

# Git

`git reset`

# Commit v1

- `git commit -am "v1"`

## HISTORY

f1.txt
v1

## STAGED

f1.txt
v1

## CARPETA LOCAL

f1.txt
v1

# Commit v2

- `git commit -am "v2"`

## HISTORY

f1.txt
v1 v2

f2.txt
v2

## STAGED

f1.txt
v1 v2

f2.txt
v2

## CARPETA LOCAL

f1.txt
v1 v2

f2.txt
v2

# Commit v3

- `git commit -am "v3"`

## HISTORY

f1.txt
v1 v2 v3

f2.txt
v2 v3

f3.txt
v3

## STAGED

f1.txt
v1 v2 v3

f2.txt
v2 v3

f3.txt
v3

## CARPETA LOCAL

f1.txt
v1 v2 v3

f2.txt
v2 v3

f3.txt
v3

# Git log

- `git log --pretty="format:%h %s"`

5687d92 v3

a3aeb2a v2

0d588ed v1



`git reset --hard`

# git reset --hard

- git reset --hard 0d588ed

## HISTORY

f1.txt
v1

## STAGED

f1.txt
v1

## CARPETA LOCAL

f1.txt
v1

# git log

- `git log --pretty="format:%h %s"`

0d588ed v1



# git status

- git status

On branch master  
nothing to commit, working tree clean



```
git reset --soft
```

# Git reset --soft

- `git reset --soft 0d588ed`

## HISTORY

f1.txt
v1

## STAGED

f1.txt
v1 v2 v3

f2.txt
v2 v3

f3.txt
v3

## CARPETA LOCAL

f1.txt
v1 v2 v3

f2.txt
v2 v3

f3.txt
v3

# Git log

- `git log --pretty="format:%h %s"`

0d588ed v1

# git status

- git status

On branch master

Changes to be committed:

(use "git reset HEAD <file>..." to unstage)

modified: f1.txt

new file: f2.txt

new file: f3.txt



`git reset --mixed`

# Git reset --mixed

- `git reset --mixed 0d588ed`

## HISTORY

f1.txt
v1

## STAGED

f1.txt
v1 v2 v3

## CARPETA LOCAL

f1.txt
v1 v2 v3

f2.txt
v2 v3

f3.txt
v3

# Git log

- `git log --pretty="format:%h %s"`

0d588ed v1



# git status

- git status

On branch master

Changes not staged for commit:

(use "git add <file>..." to update what will be committed)

(use "git checkout -- <file>..." to discard changes in working directory)

modified: f1.txt

Untracked files:

(use "git add <file>..." to include in what will be committed)

f2.txt

f3.txt

no changes added to commit (use "git add" and/or "git commit -a")