

# GitHub

Fernando Anselmo - Versão 1.0

## Inicializações

Clonar um repositório existente:  
`git clone [endereço]`

Criar um novo repositório local:  
`git init`

## Modificações

Verificar as mudanças do diretório de trabalho:  
`git status`

Comparar alterações do diretório de trabalho:  
`git diff`

Adicionar todas mudanças ao próximo *commit*:  
`git add .`

Adicionar um arquivo ao próximo *commit*:  
`git add [arquivo]`

Mensagem do *commit* aos arquivos adicionadas:  
`git commit -m "mensagem"`

Marcar o *commit* atual:  
`git tag [nome]`

Modificar a branch atual:  
`git switch -C [próximo] [anterior]`

Enviar os arquivos adicionados ao repositório remoto:  
`git remote add origin [url]`

## Histórico

Listar todos as configurações do repositório:  
`git remote -v` Mostrar as informações sobre o repositório:

`git remote show [nome]` Verificar todos os *commits* realizados (**q** para sair):  
`git log`

Verificar *commits* de um arquivo (**q** para sair):  
`git log [arquivo]`

Quem/quando mudou um arquivo (**q** para sair):  
`git blame [arquivo]`

## Desfazer Modificações

Descartar as mudanças locais:  
`git reset --hard HEAD`

Retornar a um ponto de *commit* (SEM descartar):  
`git reset [commit]`

Retornar a um ponto de *commit* (manter mudanças atuais):  
`git reset --keep [commit]`

Retornar a um ponto de *commit* (descartar):  
`git reset --hard [commit]`

Descartar as mudanças de um arquivo:  
`git checkout HEAD [arquivo]`

Reverter um *commit*:  
`git revert [commit]`

## Branches e Tags

Listar todas as *branches* existentes:  
`git branch -av`

Modificar o cabeçalho da *branch*:  
`git checkout [branch]`

Criar uma nova *branch* com base na corrente:  
`git branch [novaBranch]`

Criar um nova trilha:  
`git checkout --track [remote/Branch]`

Remover a *branch* local:  
`git branch -d [branch]`

Eliminar a *branch* remota:  
`git branch -dr [remote/Branch]`

## Ações Diárias

Basicamente essas são as ações diárias para deixar tudo em dia:

```
git pull
git add .
git commit -m "mensagem"
git push -u origin master
```

## Publicações

Baixar mudanças (NÃO integra o cabeçalho):  
`git fetch [repositório]`

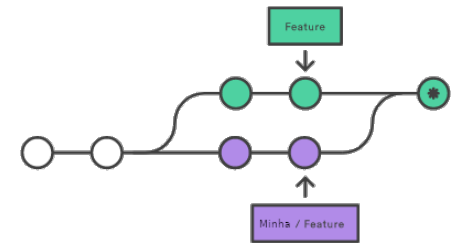
Baixar mudanças (integra o cabeçalho):  
`git pull [repositório] [branch]`

Publicar as mudanças:  
`git push -u origin [branch]`

Publicar as *tags*:  
`git push --tags`

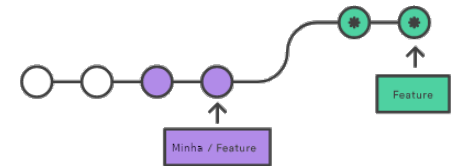
## Integrar Alterações da Ramificação

### Mesclagem



Realizar uma mesclagem:  
`git merge [branch]`

### Rebase



Realizar um rebase:  
`git rebase [branch]`

Parar uma rebase:  
`git abort [branch]`

Resolver conflitos pela ferramenta:  
`git mergetool`