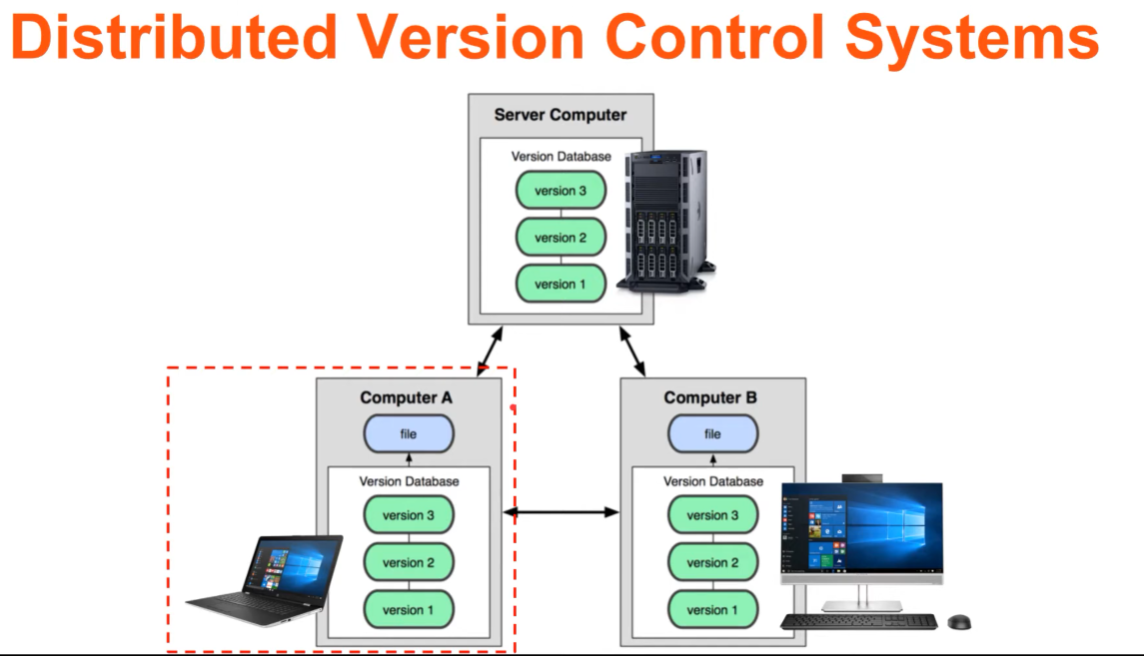
**การใช้งาน Git ด้วยคำสั่ง command line**

ในบทนี้เราจะมาลองใช้ Git โดยไม่ผ่าน Git Desktop กัน เนื่องจากเวลาเราทำงานกันจริงๆ เราสามารถใช้ VS code ในการทำงานคู่กับ Github ได้เลย โดยไม่ต้องเสียเวลาผ่านโปรแกรมตัวที่ 3 แต่ความยากของมันก็คือเราจะต้องจำคำสั่งในการใช้งาน Git ให้ได้

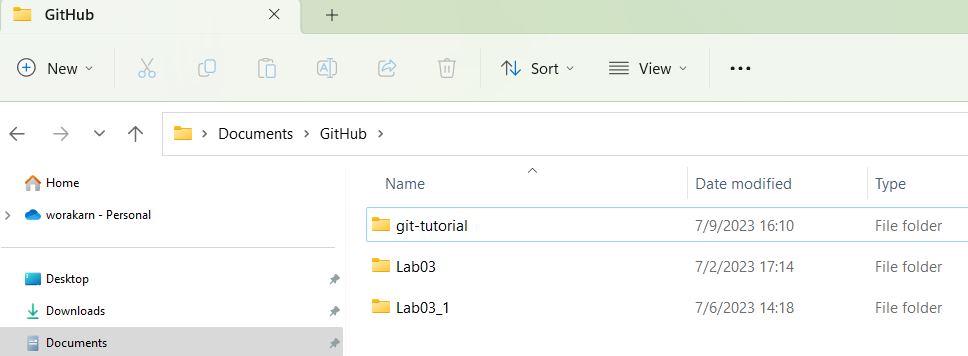
**Local Repository**

เวลาที่เราเริ่มสร้าง Project นั้นปกติแล้ว เราจะทำในเรื่องเราก่อน แล้วค่อย Upload ไปที่ Github ดั้งในนั้น Lab ครั้งนี้เราจะสร้างไฟล์ต่างๆ ไว้ในเรื่องเราก่อนจากนั้นก็จะเรียนรู้คำสั่งต่างๆ ของ Git กับส่วนของ Local นะครับ

