**TUTORIAL PARA UTILIZAR O GIT**

* Para iniciar um repositório:

Crie uma pasta, já com o nome do seu repositório e execute o comando dentro dela:

**git init**

* Primeiro commit:

Adicione o arquivo a um commit com o comando:

**git add README.md**

* Escreva o comando de commit incluindo uma mensagem que explique o que sua alteração faz no repositório.

**git commit -m "Adiciona arquivo README.md"**

* Para realizar edições ao seu repositório, pode utilizar o comando git add, adicionando um commit igual o exemplo a seguir:

**git add *nome do arquivo***

**git commit**

* Ao compartilhar uma alteração com sua equipe, podemos utilizar o comando git push, desta forma:

**> git push origin**

* Para desfazer alterações locais completamente, sem salvá-las, utilize:

**git clean -fd.**

* Caso queira resolver algum conflito com sua merge, terá alguns passos a seguir:

**Abaixo do nome do seu repositório, clique em *Pull requests*.**

**Na lista "*Pull Requests*", clique na pull request que tem um conflito de merge que você deseja resolver.**

**Próximo à parte inferior da *pull request*, clique em *Resolve conflicts* (Resolver conflitos).**

* Quando quiser iniciar um novo recurso, crie um novo branch a partir do branch main usando o comando:

**git branch new\_branch**

Então, você vai poder usar o ***git checkout new\_branch*** para mudar para esse branch.

* Caso necessite encontrar bugs em seu repositório, podemos utilizar o comando:

**git bisect start**

Para informar para que o bug continua acontecendo use:

**git bisect bad**

Se o teste não falhar, use:

**git bisect good**

Para encerrar o git bisect use:

**git bisect reset**

* O cherry-pick recebe como argumentos uma lista de identificadores de commits. Isto é, ele é usado da seguinte forma:

**git cherry-pick commit1 commit2 ...**

Com isso, ele irá pegar a diferença que cada um destes commits introduziu (pode ser vista com git show commitX) e aplicar na posição (HEAD) em que você está.