

### REMOTE INFORMATION

#### Clone a repository

```
git clone <repository_url>
```

#### Get remote refs and objects

```
git fetch
```

#### Get remote branch (fetch previously)

```
git checkout <remote_branch_name>
```

#### Get remote changes from branch

```
git pull
```

#### Get remote changes by putting ours on top

```
git pull --rebase
```

#### Same and stashing local changes

```
git pull -r --autostash
```

#### Send changes to remote branch

```
git push origin <local_branch>
```

#### Send changes to local branch to another remote branch

```
git push origin <local_branch>:<remote_branch>
```

#### Force push changes

```
git push -f origin <local_branch>
```

### BRANCHING

#### Create branch

```
git branch <branch_name>
```

#### Remove local branch

```
git branch -d <branch_name>
```

#### Track remote branch

```
git branch --set-upstream-to origin/<remote_branch_name> <remote_branch_name>
```

#### Move from branch

```
git checkout <branch_name>
```

#### Create branch and move to her

```
git checkout -b <branch_name>
```

#### Remove remote branch

```
git push origin :<remote_name_branch>
```

### SAVE CHANGES

#### Add file changes to the stage for commit

```
git add <filename>
```

#### Add all changes to the stage

```
git add -A
```

#### Create commit with message

```
git commit -m "<message>"
```

#### Add files to existing commit holding message

```
git commit --amend --no-edit
```

#### No committed changes quick saves in stack

```
git stash -u
```

#### Show all stashes

```
git stash list
```

#### Apply changes from stash

```
git stash apply
```

#### Apply last stash and take out from stack

```
git stash pop
```

### UNDO CHANGES

#### Undo file changes out of stage

```
git checkout <filename>
```

#### Take out file of stage

```
git reset <filename>
```

#### Move from branch and undo all no committed changes

```
git checkout -f <branch>
```

#### Undo N last commits

```
git reset --hard HEAD~N
```

#### Undo N last commits holding changes in stage

```
git reset HEAD~N
```

#### Undo N last commits holding changes out of stage

```
git reset --soft HEAD~N
```

#### Revert changes N last commits (remote)

```
git revert HEAD~N
```

### STATUS

#### Show current branch

```
git branch
```

#### Show local and remote tracked branches

```
git branch -a
```

#### List branch information in verbose mode

```
git branch -vv
```

#### Show branch status

```
git status
```

#### Show file changes

```
git diff <filename>
```

#### Show commits history of current branch

```
git log
```

#### Pretty log example

```
git log --graph --abbrev-commit --decorate -
-format=format:'%C(bold blue)%h%C(reset)
-C(bold cyan)%aD%C(reset) %C(bold
green)(%ar)%C(reset)%C(bold yellow)%d-
%C(reset)%n" %C(white)%s%C(reset)
%C(dim white)- %an%C(reset)' --all
```

### MERGING

#### Add commit from another branch

```
git cherry-pick <commit_hash>
```

#### Merge branches recursively creating a new commit

```
git merge <branch_name>
```

#### Cancel merge

```
git merge --abort
```

#### Merge putting ours commit on base branch

```
git rebase <base_branch_name>
```

#### Cancel rebase

```
git rebase --abort
```

#### Squashes changes

```
git rebase -i
```

