/Desktop/MyProject (master)
$ git checkout develop
Switched to branch 'develop'

MMSLab406@DESKTOP-2GAAJ1J MINGW64 ~/Desktop/MyProject (develop)
$ git status
On branch develop
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")

MMSLab406@DESKTOP-2GAAJ1J MINGW64 ~/Desktop/MyProject (develop)
$ git add .

MMSLab406@DESKTOP-2GAAJ1J MINGW64 ~/Desktop/MyProject (develop)
$ git commit -m "develop"
[develop 97b8a5d] develop
1 file changed, 1 insertion(+), 1 deletion(-)

MMSLab406@DESKTOP-2GAAJ1J MINGW64 ~/Desktop/MyProject (develop)
$
```

圖 0-16 檢查、加入、提交

**Step6** 切換分支到 master 並合併 develop，會產生衝突，因為不同的分支修改了相同的地方，這時就需要解開衝突，保留必要的資料，如圖 0-17 所示。

```
$ git checkout master
```

```
$ git merge develop
```

```
MINGW64:/c/Users/MMSLab406/Desktop/MyProject
no changes added to commit (use "git add" and/or "git commit -a")
MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (develop)
$ git add .
MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (develop)
$ git commit -m "develop"
[develop 97b8a5d] develop
1 file changed, 1 insertion(+), 1 deletion(-)
MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (develop)
$ git checkout master
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 1 commit.
(use "git push" to publish your local commits)
MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (master)
$ git merge develop
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the result.
MMSLab406@DESKTOP-2GAAJIJ MINGW64 ~/Desktop/MyProject (master|MERGING)
$
```

圖 0-17 產生衝突

**Step7** 打開資料會有衝突標籤，可以看到不同分支衝突的範圍，如圖 0-18 所示。

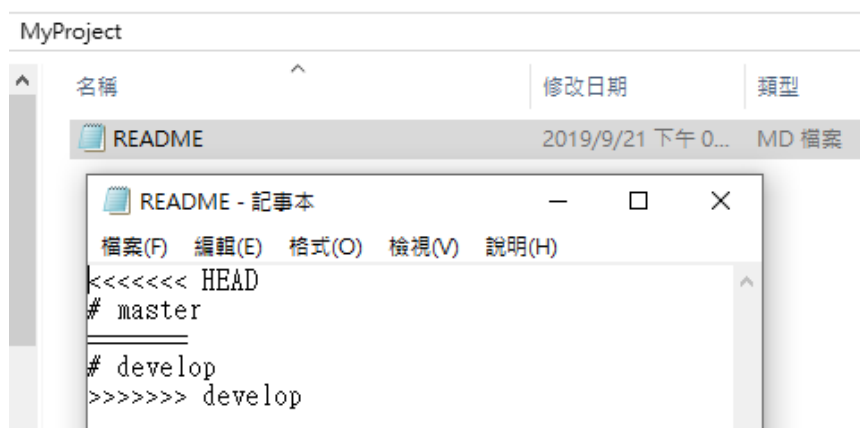


圖 0-18 衝突標籤

**Step8** 解決衝突保留 develop 的資料並儲存，如圖 0-19 所示。

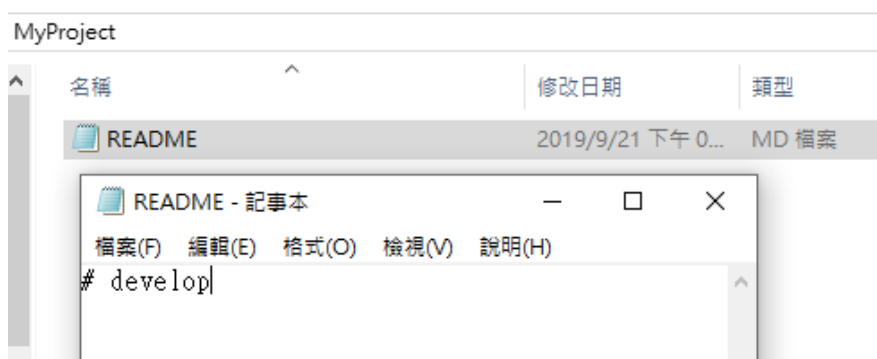


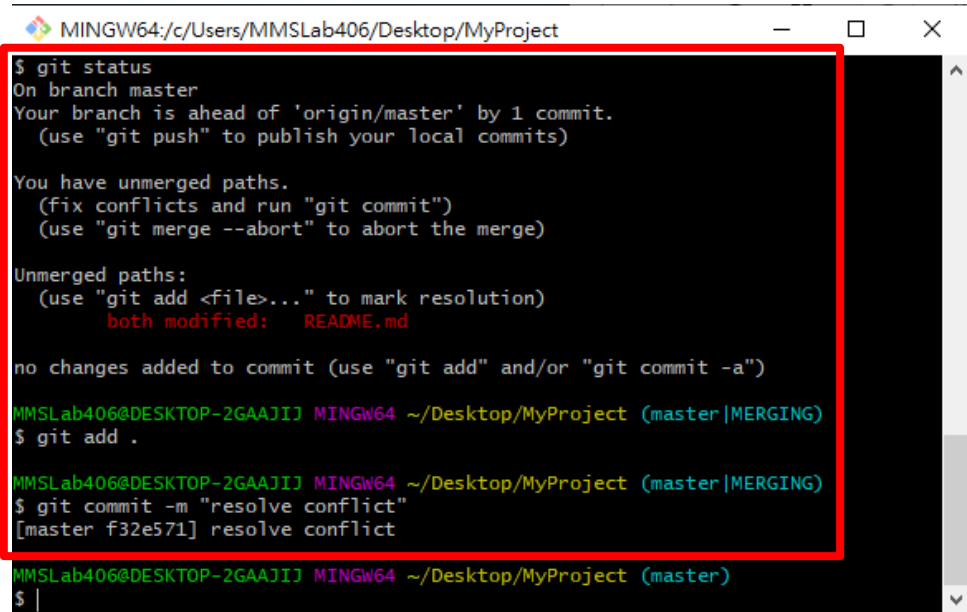
圖 0-19 解決衝突

**Step9** 檢查狀態、加入提交名單、提交，如圖 0-20 所示。

