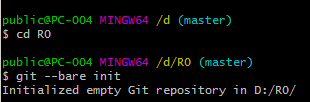
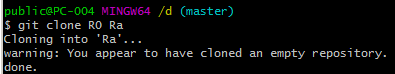
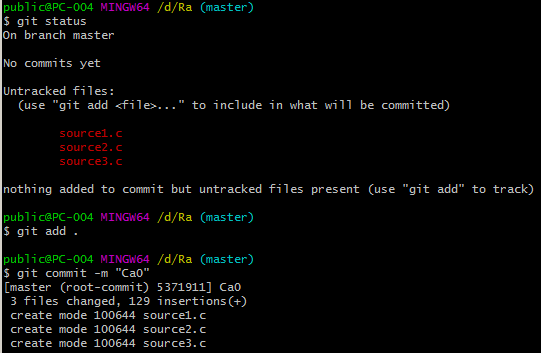
1. Create new local bare repository named **R0**



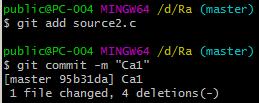
1. Clone new local repository from local bare repository created from #1, new repository named **Ra**



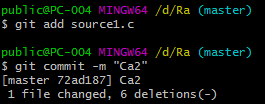
1. In Ra, create/edit source file(s). Add all file and commit these changes (commit **Ca0**) on branch **master**



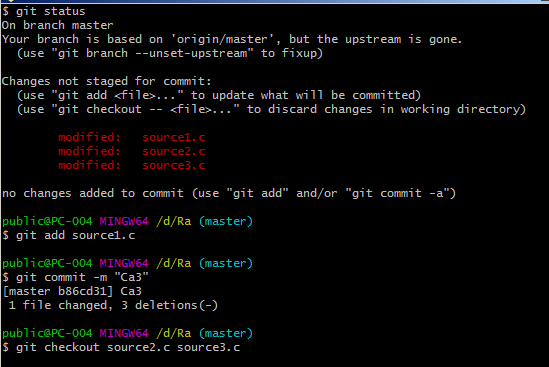
1. Create/edit multi source files. Add some of changed files (not all files), then commit added files (commit **Ca1**).



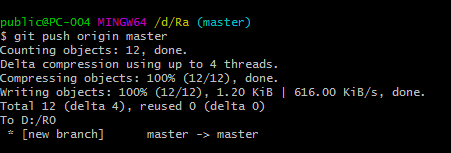
1. Commit other changed files (which did not add in #4) as a commit (**Ca2**).



1. Modify more than 2 source files, then commit 1 file and reset other files (unchanged) as commit **Ca3**.

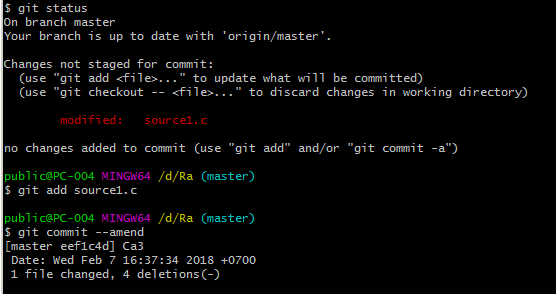


1. Push branch master to remote repository (**R0**).



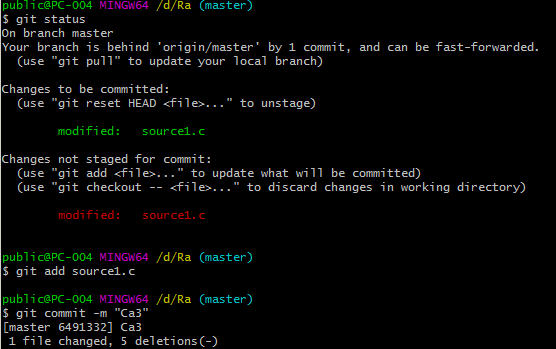
1. Edit one file (one of committed file in #6), then add this change to commit **Ca3** (2 ways, commit --amend and soft reset then commit, answer how different between these ways?).

Answer: git commit --amend change the most recent commit on current branch and git reset --soft remove commit from current branch but keep changes.

