# Git commands

git checkout -b <branch-name>

git status

git add .

git commit -m "<Description of change>"

git reset --soft HEAD~

git push origin <branch-name>

git checkout develop

git pull --rebase

git merge <branch-name>

git push origin develop

git branch -d <branch-name>

git branch

git checkout --track origin/<branch-name>

git push origin –delete <branch-name>

git push –set-upstream origin pen1

git merge master -s recursive -X renormalize

1. Git status
2. Git commit –a // commit user changes first to local or

Giit add .

Preferably do commit though GUI

1. Git fetch origin  develop
2. Git status  // may have some conflicts
3. Git fetch origin / develop
4. Git status
5. Git rebase origin / develop
6. Note : once rebase is done go to GUI to see conflict files .

Files which are unchecked have  conflicts. Carefully  check them in vs or notepad and merge the code( have any doubts check with others.

Please don’t do blindly) in those files only and save. Do not rebuild or fix any build issues at the issue as we are in the middle of rebase.

               If u do any other fixes u need to cancel rebase using  Git rebase –abort then need to rebase again

1. Git status
2. Git add f.
3. Git rebase – continue
4. \*\*\*\* once rebase done do basic testing listing all tabs pass/ fail summary etc/
5. Git push