

mighty21:47

ตอนทำ feature

1. git checkout develop

2. git pull

3. git checkout -b feature/... or git checkout -b bug/... or git checkout -b qol/... (branch ที่จะทำ)

ตอนจะ PR

1. git checkout develop

2. git pull

3. git checkout feature/... (branch ที่ทำ)

4. git rebase develop

5. resolve conflict (ถ้ามี)

6. git push -u origin feature/... (branch ที่ทำ)

คนตรวจ PR

1. git checkout develop

2. git pull

3. git fetch origin feature/... (branch ที่ตรวจ)

4. git checkout feature/... (branch ที่ตรวจ)

5. git rebase develop

6. resolve conflict (ถ้ามี)

7. git push --force origin feature/... (branch ที่ตรวจ)

8. merge PR to develop


rebase syntax แบบเดียวกับ merge เลยแต่มันได้ commit ที่ clean กว่า

6 commits3 files changed2 contributors

Commits on Apr 24, 2025



add feat1

 stmighty committed 2 hours ago


 e6fa5cc 



feature/feat1

 stmighty authored 2 hours ago


Verified 3d45168 


add feat2

 stmighty committed 2 hours ago

 a645a4d 


feature/feat2

 stmighty authored 2 hours ago

Verified b3db48b 


add feat3

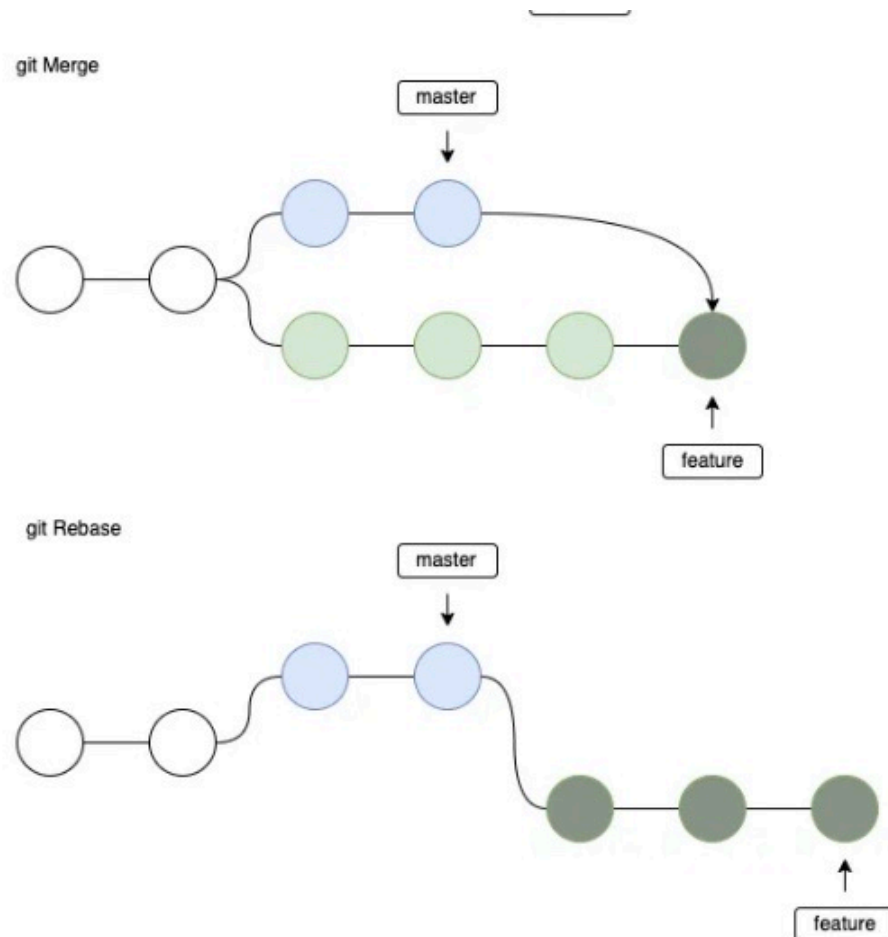
 stmighty committed 2 hours ago

 5eddcf3 

feature/feat3

 stmighty authored 2 hours ago

Verified 457308d 



ถ้าหากเรา commits change ดังภาพด้านบน git Merge จะทำหน้าที่รวม commits ต่างๆที่เกิดขึ้นภายใน feature branch เข้ากับ main branch ส่วน git Rebase จะทำการเพิ่ม commits change ที่เกิดขึ้นล่าสุดใน main branch รวมเข้ากับ feature branch

เมื่อเราทำการ git rebase feature เข้ากับ main branch เราจะย้าย base ของ feature ไปยัง master branch

git Merge จะเอา content ทั้งหมดของ feature และ main branch เข้าด้วยกัน ทำให้ main branch มีการเปลี่ยนแปลงที่เดียว ส่วน feature สามารถลบได้ หรือ มีเก็บไว้แค่ history

ในกรณีที่เราไม่อยากจะ commit logs ทั้งหมดที่เกิดขึ้นใน feature branch ให้ใช้ — squash เพื่อทำการรวม logs ให้เหลือแค่ log เดียว ใน main branch จะได้ดูไม่รกจนเกินไป

```
$ git merge --squash feature01
```