**MANUAL RESUMIDO DE AJUDA DO GIT E GITHUB**

**ESTE TUTORIAL JÁ LEVA EM CONTA QUE O GIT ESTÁ INSTALADO E CONFIGURADO PARA USO VIA SSH EM SUA MÁQUINA**

1. **INICIANDO UM PROJETO DO ZERO DIRETAMENTE NA MÁQUINA**
   1. CRIAR UMA PASTA PARA O PROJETO OU UTILIZAR UMA JÁ EXISTENTE
   2. ABRIR O GIT BASH NESTA PASTA (NAVEGAR ATÉ A PASTA VIA TERMINAL DO GIT BASH)
   3. EXECUTAR O COMANDO: **GIT INIT** – ESTE COMANDO IRÁ INICIALIZAR O CONTROLE DO GIT NA PASTA
   4. PARA TESTAR SE DEU CERTO, RODE O COMANDO: **LS -A**
   5. VAI APARECER UM ARQUIVO OCULTO DENOMINADO **.GIT**
2. **ADICIONANDO E SALVANDO MUDANÇAS NA PASTA**
   1. REALIZE AS MUDANÇAS INICIAIS QUE DESEJA NA SUA PASTA
   2. PARA ADICIONAR AS MUDANÇAS QUE REALIZOU AO GIT, RODE O COMANDO **GIT ADD .**
   3. ESTE COMANDO COM PONTO FINAL IRÁ SALVAR TODOS OS ARQUIVOS E PASTAS ALTERADAS
   4. CASO QUEIRA SALVAR SOMENTE UM OU ALGUNS ARQUIVOS ESPECÍFICOS, RODEO O MESMO COMANDO, MAS COM O NOME DOS ARQUIVOS E PASTAS
   5. O COMANDO FICARIA ASSIM: **GIT ADD TESTE1.TXT TESTE2.TXT PASTA1**
   6. CASO QUEIRA DESFAZER O GIT ADD - **GIT RESET <NOME\_ARQUIVO>**
   7. CASO QUEIRA DESFAZER TODAS MUDANÇAS DE GIT ADD - **GIT RESET --MIXED**
   8. CASO QUEIRA REMOVER TODAS MUDANÇAS DE GIT ADD - **GIT RESET --HARD**
   9. PARA VERIFICAR SE AS MUDANÇAS ESTÃO SENDO REALIZADAS E SE EXISTEM ARQUIVOS QUE NÃO ESTÃO SENDO MONITORADOS PELO GIT AINDA, RODE O SEGUINTE COMANDO: **GIT STATUS**
   10. ELE MOSTRARÁ UM LOG COM OS ARQUIVOS E PASTAS QUE VOCÊ ADICIONOU AO GIT (MAS NÃO SALVOU AINDA) E OS DEMAIS ARQUIVOS E PASTAS QUE NÃO FORAM ADICIONADOS
   11. PARA SALVAR OS ARQUIVOS QUE VOCÊ ADICIONOU, VAMOS RODAR O SEGUINTE COMANDO: **GIT COMMIT -M "INITIAL COMMIT"**
   12. O NOME INITIAL COMMIT É UM NOME QUALQUER QUE VOCÊ DARÁ, A CADA VEZ QUE FIZER ISSO, ADICIONE NOMES SUGESTIVOS QUE INDIQUEM O QUE SIGNIFICA ESTAS MUDANÇAS QUE VOCÊ ESTÁ SALVANDO (NÃO PRECISA DAS ASPAS, NÃO USE ESPAÇOS E ACENTUAÇÕES)
   13. RODE O SEGUINTE COMANDO: **GIT LOG** – ESTE COMANDO IRÁ MOSTRAR TODOS OS COMMITS QUE VOCÊS JÁ REALIZARAM
   14. PARA VER RESUMIDO OS LOGS, RODE O COMANDO: **GIT LOG –ONELINE**
   15. PARA VER UM NÚMERO X DE LOGS, RODE O COMANDO: **GIT LOG -N 5**
   16. PARA VER OS LOGS DE UM PERÍODO ESPECÍFICO EM DIANTE: **GIT LOG --SINCE=2021-01-01**
   17. PARA VER OS LOGS DE UM PERÍODO PASSADO: **GIT LOG --UNTIL=2021-01-01**
   18. PARA VER OS LOGOS COM UM NOME ESPECÍFICO OU “PEDAÇOS” DESTE NOME: **GIT LOG --GREP=COMM**
3. **REVERTENDO E ADICIONANDO ARQUIVOS NO GIT**
