GIT

A repository is your whole project (directories and files) that you clone on your computer. A **branch is a version of your repository**, or in other words, an independent line of development. A repository can contain multiple branches, which means there are multiple versions of the repository

**Local repository**

1.initializare git in folder

2.creare fisier

3.adaugare in fisier

4.commit changes

5.afisare log

6. modificare mesaj commit

7.remove last comit/ remove a specific commit

8.creaza un alt branch

9.switch from master and first branch

10.comit from first branch

11.remove a local branch

12.merge to master changes from other branches

**Remote repository**

13.start from local repo to remote repository

14.start from remote repo to local repository

15.create branches for remote repository

16. remove a remote branch

17.merge chages from branges to master on remote repository

18. remove folder or file from remote branch

**Sincronize data**

19.sent data from local repo to remote repository

20.pull date from remote repo to local repository

21.sincronize data from remote to local

22 Change the remote repository link for a local repository

23 Push changes directly to multimple remote repository

24.Branch vs Repository and merge branch

25.Revert to an OLD git version

1. Initializare git in folder

in folderul creat execute command : git init

CTRL+H ==> .git – folder devine vizibil

2. Create file

touch filename

3. Adaugare in fisier

gedit filename

4. Commit changees

git status

git add file name

git commit -m ‘comment’

git status

5. Afisare log

git log

git log ==on-line

6. Modificare mesaj commit

git commit --amend

7. Remove last commit / Remove a specific commit

(cand apar modificari...dar de fapt sunt in memorie. Apar in eclipse si in git status→ redy to be added. Dar nu e neveoie de ele)

**git reset --hard HEAD~1** //// git log –oneline ==> has pentru fiecare commit

git revert hash corespunzator pentru commitul care ma intereseaza

**reverting the last commit**

git reset HEAD~

git reset --soft HEAD~

8. Create a new branches

git checkout -b new\_branck

9. Switch from master and first branch

git branch

git checkout branch name

10. Commit from first branch

same command

11. Remove a local branche

git branch -D Branch\_Name

12. Merge to master changes from other branches

Switch to =MASTER==> git checkout master

from MASTER execute command :

git merge numele\_branchului **EX** tot ce este in branch **X** va fi trimis pe branchul local **MASTER**

if error :

Auto-merging file.txt

CONFLICT (content): Merge conflict in file.txt

Automatic merge failed; fix conflicts and then commit the result.

=== FIX with ATOM editor

OBS : After we performed commit on local repository and sent to the remote repository in case case that we have “merge conflict” we need to UNDO THE LAST COMMIT

git reset HEAD~

After this command we need to pull data from the remote repository to local , made again the changes and sent them again.

git pull origin master

somethime if it is needed

**git stash** si apoi **git pull --rebase → dar se pierde ca este pe local si se suprascrie cu este pe server**

13. Start from local repo to remote repository

On remote repo create the folder name the same with the local one

copy the coresponding link from remote repo

ex: <https://github.com/irimiacip/git_local_reposistory.git>

git remote add origin <https://github.com/irimiacip/git_local_reposistory.git>

git add README.md

git commit -m "add README"

git push origin master

in cazul in care LUCREZI DE UNUL SINGUR pe MASTER BRANCH se poate folosy :

**git push -u origin master --force**

14. Start from remote repo to local repository

copy the coresponding link from remote repo

ex: https://github.com/irimiacip/remotetolocal.git

git clone https://github.com/irimiacip/remotetolocal.git

15.Create branches for remote repository

create branch locally

git checkout -b <branch-name>

git push <remote-name> <local-branch-name>:<remote-branch-name>

remote-name = origin

pentru up-load on branch (master or other branch)

git push --set-upstream <remote-name> <local-branch-name>

16. Remove a remote branch

git push <remote\_name> --delete <branch\_name>

git push <remote\_name> :<branch\_name>

17.Merge changes from branches to master on remote repository

Pull data from remote branch A to local branch A

Merge data from local branch A to local Master

Push data from local Master to remote Master

18. remove file / folder from remote git repository

git rm --cached <filename>

git rm --cached -r <dir\_name>

git commit -m "Removed folder from repository"

git push origin master

19.sent data from local repo to remote repository

commit on local your changes

git push origin master

20.Pull date from remote repo to local repository

git pull origin master(branch name)

git fetch origin

21.Synchronize data from remote to local

git checkout master

git fetch -p origin

git merge origin/master

git checkout branch\_name

git merge master

git push origin <feature-branch>

22 Change the remote repository link for a local repository

local repository A sent changes to remote reposytory B

local repository A should sent changes to remote reposytory C

step 1: git remote set-url origin <https://github.com/irimiacip/selenium_tests.git>

step 2 : git push -u origin master

23 Push changes directly to multimple remote repository

Trebuie sa mergem in fisierul .config si acolo avem initial :

[core]

symlinks = false

repositoryformatversion = 0

filemode = false

logallrefupdates = true

[remote "origin"]

url = https://github.com/irimiacip/ZZZ.git

fetch = +refs/heads/\*:refs/remotes/origin/\*

[branch "master"]

remote = origin

merge = refs/heads/master

In acest fisier adaugam a doua calea catre remote repo :

[core]

symlinks = false

repositoryformatversion = 0

filemode = false

logallrefupdates = true

[remote "origin"]

url = https://github.com/irimiacip/ZZZ.git

**url = https://git.metrosystems.net/ciprian.irimia/zzz.git**

fetch = +refs/heads/\*:refs/remotes/origin/\*

[branch "master"]

remote = origin

merge = refs/heads/master

Cand vom face push din consola sau din eclipse ne vom autentifica separat in cele doua repository

24. Branch vs Repositoty and merge branch

Reposotory := your whole project (directories and files) that you clone on your computer.

Branch :=version of your repository, or in other words, an independent line of development

A repository can contain multiple branches, which means there are multiple versions of the repository

Merging branch : Go to destination and select : source branch 🡪 merge in existing.

Caz in care se face merge 🡺 perfect

Sau apar conflicte 🡪 se selecteaza conflictul si apoi se accepta ce trebuie schimbat

25. Revert to an old git version :

a) create another local git branch

b) run : git log

c) run : git revert --no-commit 68bfa99a0e1218fd1d60f88123b0bbac9ee8ebea..HEAD

unde “string” = ID corespunzator unui anume commit !!!!