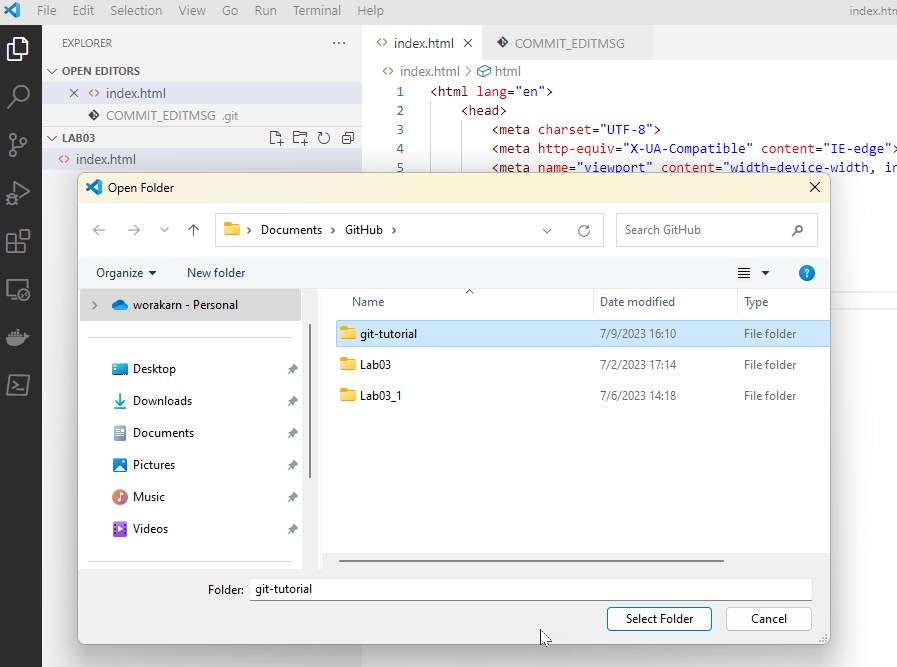


ทำงานกันส่วนนี้

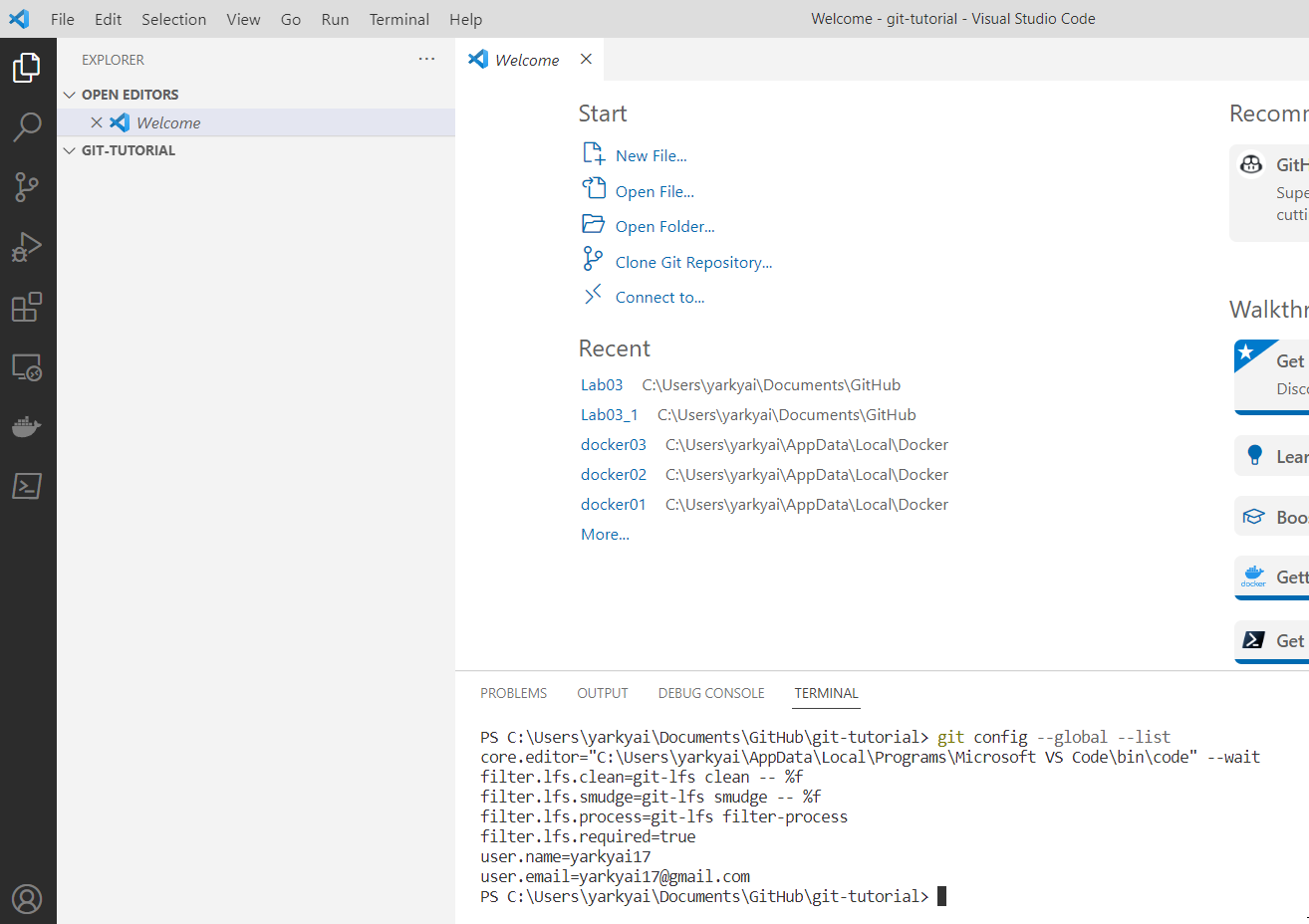
สร้าง folder git-tutorial ที่ root ของ git (ของผมอยู่ที่ C:\Users\yarkyai\Documents\GitHub)



เปิด VS Code เลือก File และ Oper folder แล้วเลือก git-tutorail ที่เราได้สร้างเอาไว้

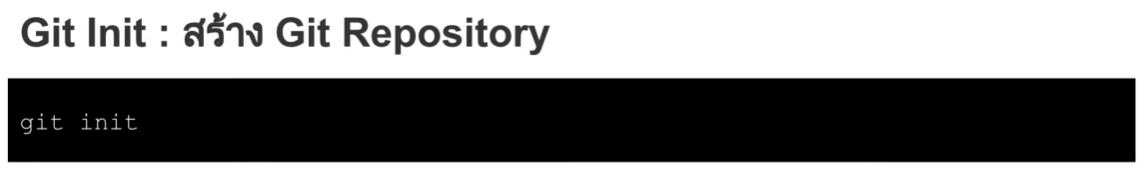


เลือก Terminal แล้ว new terminal ทดสอบพิมพ์คำสั่ง git config –global –list ซึ่งเป็นคำสั่งสำหรับแสดงคุณสมบัติของ git ทั้งหมดใน Global

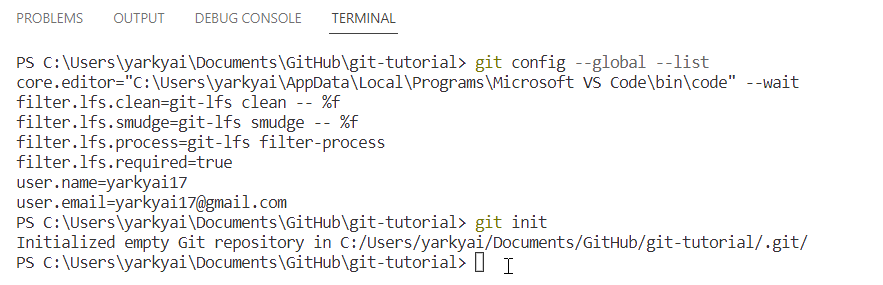


จากรูปจะเห็นว่าโปรแกรมจะแสดง user และ email ที่เราได้ทำไว้เมื่อสัปดาห์ก่อน

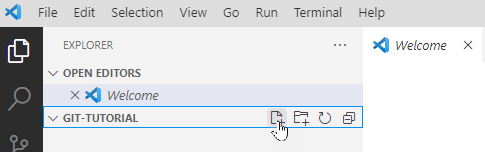
ต่อมาเราจะสร้าง git repository ฝั่ง Local ก่อน ด้วยคำสั่ง git init