   1. PARA REVERTER UM ARQUIVO QUE FOI ADICIONADO AO PROCESSO DE PREPARO PARA O COMMIT (STAGE AREA): **GIT RM --CACHED teste2.txt**
   2. PARA ADICIONAR ARQUIVOS DE UMA MESMA EXTENSÃO AO PROCESSO DE PREPARO PARA O COMMIT: **GIT ADD \*.html**
   3. PARA VERIFICAR EM DETALHES AS MODIFICAÇÕES DOS ARQUIVOS: **GIT DIFF**
   4. PARA VERIFICAR EM DETALHES AS MODIFICAÇÕES DOS ARQUIVOS NO STAGE AREA: **GIT DIFF --STAGED**
   5. TIRAR UM ARQUIVO DO STAGE AREA (QUE ESTÁ PRONTO PARA COMMIT): **GIT RESTORE --STAGED NOMEDOARQUIVO**
   6. DESFAZER AS ALTERAÇÕES NO ARQUIVO, CASO NÃO TENHA IDO PARA STAGE OU COMMIT - **GIT RESTORE NOMEDOARQUIVO**
   7. CORRIGIR O NOME DO COMMIT QUE FOI DEFINIDO: **GIT COMMIT --AMEND -M "NOVA MENSAGEM"**. ELE NÃO VAI REMOVER O COMMIT ANTERIOR, MAS CRIAR UM NOVO COM O NOME ATUALIZADO PARA REMOVER ARQUIVOS QUE NÃO ESTÃO NO STAGE PARA COMMIT (NÃO RASTREADOS) - GIT CLEAN -F
   8. COMO REVERTER UM COMMIT - CONTA O NÚMERO DE HEAD COM -1 (PRIMEIRO DEPOIS DO ATUAL) - **GIT REVERT HEAD~3** E FAZ UM NOVO COMMIT (NÃO SE APAGA COMMIT NO GIT)
   9. COMO REVERTER UM COMMIT - PEGA O HASH COM **GIT LOG --ONELINE** - **GIT REVERT HASH** E FAZ UM NOVO COMMIT (NÃO SE APAGA COMMIT NO GIT)
   10. QUANDO OS ARQUIVOS JÁ ESTÃO RASTREADOS, PODEMOS USAR UM ATALHO QUE FAZ O ADD E COMMIT DE UMA VEZ SÓ: **GIT COMMIT -AM "MODIFICAR ALGO"**
   11. PARA VER ALTERAÇÕES QUE OCORRERAM EM UM COMMIT MARCADAS POR CORES: **GIT SHOW HASH --COLOR-WORDS**
   12. PARA REMOVER UM ARQUIVO PERMANENTEMENTE VIA TERMINAL DO GIT BASH: **GIT RM FILE1.TXT**
4. **TRABALHANDO COM O GIT E GITHUB (ENVIADO ALTERAÇÕES DE UM REPOSITÓRIO LOCAL PARA O SERVIDOR REMOTO)**
   1. CRIE UM REPOSITÓRIO NO GITHUB
   2. BUSQUE A PARTE DE CLONAR O REPOSITÓRIO E CLIQUE EM SSH
   3. COPIE A URL GERADA
   4. ACESSE O LOCAL NO SEU COMPUTADOR QUE DESEJA SALVAR UMA CÓPIA DO REPOSITÓRIO E ENTRE NO GIT BASH NESTE LOCAL
   5. RODE O COMANDO - **GIT INIT**
   6. RODE O COMANDO DO SSH, SIMILAR A ESTE EXEMPLO - **GIT REMOTE ADD ORIGIN GIT@GITHUB.COM:NUNESFB/TESTE.GIT**
   7. PARA VERIFICAR QUE TUDO FUNCIONOU CORRETAMENTE, EXECUTE O COMANDO - **GIT REMOTE -V**
   8. REALIZE AS MUDANÇAS INICIAIS QUE DESEJA NA SUA PASTA
   9. PARA ADICIONAR AS MUDANÇAS QUE REALIZOU AO GIT, RODE O COMANDO **GIT ADD .**
   10. ESTE COMANDO COM PONTO FINAL IRÁ SALVAR TODOS OS ARQUIVOS E PASTAS ALTERADAS
   11. CASO QUEIRA SALVAR SOMENTE UM OU ALGUNS ARQUIVOS ESPECÍFICOS, RODEO O MESMO COMANDO, MAS COM O NOME DOS ARQUIVOS E PASTAS
   12. O COMANDO FICARIA ASSIM: **GIT ADD TESTE1.TXT TESTE2.TXT PASTA1**
