

What you learn

1. Git stash
2. Switch branch

Imagine real case. You work as a team for a project. (Assume that you host this project)

(Looloo retail_forecast case)

Day1. You have setup code in main branch. Let's assume that there are only 1 files in main branch

D1 : `src/base_preprocess.py` at main branch

Day2. You need to add lookback feature and training code. So, you have created a new branch which is "branch7" and adding many files in branch7

D2 : <code>src/add_lookback.py</code> , <code>src/training.py</code>	at branch7 (new files)
<code>src/base_preprocess.py</code>	at branch7 (same as D1 in main branch)
Script / ...	at branch7 (new files)
config / ...	at branch7 (new files)

Day3 you made some change in `src/base_preprocess.py`.

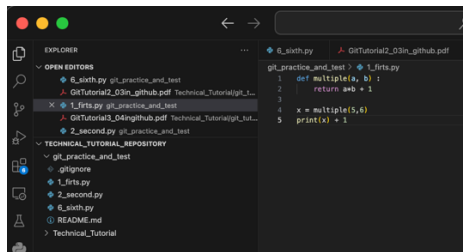
D3 : <code>src/add_lookback.py</code> , <code>src/training.py</code>	at branch7 (new files)
<code>src/base_preprocess.py</code>	at branch7 (change made from main)
Script / ...	at branch7 (new files)
config / ...	at branch7 (new files)

Day4 You want to get back to main branch to recheck your code.

You have 2 choices

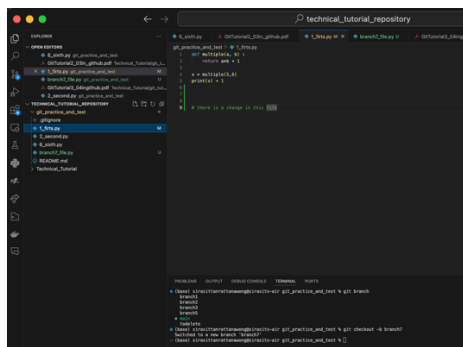
1. Git add, git commit before checkout main (recommended)
2. Git stash (for more detail : <https://www.atlassian.com/git/tutorials/saving-changes/git-stash>)

D1 at main



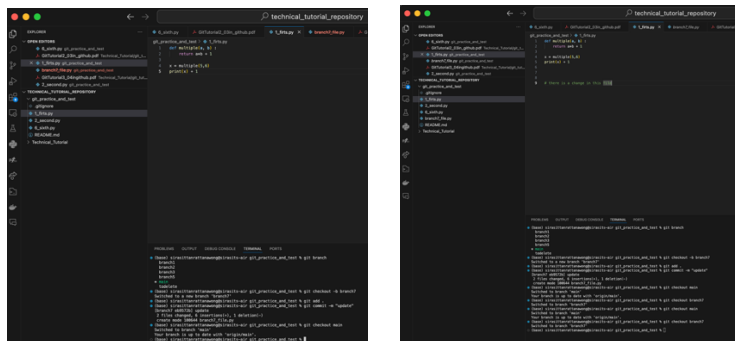
D2,3 at branch7

- Change 1_firts.py
- New file (branch7_file.py)



D4

1. Git add and commit at branch7
 2. Git checkout main
 3. You would see that
- 1_firts.py code จะเหมือนกับตอนแรกที่ branch main เลข # there is a change in this file ใน branch 7 ไม่ตามมาด้วย (ถึงเปิดไฟล์นั้นอยู่พอ checkout ส่วนที่เพิ่มมานั้นก็จะหายไปเลย re-render ให้เหมือน main ตอนแรก)
 - ไม่มี branch7_file.py ตามมาด้วย ** อย่าเผลอไปกด save ละ เวลามันขึ้นสีแดงๆขีด
 - กลับไป checkout branch7 ก็จะเจอของที่แก้ไข



What you learn

1. git reset (revert commit)

Imagine real case. You work as a team for a project. (Assume that you host this project)

(Looloo retail_forecast case)

Day1. You have set up code in main branch. Let's assume that there are only 1 file in main. Then new branch "branch7"

Also git push -u origin branch7 to remote repo

D1 : src/base_preprocess.py at main branch

Day2. You need to add lookback feature and training code

D2 : src/add_lookback.py	src/training.py	at branch7 (new files)
src/base_preprocess.py		at branch7 (same as D1 in main branch)
Script / ...		at branch7 (new files)
config / ...		at branch7 (new files)

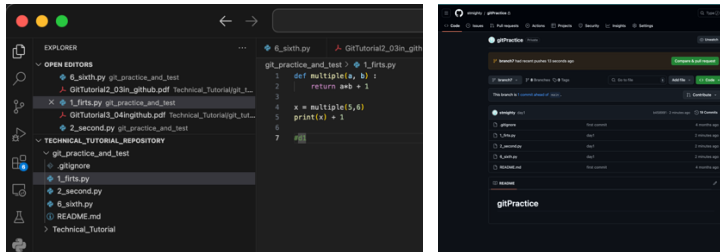
Day3 You accidentally push all code to the branch7 remote repo (In fact, **you should push only src/add_lookback.py**)

D3 : src/add_lookback.py		at branch7 (in remote branch7)
src/base_preprocess.py, src/training.py		at branch7 (in remote branch7)
Script / ...		at branch7 (in remote branch7)
config / ...		at branch7 (in remote branch7)

Day4 You want to revert remote repository of branch7 to the previous version and then push only src/add_lookback.py again

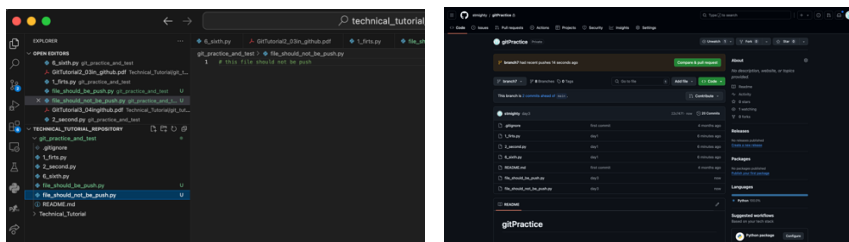
(No change in local, only remote)

D1 at main



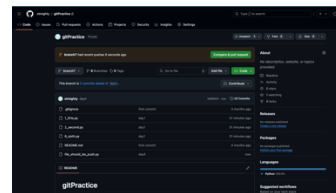
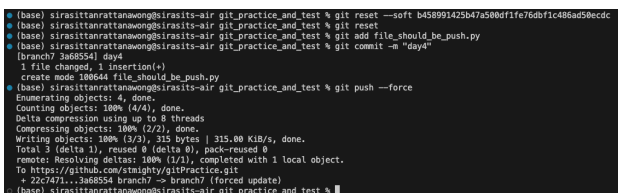
D2,3 at branch7

- push file_should_be_push.py
- push file_should_not_be_push.py



D4 at branch7 (No change in local, only remote)

1. git log and find day1 commit (press q to exit)
2. git reset --soft _____ (commit code)
3. git reset
4. git add file you want to add
5. git commit
6. git push --force (same as git push origin branch7 --force) (it will push to the checkout branch)



Note read git reset --hard if you also want to change local file not only remote