```
$ git status
```

```
$ git add .
```

```
$ git commit -m "resolve conflict"
```

A screenshot of a terminal window titled 'MINGW64:/c/Users/MMSLab406/Desktop/MyProject'. The terminal shows the output of 'git status', which indicates the user is on the 'master' branch, ahead of 'origin/master' by 1 commit. It also shows unmerged paths for 'README.md'. The user then runs 'git add .' and 'git commit -m "resolve conflict"', resulting in a new commit 'f32e571'.

```
MINGW64:/c/Users/MMSLab406/Desktop/MyProject
$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

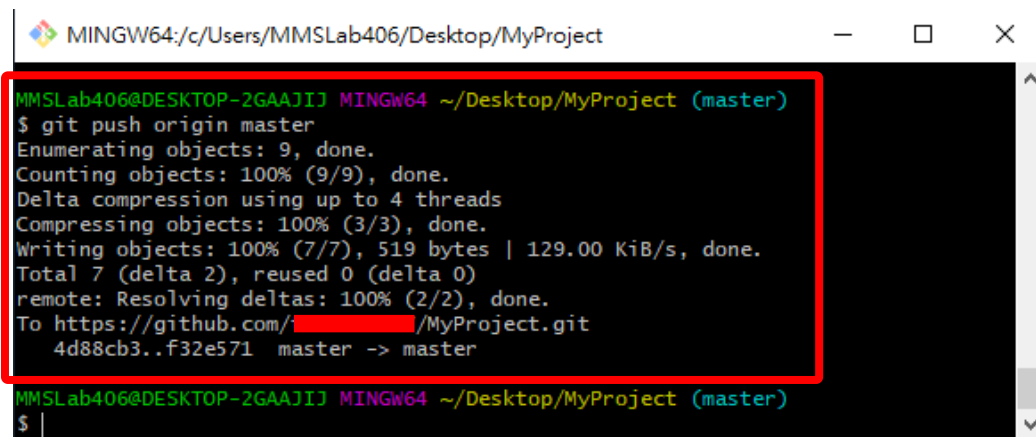
Unmerged paths:
  (use "git add <file>..." to mark resolution)
        both modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")
MMSLab406@DESKTOP-2GAAJII MINGW64 ~/Desktop/MyProject (master|MERGING)
$ git add .
MMSLab406@DESKTOP-2GAAJII MINGW64 ~/Desktop/MyProject (master|MERGING)
$ git commit -m "resolve conflict"
[master f32e571] resolve conflict
MMSLab406@DESKTOP-2GAAJII MINGW64 ~/Desktop/MyProject (master)
$
```

圖 0-20 檢查、加入、提交

**Step10** 檔案推至遠端的 GitHub，如圖 0-21 所示。