   13. PARA VERIFICAR SE AS MUDANÇAS ESTÃO SENDO REALIZADAS E SE EXISTEM ARQUIVOS QUE NÃO ESTÃO SENDO MONITORADOS PELO GIT AINDA, RODE O SEGUINTE COMANDO: **GIT STATUS**
   14. ELE MOSTRARÁ UM LOG COM OS ARQUIVOS E PASTAS QUE VOCÊ ADICIONOU AO GIT (MAS NÃO SALVOU AINDA) E OS DEMAIS ARQUIVOS E PASTAS QUE NÃO FORAM ADICIONADOS
   15. PARA SALVAR OS ARQUIVOS QUE VOCÊ ADICIONOU, VAMOS RODAR O SEGUINTE COMANDO: **GIT COMMIT -M "INITIAL COMMIT"**
   16. O NOME INITIAL COMMIT É UM NOME QUALQUER QUE VOCÊ DARÁ, A CADA VEZ QUE FIZER ISSO, ADICIONE NOMES SUGESTIVOS QUE INDIQUEM O QUE SIGNIFICA ESTAS MUDANÇAS QUE VOCÊ ESTÁ SALVANDO (NÃO PRECISA DAS ASPAS, NÃO USE ESPAÇOS E ACENTUAÇÕES)
   17. RODE O SEGUINTE COMANDO: **GIT LOG** – ESTE COMANDO IRÁ MOSTRAR TODOS OS COMMITS QUE VOCÊS JÁ REALIZARAM
   18. VAMOS ENVIAR AS ALTERAÇÕES PARA A BRANCH MASTER AGORA, QUE É A BRANCH QUE CONTÉM O CONTEÚDO PRINCIPAL SEMPRE
   19. RODE O COMANDO - **GIT PUSH ORIGIN MASTER**

COM ISSO, VOCÊ ESTÁ ENVIANDO AS ALTERAÇÕES FEITAS LOCALMENTE PARA A BRANCH PRINCIPAL, MASTER , DO SEU REPOSITÓRIO REMOTO, ORIGIN

1. **TRABALHANDO COM O GIT E GITHUB (CLONANDO UM REPOSITÓRIO E CRIANDO BRANCHS)**
   1. BUSQUE POR UM REPOSITÓRIO NO GITHUB
   2. FAÇA O CLONE DELE CLICANDO EM SSH E COPIANDO A URL
   3. EM UMA PASTA ABRAR O TERMINAL GIT BASH E RODE O COMANDO – GIT CLONE GIT@GITHUB.COM:NUNESFB/CURSO\_NODEJS\_MODULO02.GIT
   4. ENTRE NA PASTA DO REPOSITÓRIO CLONADO COM O GIT BASH
   5. PARA CRIAR UMA BRANCH, QUE É UMA CÓPIA DO PROJETO ORIGINAL E MODIFICAR OS ARQUIVOS E PASTAS NESTA BRANCH, RODE O COMANDO: **GIT BRANCH “NOME”**
   6. PARA SABER A LISTA DE BRANCHES QUE EXISTEM NO SEU REPOSITÓRIO - **GIT BRANCH --LIST**
   7. PARA TROCAR DE BRANCH, EXECUTE O COMANDO - **GIT CHECKOUT NOMEDABRANCH**
   8. EXECUTE ALGUMA MUDANÇA E FAÇA OS COMANDOS DE **GIT ADD** E **GIT COMMIT**
   9. AGORA VAMOS SUBIR PARA O SERVIDOR REMOTO DO GIT AS MUDANÇAS QUE FIZEMOS EM NOSSA BRANCH, RODE O COMANDO SIMILAR A ESTE - **GIT PUSH --SET-UPSTREAM ORIGIN TESTE**
   10. EXECUTE ALGUMA MUDANÇA E FAÇA OS COMANDOS DE **GIT ADD** E **GIT COMMIT**
   11. AGORA VAMOS SUBIR NOVAMENTE, MAS USE SOMENTE O COMANDO – **GIT PUSH**
   12. AGORA VOCÊ PODE IR NO GITHUB E ENTRAR NA SUA BRANCH, VAI TER UM BOTÃO CHAMADO COMPARE E PULL REQUEST, QUE VAI SOLICITAR AOS AUTORES DO REPOSITÓRIO QUE VEJAM SUAS ALTERAÇÕES E AVALIEM SE DEVEM SER REALIZADAS, EM QUE A MUDANÇA CASO SEJA ACEITA, IRÁ OCORRER NA BRANCH MASTER
   13. CASO VOCÊ QUEIRA PEGAR AS VERSÕES MAIS ATUALIZADAS QUE ESTÃO NO REPOSITÓRIO, VOCÊ RODA O COMANDO – **GIT PULL**



