

# RESUMO GIT E GITHUB

---

By: Davi Ventura C. Perdigão

## Estados:

- Modificado (*modified*);
- Preparado (*staged/index*);
- Consolidado (*committed*);

## Ajuda:

- Geral
  - *git help*
- Comando específico
  - *git help add*
  - *git help commit*
  - *git help <qualquer\_comando\_git>*

## Configuração:

As configurações do GIT são armazenadas no arquivo `.gitconfig` localizado dentro do diretório do usuário do Sistema Operacional (Ex.: Windows: `C:\Users\Documents and Settings\user` ou \*nix `/home/user`).

As configurações realizadas através dos comandos abaixo serão incluídas no arquivo citado acima.

- Setar usuário (*git config --global user.name "Nome Usuário"*)
- Setar email (*git config --global user.email emailuser@mail.com.br*)
- Setar editor (*git config --global core.editor vim*)
- Setar ferramenta de merge (*git config --global merge.tool vimdiff*)
- Setar arquivos ignorados (*git config --global core.excludesfile ~/.gitignore*)
- Listar configurações (*git config --list*)

## **Ignorar Arquivos -**

Os nomes de arquivos/diretórios ou extensões de arquivos listados no arquivo `.gitignore` não serão adicionados em um repositório. Existem dois arquivos `.gitignore`, são eles:

- **Geral:** Normalmente armazenado no diretório do usuário do Sistema Operacional. O arquivo que possui a lista dos arquivos/diretórios a serem ignorados por todos os repositórios deverá ser declarado conforme citado acima. O arquivo não precisa ter o nome de `.gitignore`.

- **Por repositório:** Deve ser armazenado no diretório do repositório e deve conter a lista dos arquivos/diretórios que devem ser ignorados apenas para o repositório específico.

## Repositório Local:

- Criar novo repositório (*git init*)
- Verificar estado dos arquivos/diretórios (*git status*)
- Adicionar um arquivo em específico (*git add meu\_arquivo.txt*)
- Adicionar um diretório em específico (*git add meu\_diretorio*)
- Adicionar todos os arquivos/diretórios (*git add .*)
- Adicionar um arquivo que esta listado no .gitignore (*git add -f arquivo\_no\_gitignore.txt*)
- Comitar um arquivo (*git commit meu\_arquivo.txt*)
- Comitar vários arquivos (*git commit meu\_arquivo.txt meu\_outro\_arquivo.txt*)
- Comitar informando mensagem (*git commit meuarquivo.txt -m "new commit"*)
- Remover arquivo (*git rm meu\_arquivo.txt*)
- Remover diretório (*git rm -r diretório*)
- Exibir histórico (*git log*)
- Exibir resumo do histórico (*git log --stat*)
- Exibir informações resumidas em uma linha (*git log --pretty=oneline*)
- Exibir histórico com formatação específica (*git log --pretty=format:"%h - %an, %ar : %s"*) - %h: Abreviação do hash; %an: Nome do autor; %ar: Data; %s: Comentário.
- Exibir histórico de um arquivo específico (*git log -- <caminho\_do\_arquivo>*)
- Exibir histórico de um arquivo específico que contém uma determinada palavra (*git log --summary -S<palavra> [<caminho\_do\_arquivo>]*)
- Exibir histórico modificação de um arquivo (*git log --diff-filter=M -- <caminho\_do\_arquivo>*)
- Exibir histórico de um determinado autor (*git log --author=usuário*)
- Exibir revisão e autor da última modificação de uma bloco de linhas (*git blame -L 12,22 meu\_arquivo.txt*)
- Desfazendo alteração local (*git checkout -- meu\_arquivo.txt*)
- Desfazendo alteração local (*git reset HEAD meu\_arquivo.txt*). Se o comando reset não alterar o diretório de trabalho, alteração pode ser realizada através do comando (*git checkout meu\_arquivo.txt*)

## **Repositório Remoto:**

- Exibir os repositórios remotos (*git remote ou git remote -v*)
- Vincular repositório local com um repositório remoto (*git remote add origin git@github.com:user/repositorio*)
- Exibir informações dos repositórios remotos (*git remote show origin*)
- Renomear um repositório remoto (*git remote rename origin repositorio*)
- Desvincular um repositório remoto (*git remote rm curso-git*)
- Enviar arquivos/diretórios para o repositório remoto (*git push*)
- Atualizar os arquivos no branch atual (*git pull*)
- Buscar as alterações, mas não aplica-las no branch atual (*git fetch*)
- Clonar um repositório remoto já existente (*git clone git@github.com:user/repositorio*)