**Tutorial: Como usar o Git e o GitHub**

-Para saber a versão do seu git, vá no terminal e digite:

git --version

-Abaixo segue uma lista de comandos sugeridos pelo próprio git através do comando:

git --help

Texto

Descrição gerada automaticamente

-Com sua conta já criada no GitHub, vá para a aba repositórios e crie um repositório;

-Copie o link gerado por este repositório;

-No seu computador crie uma pasta onde você deseja armazenar os arquivos deste repositório e inicie o git Bash dentro dela;

-No terminal do Git digite:

git clone (seu link)

Com isso o repositório será clonado para a sua máquina;

OBS: Para colar o link no terminal, aperte o botão de scroll do mouse.

-Caso queira iniciar o repositório pela sua máquina, abra o git Bash nesta pasta e digite o comando:

git init

-Caso você queira desfazer a inicialização de um repositório em alguma pasta, vá na respectiva, abra o terminal do git e digite:

rm -r .git

-Quando um repositório git é iniciado, aparecerá a palavra “master”, ela é o indicativo de que esta pasta é um repositório. Contudo, algumas empresas podem utilizar como padrão a palavra “main”, mas isto não interfere no uso;

(a palavra master que aparece significa que já estamos dentro da nossa branch master, tempo cronológico principal)

(também é criado uma pastinha. git, não apague ela)

-Para mudar a Master para Main usa-se os comandos:

git branch -M “main”

Texto

Descrição gerada automaticamente com confiança média

-Para enviarmos os arquivos criados nesta pasta para a área de “staging”, área de espera, dizer ao git que queremos incluir atualizações a um arquivo específico no próximo commit, usamos:

Git add (nome do arquivo. Extensão)

-Para verificar se os arquivos da “staging” estão atualizados de acordo com as alterações que estão sendo feitas no respectivo arquivo, devemos usar o comando:

Git status

-Para salvar as alterações no repositório local após o staging, devemos aplicar um comando junto de uma mensagem, essa mensagem deve ser clara informando a alteração que foi feita:

Git commit -m “mensagem desejada”

-Dando outro git status verificamos se tudo deu certo;

-Antes de enviar as alterações para o repositório remoto, é preciso saber se já existe alguma conexão entre ambos (local e remoto), se não houver ainda conexão digitamos o comando:

Git remote add origin (link do repositório)

-Para “empurrar” os arquivos locais para o repositório remoto utiliza-se o comando:

Git push -u origin main

Texto

Descrição gerada automaticamente

Após essa ação, irá abrir uma aba para se logar no GitHub (caso tenha usado o padrão HTTP).

Pronto, seu projeto está salvo no repositório do GitHub.

-Caso deseje criar uma nova branch (ramificação secundária) utilize os comandos:

Git checkout -b “nome da nova branch”

git add .

git commit -m “nome do arquivo na nova branch”

git push origin “nome da nova branch”

-Para sair dessa nova branch e voltar para a master (main), digite:

Git checkout main

-Agora, para mesclar as duas branchs, utilize:

Git merge (nome da nova branch)

Git push origin main

-Por fim, para manter um repositório clonado atualizado em sua máquina, utilize:

Git pull