

.git*

git

.gitconfig

- ~/.gitconfig:

```
[alias]
  ck = checkout
  s = status
  l = log --oneline --max-count=10
  ll = log --pretty=format:"%C(yellow)%h%Cred%d\\ %Creset%s%Cblue\\ [%cn]" --
decorate --numstat
  lg = log --oneline --decorate --graph
  ls = log --pretty=format:"%C(yellow)%h%Cred%d\\ %Creset%s%Cblue\\ [%cn]" --
decorate
alias = ! git config --get-regexp ^alias\\. | sed -e s/^alias\\.\\.// -e s/\\
/\\ =\\ /
  a = add
  b = branch
  c = commit
  bv = branch -avv
  lv = log --walk-reflogs --max-count=10
  ai = add --interactive
  ss = status --short
```

```
# https://gist.github.com/johnpolacek/69604a1f6861129ef088
```

```
[alias]
  co = checkout
  cob = checkout -b
  coo = !git fetch && git checkout
  br = branch
  brd = branch -d
  brD = branch -D
  merged = branch --merged
  st = status
  aa = add -A .
  cm = commit -m
  aacm = !git add -A . && git commit -m
  cp = cherry-pick
  amend = commit --amend -m
  dev = !git checkout dev && git pull origin dev
  staging = !git checkout staging && git pull origin staging
  master = !git checkout master && git pull origin
  po = push origin
  pod = push origin dev
  pos = push origin staging
  pom = push origin master
  poh = push origin HEAD
  pogm = !git push origin gh-pages && git checkout master && git pull origin
master && git rebase gh-pages && git push origin master && git checkout gh-pages
  pomg = !git push origin master && git checkout gh-pages && git pull origin gh-
pages && git rebase master && git push origin gh-pages && git checkout master
```

```

plo = pull origin
plod = pull origin dev
plos = pull origin staging
plom = pull origin master
ploh = pull origin HEAD
unstage = reset --soft HEAD^
ls = log --pretty=format:"%C(yellow)%h%Cred%d\\ %Creset%s%Cblue\\ [%cn]" --
decorate
ll = log --pretty=format:"%C(yellow)%h%Cred%d\\ %Creset%s%Cblue\\ [%cn]" --
decorate --numstat
f = "!git ls-files | grep -i"
gr = grep -Ii
la = "!git config -l | grep alias | cut -c 7-"

```

.gitignore

```

outgoing/*
!outgoing/.gitkeep

data/*
!data/.gitkeep

code/*
!code/fingscan_linux
!code/start.sh
!code/IBScanUltimate_x64_3.9.3.tgz

__pycache__/*

credentials.csv

```

comprobar porqué se ve afectado un fichero:

```
git check-ignore -v <ruta/al/archivo>
```

aunque no esté afectada por ninguna regla:

```
git check-ignore -v --no-index --non-matching <ruta/al/fichero>
```

Eliminar archivos que se estaban rastreando y que se han añadido posteriormente a .gitignore:

```

# añadir directorio a .gitignore
git add .gitignore
git rm -r --cached <DIR> # eliminar del indice de git, no localmente
git commit -m "Eliminación indice git directorio <DIR>"

```

[git, sacar archivos del indice git \(.gitignore\)](#)

.gitkeep

Fichero vacío para indicar que ese directorio se ha de mantener, aunque no contenga más archivos.

.gitattributes

indica la naturaleza de cierto tipos de archivos para optimizar el control de cambios.

```
*.tgz binary  
code/*.zip binary
```

From:

<https://miguelangel.torresegea.es/wiki/> - **miguel angel torres egea**

Permanent link:

<https://miguelangel.torresegea.es/wiki/tech:git:dotfiles?rev=1765542877>

Last update: **12/12/2025 04:34**