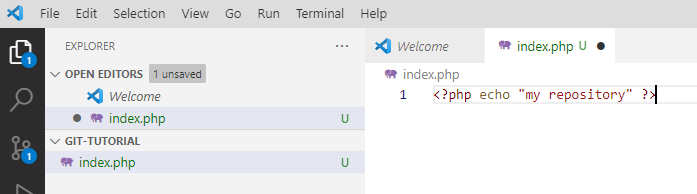


ที่ VS code ของเราให้สังกตุว่า prompt ของเราอยู่ที่ git-tutorial หรือไม่ (ปกติเราเลือกมาถูกอยู่ละ) ให้พิมพ์คำสั่ง git init เพื่อบอกเครื่องว่าเราจะให้ folder ดังกล่าวเป็น repository ของเรา



ต่อมาเราจะลองสร้าง file สั่ง 1 ไฟล์เพื่อทดสอบคำสั่งต่าง ๆ ใน git โดยให้กดปุ่มตามรูป และสร้างไฟล์ชื่อว่า index.php และเขียนโปรแกรมตามรูป



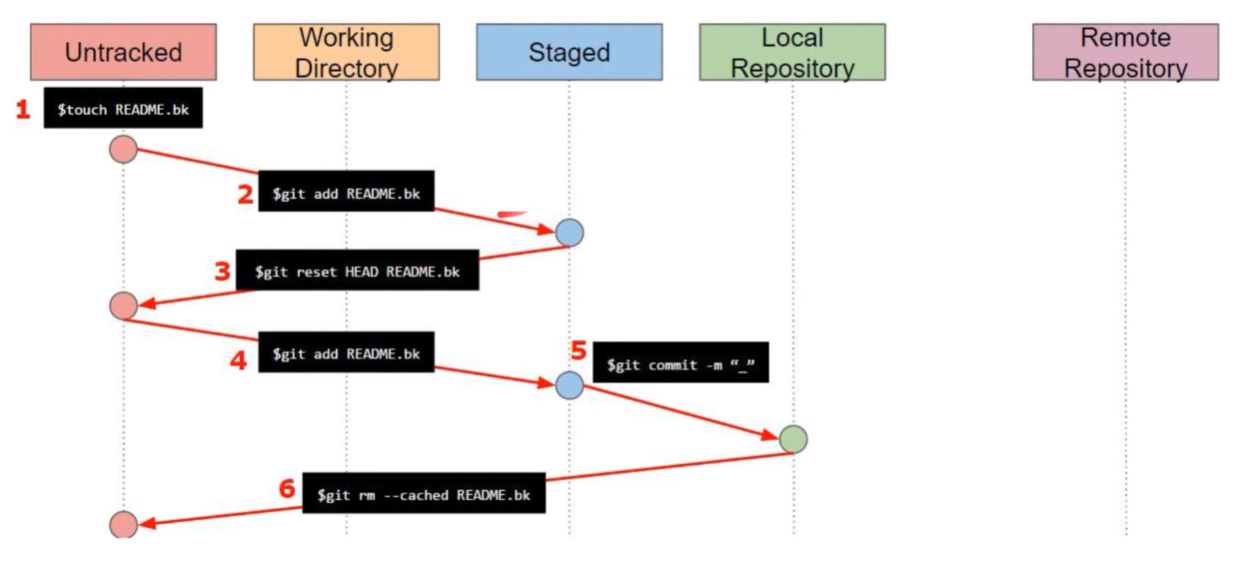


ทำการ save file จะเห็นว่าสถานะของ file จะเป็น U

**สถานะของไฟล์เมื่อเราทำงานบน Git**

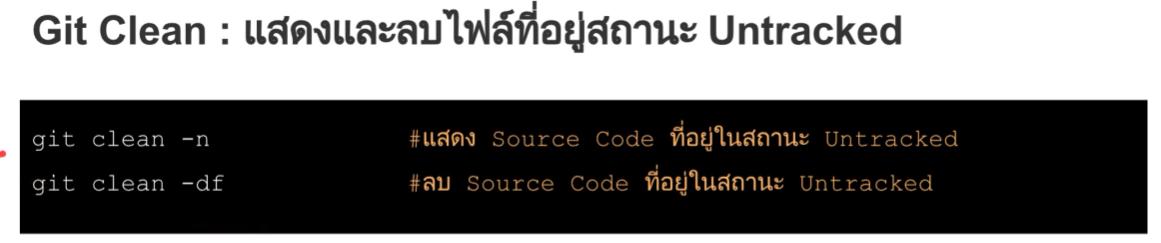


**การทำงานของสถานะต่าง**

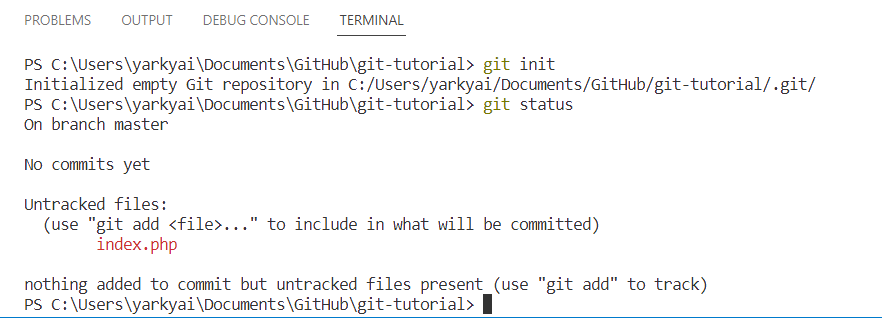


**คำสั่งพื้นฐานของ Git**

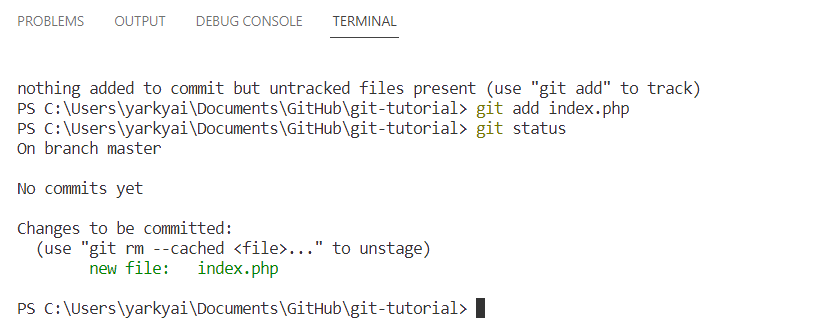




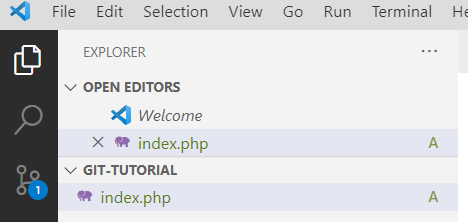
มาทดลองคำสั่งต่างๆ กัน ตอนนี้ใน VS code เราจะเห็นว่าไฟล์ของเราอยู่ในสถานะ U หรือเราสามารถพิมพ์คำสั่ง git status ใน terminal โปรแกรมก็จะแสดงข้อมูลว่ามีไฟล์ index.php อยู่ในสถานะ untracked



