ในปัจจุบันการเรียนรู้การใช้งาน git ได้ง่ายๆ เนื่องจากมีเอกสารและตัวอย่างการใช้งานมากมาย และอีกทางที่ขอแนะนำคือ การทดลองใช้งานผ่าน browser ซึ่งพัฒนาโดย github.com นั่นก็คือ try.github.com เป็นการใช้งานแบบ interactive ซึ่งจะสอนให้รู้จักชุดคำสั่งต่างๆ ของ git ตลอดจนการใช้งานผ่าน github ตัวอย่างชุดคำสั่งที่น่าสนใจคือ

git init

git status

git add

git commit

git clone

git push

git pull

เป้าหมายหลักของ try.github นี้ไม่ใช่เพียงแค่ developer เท่านั้น แต่ยังต้องการให้คนทั่วไปหรือคนที่อยู่ในทีมทาง IT ให้เข้าใจการใช้งานพื้นฐานของ git แล้วเราจะมัวเสียเวลากันทำไม Let's go

หลังจากที่ผมทดลองใช้งานมาแล้ว สามารถสรุปชุดคำสั่งต่างๆ ได้ดังนี้

1. git init

เป็นคำสั่งเริ่มต้นใช้งาน git มาควบคุมเวอร์ชั่นของ files และ folders ในตำแหน่งที่อยู่ปัจจุบัน โดยจะสร้าง folder .git ขึ้นมาซึ่งเป็น hidden folder ซึ่ง folder นี้จะเก็บข้อมูลต่างๆ ของ git repository ไว้

2. git status

เป็นคำสั่งใช้ในการดูว่ามีการเปลี่ยนแปลงเกิดขึ้นหรือไม่

3. git add <file name>

เป็นคำสั่งการเพิ่ม files หรือ folders ที่สร้างขึ้นมาใหม่เข้าไปยัง git repository

4. git commit -m '<comment message>'

เป็นคำสั่งเพื่อยืนยัน file ที่มาจากคำสั่ง add เพื่อเข้าไปเก็บใน git repository จริงๆ

5. git log และ git log --sum­mary

เป็นคำสั่งเพื่อดูประวัติการ commit ต่างๆ ใน git repository

6. จะให้เราทำการ login เข้าใช้งาน github.com ดังนั้นจำเป้นจะต้องสมัครและทำการ authentication ผ่าน OAuth โดยเป้นการสอนให้รู้จักใช้งาน remote repository นั่นเอง

7. เมื่อทำการ login เรียบร้อย ทางระบบจะสร้าง repository ชื่อว่า try\_git ขึ้นมา

8. git remot­e add origi­n git@g­ithub.com:<username>/try\_gi­t.git

เป็นคำสั่งใช้สำหรับการเพิ่ม remote repository ขึ้นมาชื่อว่า origin เพื่อนำมาใช้งานต่อไป

9. git push -u origi­n maste­r

เป็นคำสั่งส่ง files และ folders จาก repository ของเราที่ทำการ commit ไว้ ไปยัง remote repository หรือ github.com นั่นเอง

10. git pull origi­n maste­r

เป็นคำสั่งสำหรับดึง files และ folders ต่างๆ ที่มีการเปลี่ยนแปลงลงมาที่เครื่องของเรา

11. git diff HEAD

เป็นคำสั่งใช้ตรวจสอบว่ามีการเปลี่ยนแปลงอะไรที่แตกต่างจากสิ่งที่เรา commit ไว้บ้าง

12. git reset­ <file name>

เป็นคำสั่งสำหรับการยกเลิก stated ของ file ที่เราต้องการออกไป ก่อนที่จะ commit

13. git branch <branch name>

เป็นคำสั่งสำหรับการสร้าง branch ขึ้นมาใหม่ ซึ่งสร้างเพื่อนำไปใช้งานตามที่เราต้องการ เช่น ทำการแก้ไขส่วนต่างๆ โดยไม่ไปกระทบคนอื่นๆ

14. git checkout <branch name>

เป็นคำสั่งสำหรับเปลี่ยนการใช้งานไปยัง branch นั้นๆ

15. git rm <file name>

เป็นคำสั่งสำหรับลบ files และ folders ต่างๆ ออกไปจาก git repository และจาก storage อีกด้วย

16. git merge <branch name>

เป็นคำสั่งสำหรับการ merge files และ folders จาก <branch name> กับ current branch หรือ master ( จากตัวอย่าง )

17. git branch -d <branch name>

เป็นคำสั่งสำหรับลบ branch ที่ต้องการ

18. git push

เป็นคำสั่งสำหรับส่ง files และ folders ที่ commit เรียบร้อยไปยัง remote repository