

Lista de comandos Linux para manipulação de arquivos e diretórios

Comando	Descrição
<code>ls</code>	Lista arquivos e diretórios no diretório atual.
<code>cd</code>	Navega entre diretórios. Ex: <code>cd /Documents/projeto1</code>
<code>cd ..</code>	Retorna para o diretório anterior.
<code>pwd</code>	Mostra o caminho completo do diretório atual.
<code>mkdir</code>	Cria um novo diretório. Ex: <code>mkdir projetos</code>
<code>cp</code>	Copia arquivos ou diretórios. Ex: <code>cp arquivo.txt destino/</code>
<code>touch</code>	Cria um novo arquivo vazio. Ex: <code>touch novo.txt</code>
<code>echo "Texto" >> arquivo.txt</code>	Cria/atualiza arquivo com o texto entre aspas. Ex: <code>echo "Texto" >> arquivo.txt</code>
<code>cat</code>	Mostra o conteúdo de um arquivo.

✳️ Comandos Básicos do Git

Comando	Descrição
<code>git init</code>	Inicia um novo repositório Git no diretório atual.
<code>git clone <URL></code>	Clona um repositório remoto para o seu computador.
<code>git status</code>	Mostra o status atual do repositório (arquivos modificados, não rastreados, etc.).
<code>git add <arquivo></code>	Adiciona arquivos à staging area (área de preparação).
<code>git add .</code>	Adiciona todos os arquivos modificados à staging area.
<code>git commit -m "mensagem"</code>	Cria um commit com os arquivos adicionados e uma mensagem descriptiva.
<code>git push</code>	Envia os commits locais para o repositório remoto.
<code>git pull</code>	Puxa (atualiza) os dados do repositório remoto para o local.
<code>git fetch</code>	Baixa atualizações do repositório remoto sem mesclar.
<code>git merge <branch></code>	Mescla outra branch com a branch atual.
<code>git mv <arquivo></code>	Move ou renomeia arquivos/diretórios. Ex: <code>git mv antigo.txt novo.txt</code>

Comando	Descrição
<code>git rm <arquivo></code>	Remove arquivos ou diretórios (-r para recursivo). Ex: <code>git rm arquivo.txt</code>

🌿 Trabalho com Branches

Comando	Descrição
<code>git branch</code>	Lista todas as branches locais.
<code>git branch <nome></code>	Cria uma nova branch.
<code>git checkout <nome></code>	Troca para outra branch.
<code>git checkout -b <nome></code>	Cria e já troca para a nova branch.
<code>git branch -d <nome></code>	Deleta uma branch local.

⌚ Controle de Histórico

Comando	Descrição
<code>git log</code>	Mostra o histórico de commits.
<code>git log --oneline</code>	Mostra o log de forma resumida (uma linha por commit).
<code>git diff</code>	Mostra as diferenças entre arquivos modificados e o último commit.
<code>git reset</code>	Remove arquivos da staging area.
<code>git reset --hard HEAD</code>	Desfaz todas as mudanças desde o último commit (cuidado!).

🔗 Conexão com Repositórios Remotos

Comando	Descrição
<code>git remote -v</code>	Mostra os repositórios remotos vinculados.
<code>git remote add origin <URL></code>	Adiciona um repositório remoto com o nome <code>origin</code> .
<code>git push -u origin <branch></code>	Envia a branch local para o repositório remoto e a vincula.

Outros Comandos Úteis

Comando	Descrição
<code>git config --global user.name "Seu Nome"</code>	Define seu nome global para commits.
<code>git config --global user.email "email@example.com"</code>	Define seu e-mail global.
<code>git stash</code>	Salva temporariamente mudanças não commitadas.
<code>git stash pop</code>	Recupera as mudanças salvas com <code>stash</code> .
<code>git tag</code>	Lista ou cria tags (versões).
<code>git clean -fd</code>	Remove arquivos não rastreados (⚠️ cuidado: é irreversível).

Para obter ajuda sobre os comandos no Git Bash, utilize os comandos: `help` ou `git help`.