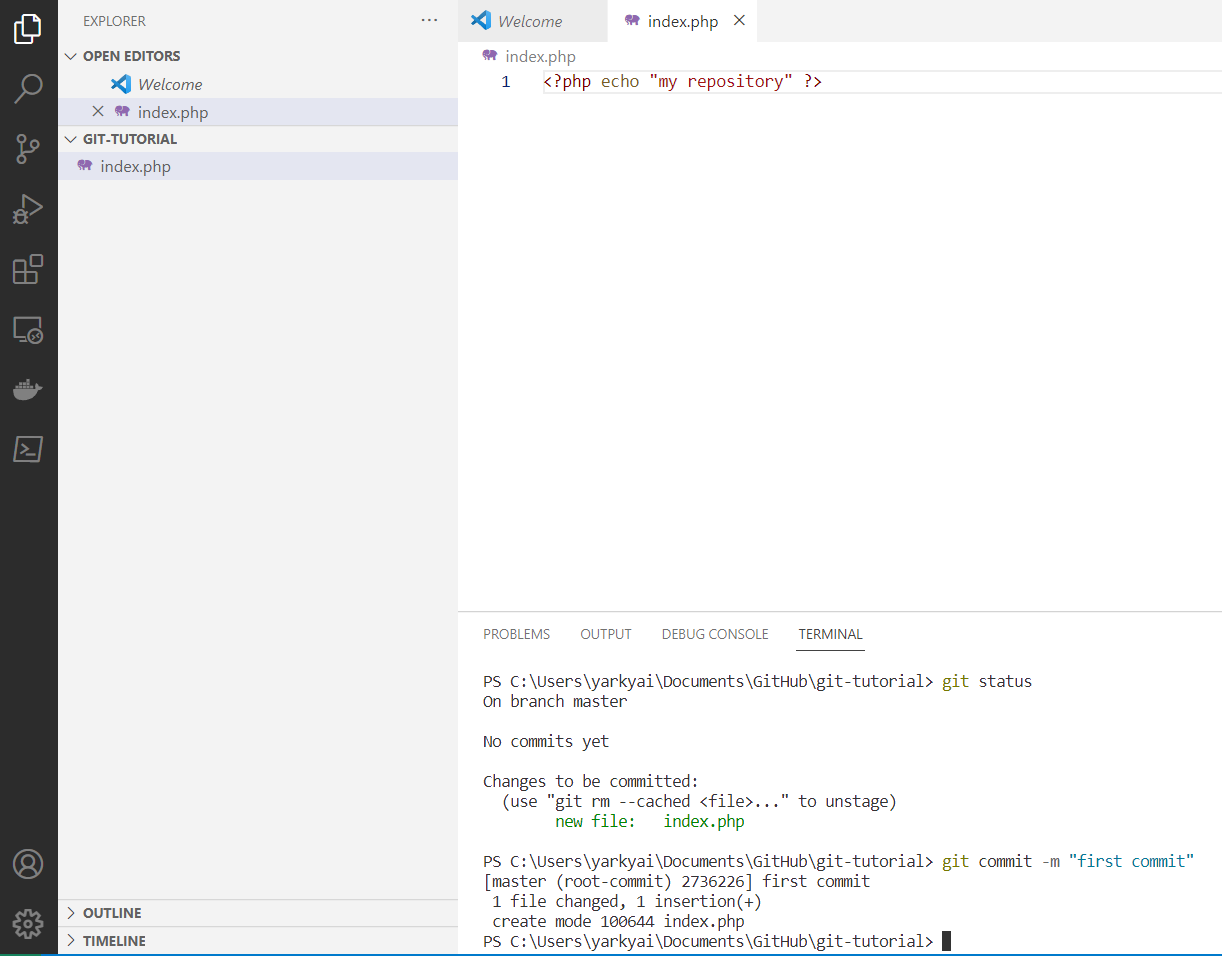
ต่อมาเราจะทำการ add file เข้าไปใน git ของเราด้วยคำสั่ง git add filename



จากรูปจะเห็นว่าเราใช้คำสั่ง git add index.php เพื่อ add ไฟล์ของเราเพื่อเตรียมเข้าสู่ repository จากนั้นลองพิมพ์ git status เพื่อดูสถานะของไฟล์ หรือเราสังเกตที่ VS code ไฟล์ของเราจะมีสถานะเป็น A



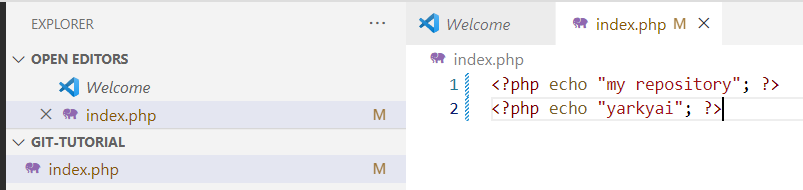
จากขั้นตอนนี้ไฟล์ของเราจะยังไม่ถูก save ไปที่ repository นะครับเราต้องพิมพ์คำสั่งอีก 1 คำสั่งคือการ commit นั่นเอง โดยให้เราใช้คำสั่ง git commit -m “frist commit”



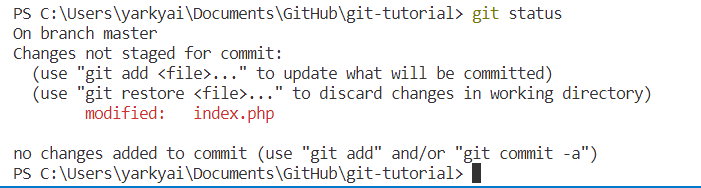
จากรูปจะเห็นว่าสถานะของไฟล์นั่นหายไปแล้ว หมายความว่าตัวไฟล์ได้ถูก commit เข้า repository เรียบร้อยแล้ว ทีนี้ถ้าเราอยากรู้กระบวนการทำงานการ commit ย้อนหลังเราก็สามารถใช้คำสั่ง git log ได้ ซึ่งโปรแกรมจะแสดงข้อมูลดังนี้



