

CS113 - Git - 2

(Branch Merge Conflict)

conflict เกิดจาก การปรับแก้ ไฟล์ข้อมูล ใน ไฟล์ เดิม บน บัรช เดิม ที่ อยู่ ต้น branch

ถ้า มี conflict ต้องเลือก ที่จะ ใช้ข้อมูล ของ branch ใด

การสลับ / แยก Branch

คำสั่งสลับ Branch.

git Branch (ชื่อ branch)

การสลับ branch แยก ออกมา master เป็น การแยกส่วนพัฒนา ออกจากกัน ไม่ได้อยู่บนกัน เพื่อตรวจสอบ check point ที่เป็นส่วนใหม่

ตรวจสอบ branch

git branch อยู่ branch ใด * ของ เปลี่ยน branch

git checkout ชื่อ branch

* จะ เปลี่ยน ไปอยู่ ที่ ที่ เราเลือก

การ Merge Branch แบบไม่มี conflict

```
$ git checkout feature
..... เพิ่มไฟล์ feature1.txt
$ git status (สังเกตบรรทัดแรก จะบอกว่าอยู่ branch ใดอยู่)
$ git add feature1.txt
$ git commit -m "Add feature1.txt in branch feature"
```

```
$ git checkout master (สังเกตว่าไฟล์ feature1.txt จะหายไป)
..... เพิ่มไฟล์ master1.txt
$ git status
$ git add master1.txt
$ git commit -m "Add master1.txt in branch master"
```

git log --all --decorate --oneline --graph
ใช้ ดู checkpoint ทั้งหมด

push master ขึ้น github

```
git checkout master
git log --all --decorate --oneline --graph
git push origin master
git log --all --decorate --oneline --graph
```

push feature ขึ้น github

```
git checkout feature
git log --all --decorate --oneline --graph
git push origin feature
git log --all --decorate --oneline --graph
```

merge นำ feature ให้นำมา กับ master

เปลี่ยน ไปที่ branch ปลายทางก่อน

git checkout master

รวม branch (merge) feature
นำ ไปอยู่ ใน master

git merge feature

ถ้า ไม่มี conflict จะทำให้ git สลับ commit ของการ merge นั้นไปเอง

เมื่อ merge เสร็จ จะเห็นว่า มี ไฟล์ ของ master และ feature อยู่ ใน branch master

push master ขึ้น github

git push origin master

การ pull แบบไม่มี conflict

นำ clone git repository มาไว้ที่เครื่อง หรือ directory ใด

เพิ่มไฟล์ ใน directory แล้ว push ไปที่ github,

กลับมาที่ เครื่อง, นำที่ directory ที่ แล้ว pull การเปลี่ยนแปลง มาที่ local repository

git pull origin master

ถ้ามีการเปลี่ยนแปลง จาก branch master