```
$ git push origin master
```



```
MINGW64:/c/Users/MMSLab406/Desktop/MyProject
MMSLab406@DESKTOP-2GAAJ1J MINGW64 ~/Desktop/MyProject (master)
$ git push origin master
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (7/7), 519 bytes | 129.00 KiB/s, done.
Total 7 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/[redacted]/MyProject.git
 4d88cb3..f32e571  master -> master
```

圖 0-21 檔案上傳成功

**Step11** 回瀏覽器，F5 重新整理 GitHub 的頁面，可以看到剛剛上傳的檔案，如

圖 0-22 所示。

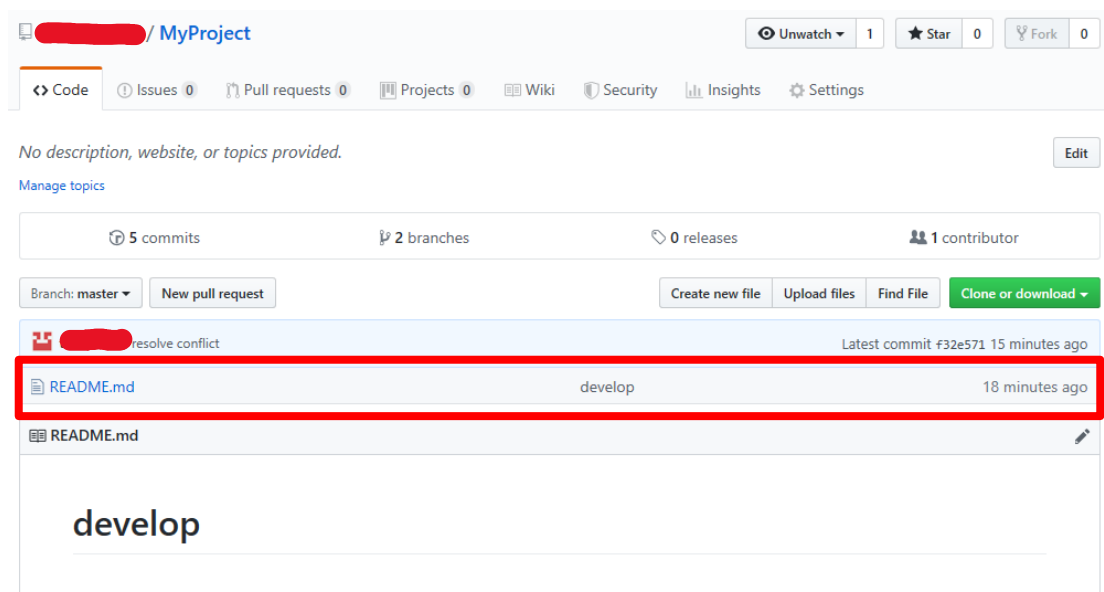


圖 0-22 查看 GitHub 上的專案

**Step12** 請開啟分支圖讓助教檢查，如圖 0-23 所示。

<> Code

! Issues

🔗 Pull requests

▶ Actions

📁 Projects

📖 Wiki

🛡 Security

**📊 Insights**

⚙ Settings

Pulse

Contributors

Community

Traffic

Commits

Code frequency

Dependency graph

**Network**

Forks

### Network graph

Timeline of the most recent commits to this repository and its network ordered by most

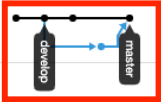
Owners	Sep
t108368007	10 

圖 0-23 查看分支圖