ทดลองเพิ่มคำสั่งเข้าไปใน index.php ของเราจากนั้นให้ save file แล้วสังเกตผล



พิมพ์ git status โปรแกรมจะบอกว่ามีการแก้ไขเปลี่ยนแปลงไฟล์

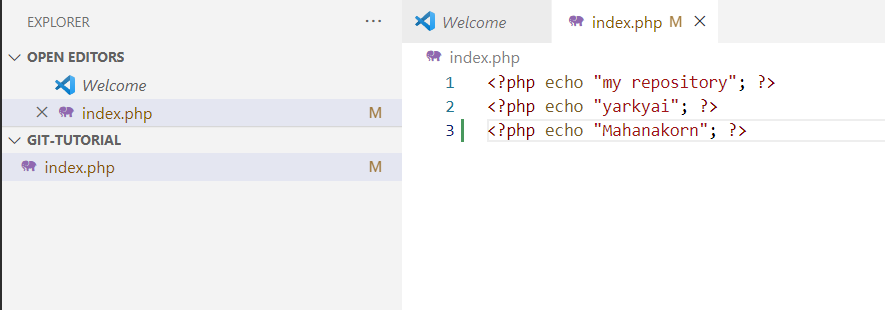


ให้ นศ. ทำการ add และ commit อีกครั้ง จากนั้นพิมพ์คำสั่ง git log (สังเกตเลข commit ดีๆ นะเดี๋ยวเราจะใช้เลขนี้กัน)



ปัญหาหลักๆ ของ Programming คือ เมื่อเราพิมพ์ code ไปเรื่อยๆ เราจะพบว่า เอ้ย เรากลับไปใช้ version เก่าดีกว่า จริงๆ ก็สามารถทำได้ด้วยการ Ctrl Z แต่ถ้าเราแก้ไขไปเยอะเป็นร้อยบรรทัดละ จะต้องกดกี่ครั้ง ที่นี้เราจะเห็นความสำคัญของพวก Version control ละ

ทดลองแก้ไขไฟล์อีกครั้ง และทำการ save อีกครั้ง จะเห็นว่าไฟล์จะขึ้นสถานะ M แล้วเราเอาไฟล์ไปลองแล้วพบว่าเราย้อนกลับไป version แรกดีกว่า

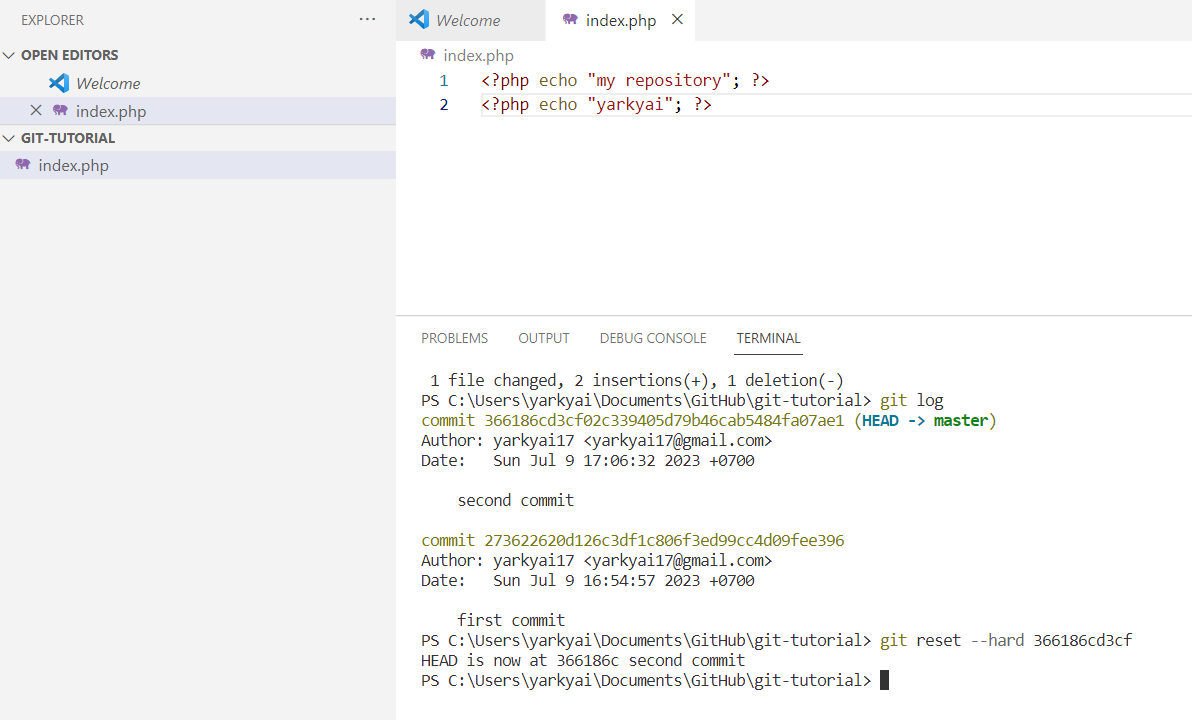


จากเหตุการณ์ดังกล่าวเราสามารถใช้คำสั่งได้ดังนี้

* พิมพ์คำสั่ง git log จะเห็นเลข commit ทั้งสองครั้งของเรา



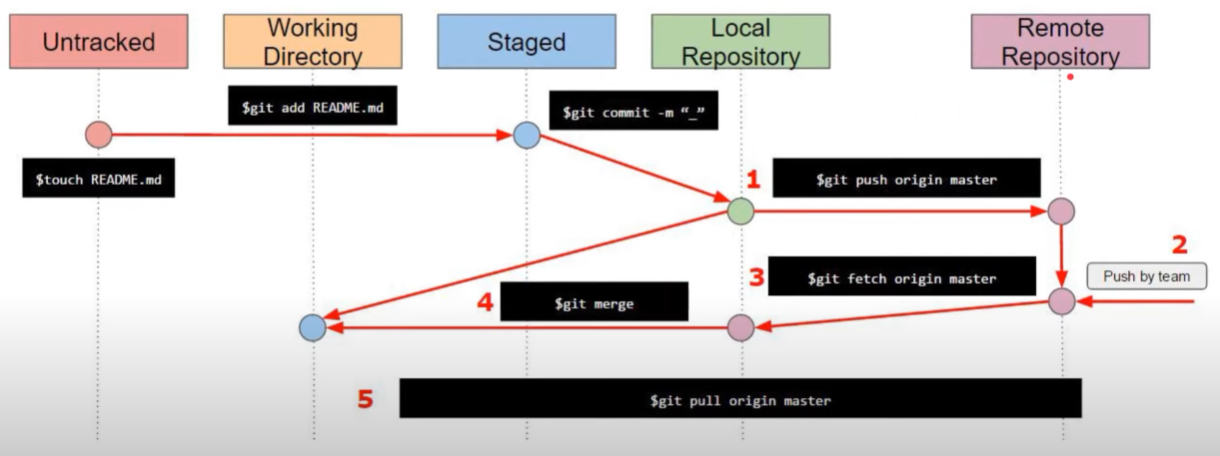
* ทำการ copy ตัวเลขมาซัก 6-7 ตัวอักษร แล้วใช้คำสั่ง git reset –hard แล้วตามด้วยเลขที่เรา copy มาเมื่อกี้