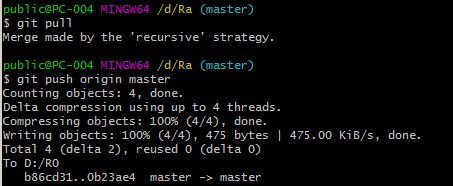


Method 2:

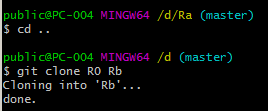




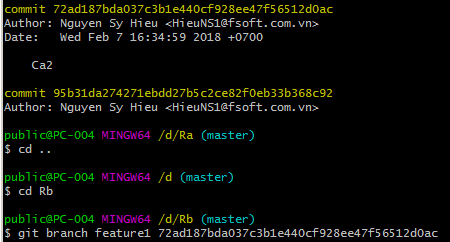
1. Push master branch to remote repository (**R0**) again.



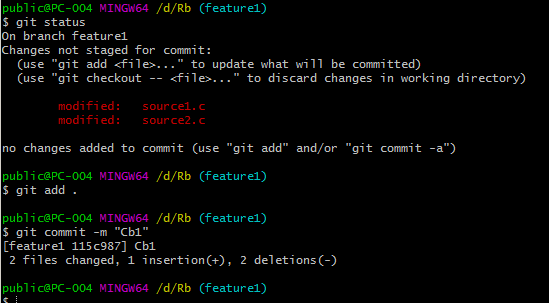
1. Clone another local repository from local bare repository created from #1, new repository named **Rb**.



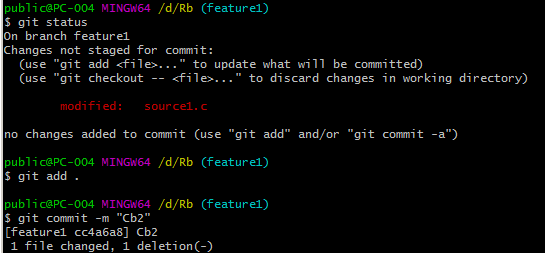
1. On **Rb**, create new branch named **feature1** from commit **Ca2** of **master** branch.



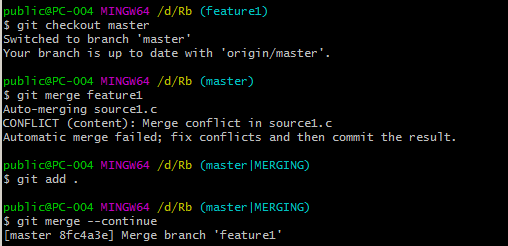
1. Edit 2 files then commit to **feature1** branch (commit **Cb1**).



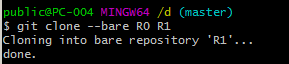
1. Edit 1 file which is changed in commit **Ca3** (#6 or #8), edit the same location (in file) with change from **Ca3**. After that, commit this commit as **Cb2**.



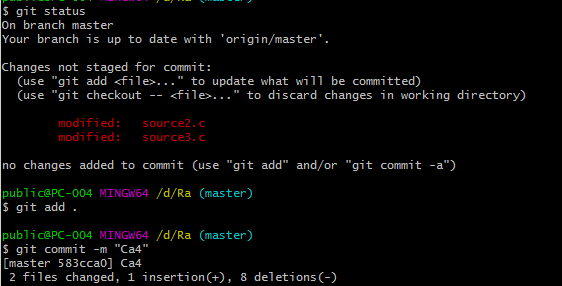
1. Merge branch **feature1** (**Cb2**) with **master** branch (**Ca3**). Resolve conflicts then push back to **master** branch of remote repository **R0**.



1. Create another remote repository named**R1** by copy**R0**.



1. On **master** branch of **Ra**, edit some files then commit (**Ca4**).



1. Add new remote named **rmt-r1** has URL points to **R1**.
2. Push new changes of **master** branch (of **Ra**) to **R1**.
3. On **Ra**, create new branch named **test\_rebase** based on commit **Ca2**.
4. On **test\_rebase** branch, edit one file which is changed in commit **Ca3** or **Ca4** (#8 or #16), edit the same location (in file) with change from **Ca3** or **Ca16**. Then commit these changes to **Cat1**.
5. Rebase the **Cat1** to the latest of **Ra/master** with conflicts resolved.