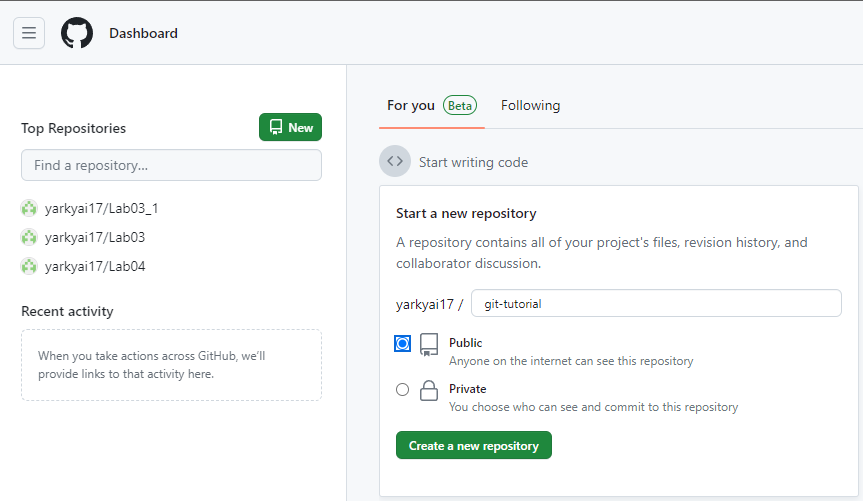
เมื่อใช้คำสั่ง git reset โปรแกรมก็จะย้อนกลับไปเวอร์ชั่นที่เราต้องการได้

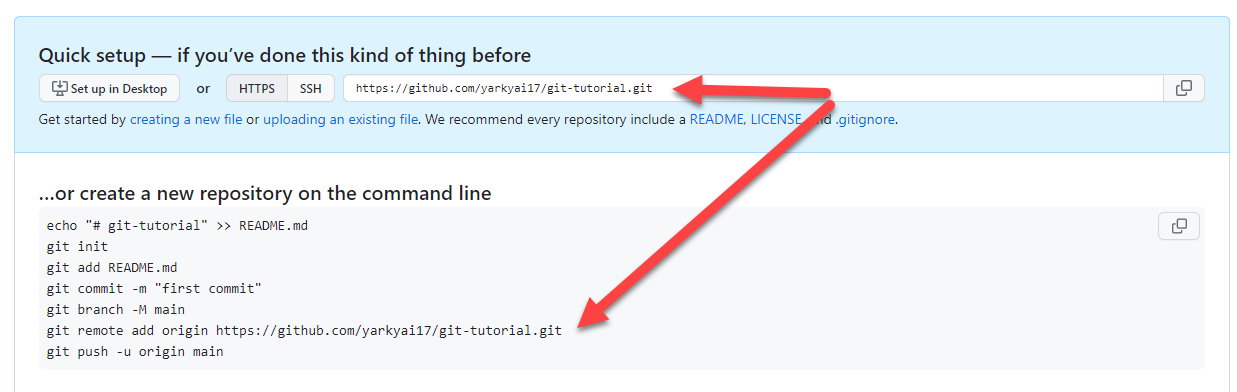
**การทำงานร่วมกับ Github**

จากการทดลองข้างต้นเราจะเป็นการทำอยู่ในฝั่ง local repository เท่านั้น ซึ่งถ้าเราต้องการจะ push ไปที่ฝั่ง remote repository (github) ก็สามารถทำได้ด้วยคำสั่งดังต่อไปนี้



ไปที่เวป github สร้าง repository ชื่อว่า git-toturial (พยายามตั้งให้เหมือนกับเครื่องของเราที่ทำไว้ข้างต้น)





สังเกตสองจุดนี้ไว้ดีๆ เดี๋ยวเราจะต้องใช้

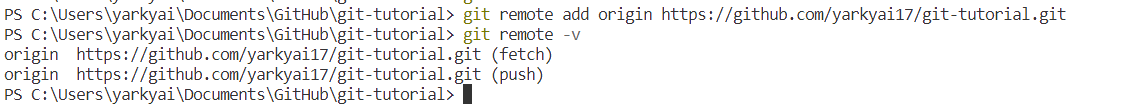
กลับมาที่ VS code พิมพ์คำว่า git remote -v เพื่อดูว่าเราได้เชื่อต่อกับใครไว้หรือไม่ ในที่นี้จะไม่ขึ้นอะไรเพราะเรายังไม่ได้เชื่อมกับ remote ใดๆ เลย



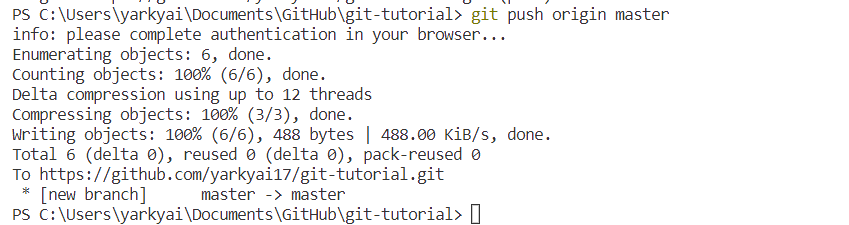
ต่อมาเราจะเชื่อต่อกับ remote ละ ให้เราไป copy ข้อมูลที่หน้าเวป ที่เป็นคำสั่ง

git remote add origin <https://github.com/yarkyai17/git-tutorial.git>

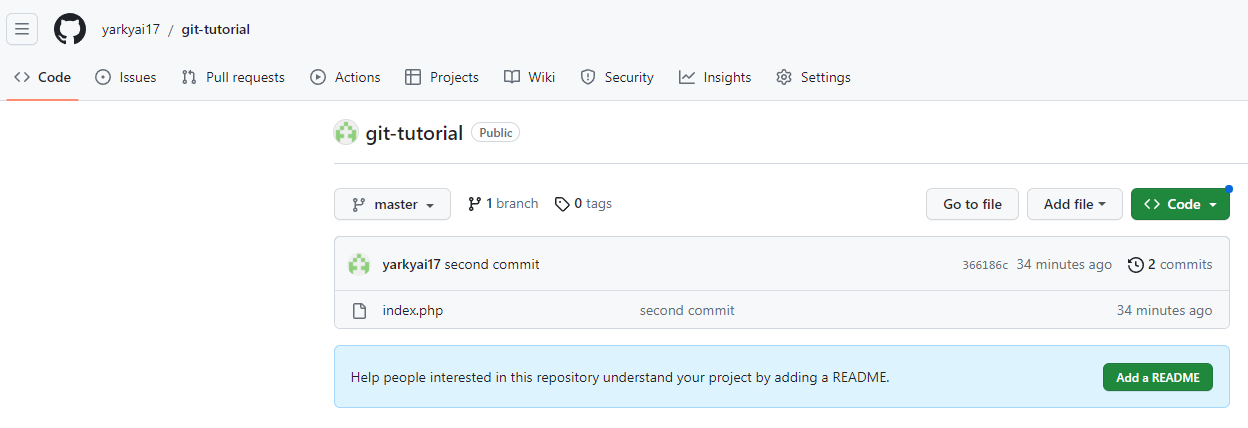
จากนั้นให้เอาไปแปะใน VS code ของเรา แล้วใช้คำสั่ง git remote -v อีกครั้ง จะเห็นว่าเรื่องเราได้ remote ไปที่ปลายทางละ

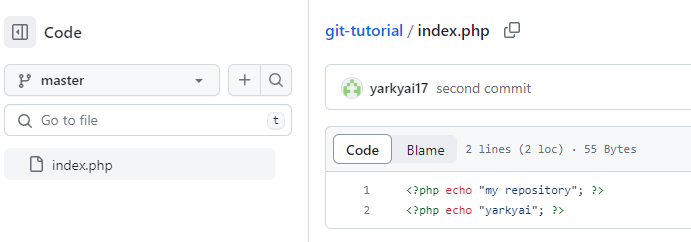


ในตอนนี้เครื่องเราจะต่อกับ remote แล้วแต่ไฟล์ในเครื่องของเราจะยังไม่ไปนะครับ ต้องใช้คำสั่ง git push origin master เพื่อทำการ upload file เราไปที่ github



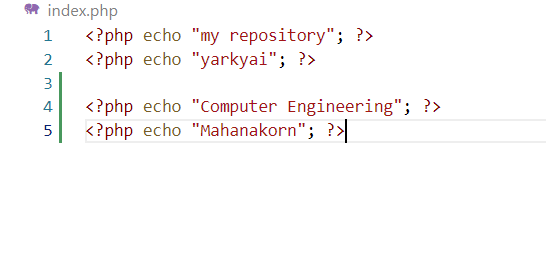
หลังจากพิมพ์คำสั่ง โปรแกรมจะทำงานให้เรารอซักพัก (มีกรณีที่โปรแกรมจะถาม user pass web github ให้เรายืนยันการ login อีกครั้งก็ให้ใส่ password เวป github ไป) กลับไปที่หน้าเวป กด refresh จะเห็นว่าไฟล์ index.php ของเรามาละ



กดเข้าไปดู code ได้ดังนี้

**คำสั่ง push**

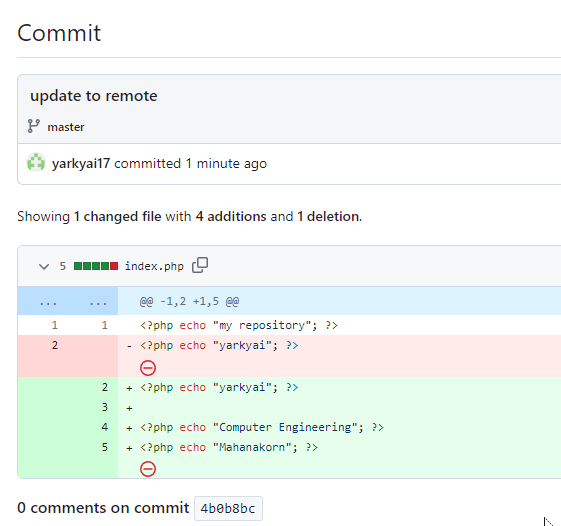
ต่อมาเมื่อเราทำการแก้ไขไฟล์ใน VS code แล้วต้องการ push ไปที่ remote ก็สามารถใช้คำสั่งที่ได้เรียนไปเบื้องต้นแล้วดังนี้



ใช้คำสั่งทั้งหมดอีกครั้งตามรูป

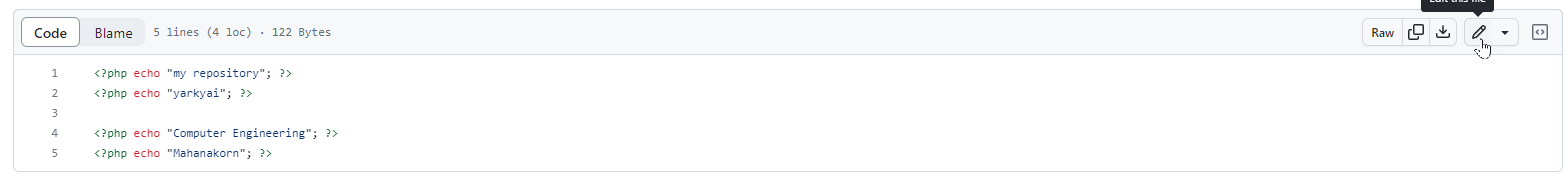


จากนั้นกลับไปดูไฟล์ที่ตัว github สังเกตว่าจะมีการเปลี่ยนแปลงตามที่เราได้ทำไว้

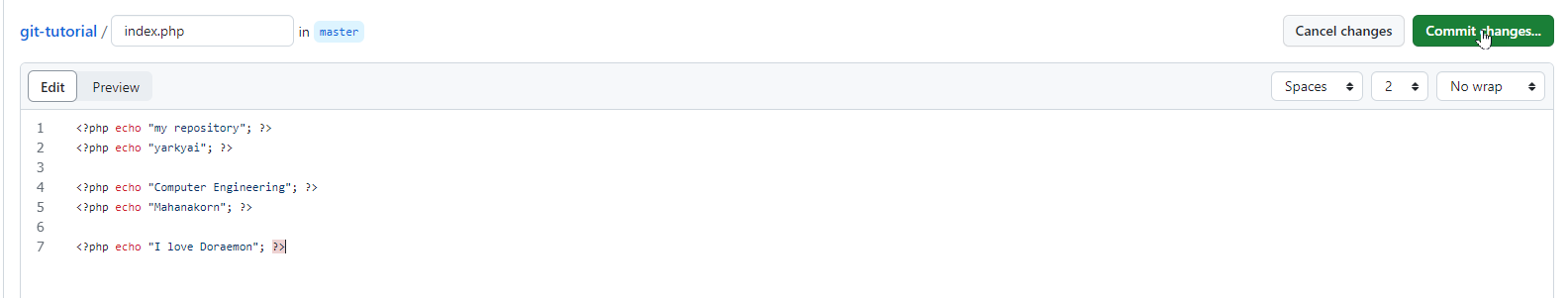


**คำสั่ง git pull**

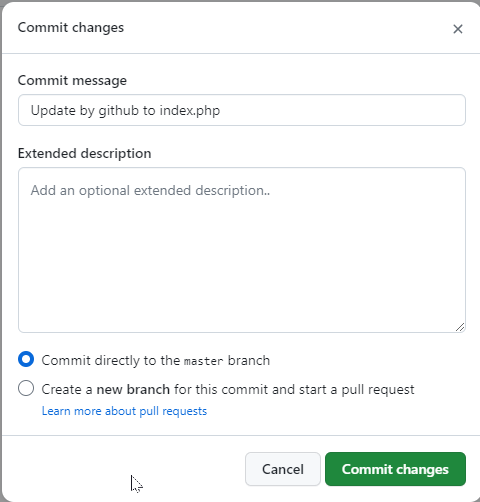
ถ้าไฟล์ของเรามีการทำงานเป็นทีม และเพื่อนร่วมทีมของเรามีการแก้ไขไฟล์ที่เราได้ push ไป ถ้าเราจะเขียนโปรแกรมต่อ ก็ต้องมีการ pull ไฟล์ใหม่มาเพื่อเขียนโปรแกรมต่อ ในทีนี้ผมจะทำแบบง่ายๆ ก่อนนะครับ ให้เราทดลองแก้ไขไฟล์จาก github ได้เลยโดยกดปุ่มแก้ไข



เพิ่มโค้ท บรรทัดที่ 7 เข้าไป แล้วกดปุ่ม Commit changes..

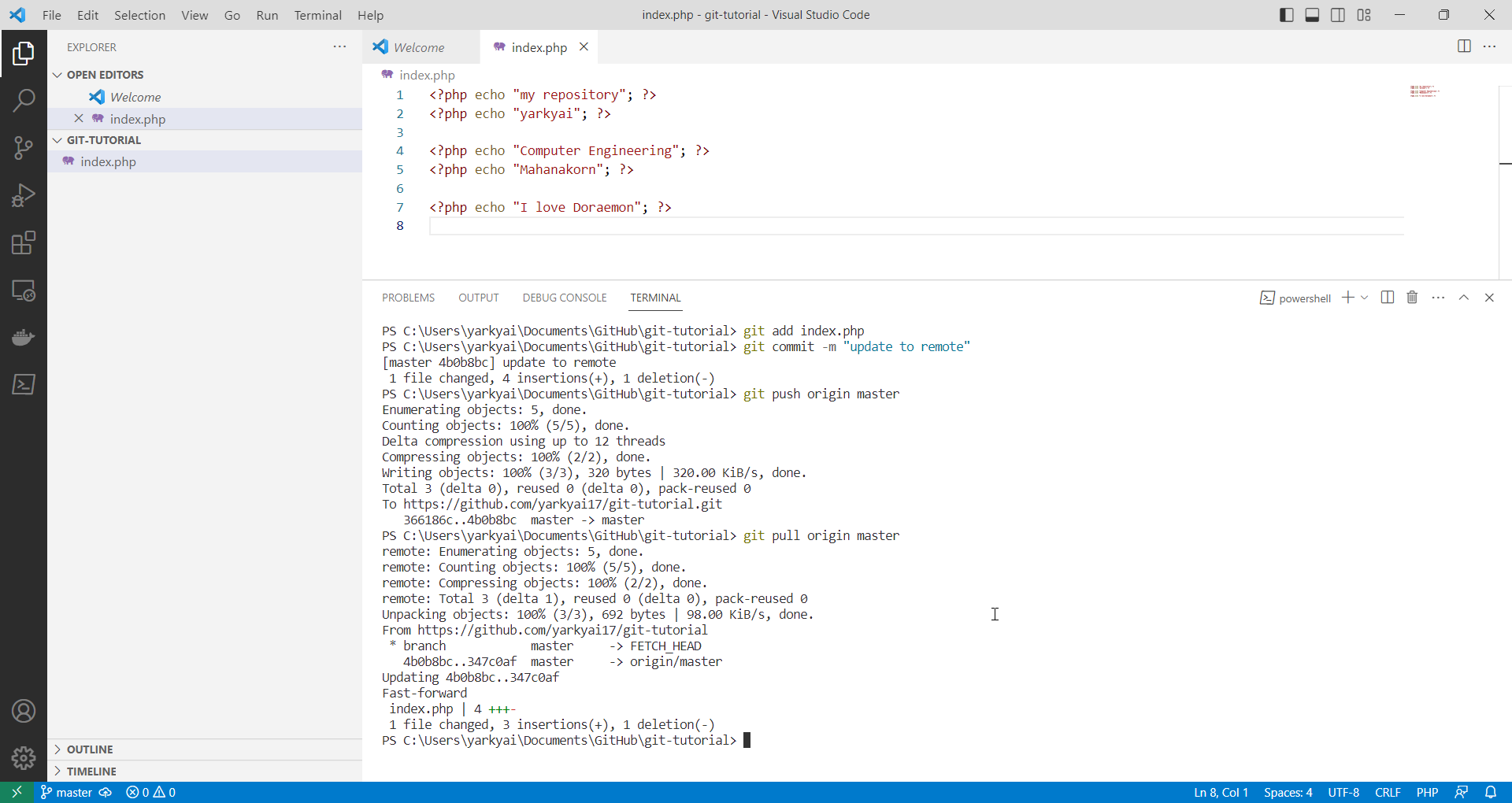


ใน commit message ให้เราพิมพ์ Update by github to index.php แล้วกด Commit changes อีกครั้ง



ในตอนนี้ Code ของเรากับใน remote จะไม่เหมือนกันละ (เราจะไม่สามารถ push ได้ เดี๋ยวจะสอนในอาทิตย์หน้า) หากเราจะทำการแก้ไขไฟล์เราจะต้อง pull Code จาก github มาก่อนด้วยคำสั่งดังนี้

git pull origin master จะเห็นว่า code ของเราจะเปลียนตาม git hub ละครับ



เหนื่อยละอาทิตย์นี้พอก่อนเดี๋ยวอาทิตย์หน้าเราจะมาเรียนคำสั่ง

* Git Branch
* Merge
* Gitignore