**ATIVIDADE DE GIT E GITHUB- 22/01/24**

1- O que é o Git e qual é a sua principal função no controle de versão de código-fonte?

O Git é um projeto de código aberto maduro e com manutenção ativa desenvolvido em 2005 por Linus Torvalds, o famoso criador do kernel do sistema operacional Linux. Um número impressionante de projetos de software depende do Git para controle de versão, incluindo projetos comerciais e de código-fonte aberto.

O controle de versão protege o código-fonte contra catástrofes e a possível degradação causada por erro humano e consequências ruins.

font :https://www.atlassian.com/br/git/tutorials/what-is-version-control#:~:text=O%20controle%20de%20vers%C3%A3o%20protege,e%20modificando%20c%C3%B3digo%2Dfonte%20existente.

2- Como podemos criar um novo repositório Git em um projeto existente no seu sistema local?

Para criar um novo repositório, você vai usar o comando git init. Portanto para criar um novo repositório em um projeto ja existente,basta da o comando cd <nome da pasta do projeto que quer acessar> e dar esse comando esse ‘git init’ em seguida.Criando aí um repositório novo em um projeto já existente.

font:

<https://www.atlassian.com/br/git/tutorials/setting-up-a-repository#:~:text=Para%20criar%20um%20novo%20reposit%C3%B3rio,comando%20cria%20um%20novo%20subdiret%C3%B3rio%20>.

3- Qual a diferença entre "git add" e "git commit" no contexto do Git?

O comando git add vai estar adicionando uma arquivo ao seu projeto e o git commit vai estar emitindo a sua mensagem escrita , informando o que foi feito no projeto.Portanto durante a realização do projeto e esteja sendo feito alguma mudança, o comando git commit é importante pois pode se referindo a essas mudanças.

font:

https://pt.stackoverflow.com/questions/572430/comandos-git-git-add-e-git-commit#:~:text=git%20add%20significa%20Adicionar%2C%20ou,voc%C3%AA%20realizou%20em%20seu%20projeto.

4- Explique o propósito do comando "git push" e quando você o utilizaria?

O comando git push vai estar empurrando os arquivos para o repositório.Quando se está mexendo no seu projeto e quer enviar essas novas modificações para o repositório, como atualização do projeto se usa esse comando.

font:

<https://blog.geekhunter.com.br/comandos-git-mais-utilizados/#Git_push>

5- O que são branches no Git e por que são úteis em um projeto de desenvolvimento?

O comando git branch permite criar, listar, renomear e excluir ramificações. Ele não permite alternar entre as ramificações ou reunir um histórico bifurcado de novo. Por esse motivo, o comando git branch é muito integrado aos comandos git checkout e git merge.Uma das mais importantes funções do Git é o controle das ramificações (branches) do desenvolvimento que ajudam a melhorar a criação de um projeto.

font:

<https://www.atlassian.com/br/git/tutorials/using-branches#:~:text=O%20comando%20git%20branch%20permite,git%20checkout%20e%20git%20merge>.

<https://www.hostinger.com.br/tutoriais/git-branch#:~:text=Uma%20das%20mais%20importantes%20fun%C3%A7%C3%B5es,a%20cria%C3%A7%C3%A3o%20de%20um%20projeto>.

6- .Como podemos verificar o status atual do repositório local no Git?

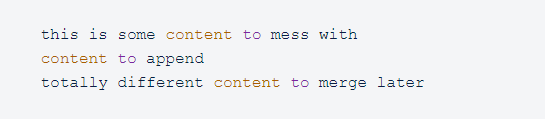
O comando git status exibe as condições do diretório de trabalho e da área de staging.

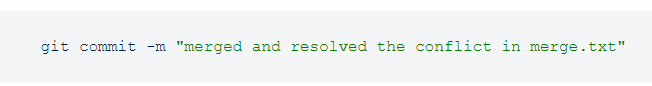
font:

https://www.atlassian.com/br/git/tutorials/inspecting-a-repository#:~:text=O%20comando%20git%20status%20exibe,est%C3%A3o%20sendo%20monitorados%20pelo%20Git.

7- Explique o que é um conflito de merge no Git e como você resolveria esse problema

Os conflitos de merge ocorrem quando alterações concorrentes são feitas na mesma linha de um arquivo ou quando uma pessoa edita um arquivo e outra pessoa exclui o mesmo arquivo.

O modo mais direto de resolver conflitos de merge é editar o arquivo com conflitos. Abra o arquivo merge.txt no editor que você preferir. Como exemplo, a gente vai apenas remover todos os divisores com conflito. O conteúdo do arquivo merge.txt modificado deve se parecer com o seguinte:

Quando o arquivo tiver sido editado, use git add merge.txt para fazer stage do novo conteúdo que passou pelo merge. Para finalizar o merge, crie o commit executando:

O Git percebe que o conflito foi resolvido e cria o commit de merge para finalizar o processo.

link:

https://www.atlassian.com/br/git/tutorials/using-branches/merge-conflicts

8. O que é o GitHub e como ele se diferencia do Git em termos de funcionalidades?

O **Github** é uma “rede social dev” em que é possível armazenar e compartilhar projetos de desenvolvimento de software.

O **Git** é um sistema de controle de versão de arquivos; em outras palavras, é responsável por guardar o histórico de alterações sempre que alguém modificar algum arquivo que está sendo monitorado por ele.

Portanto são ideias diferentes,mas se complementam.

link:

<https://www.alura.com.br/artigos/o-que-e-git-github#:~:text=O%20Github%20%C3%A9%20uma%20%E2%80%9Crede,est%C3%A1%20sendo%20monitorado%20por%20ele>.

9. Qual a diferença entre um repositório público e privado no GitHub?

A diferença é que um repositório público todos possuem acesso ao seu trabalho,criar um clone,baixar o código.Um repositório privado apenas aquelas que possuem acesso que foi permitido pelo proprietário do repositório podem ter direito ao código que lhe foi permitido.

font:

https://pt.stackoverflow.com/questions/156311/todos-podem-ver-meus-projetos-no-github#:~:text=Projetos%20p%C3%BAblicos%3A%20qualquer%20um%20pode,liberar%20ter%C3%A3o%20acesso%20ao%20c%C3%B3digo.

10. Como podemos realizar o processo de clonagem (clone) de um repositório do GitHub para o seu computador local?

Para começar, acesse sua conta no github e clique em **Repositórios** e acesse o repositório desejado. No canto superior direito, você verá a opção <>Code.Ao clicar em **<>**code abrirá um menu de opções, onde poderá escolher a URL que deseja copiar para fazer o clone. Seja com o HTTP ou mediante uma chave SSH.Abra a janela do terminal Git Bash e acesse a pasta Desktop (pode utilizar qualquer outra pasta do seu diretório principal ou ou criar uma própria), onde será anexada a cópia do projeto, utilizando o comando **$ git clone** e colando.Pronto clonado o repositório.

link:

<https://www.alura.com.br/artigos/clonando-repositorio-git-github>

11.O que são Pull Requests no GitHub e qual é o seu propósito no desenvolvimento colaborativo?

Para propor as mudanças que você acabou de fazer para outros desenvolvedores trabalhando no mesmo projeto, você deve criar um pull request. São eles que fazem ser tão fácil de trabalhar junto em projetos, já que eles são a principal ferramenta de colaboração no GitHub.

font:

<https://www.hostinger.com.br/tutoriais/o-que-github#:~:text=4.-,Crie%20Pull%20Requests%20no%20GitHub,ferramenta%20de%20colabora%C3%A7%C3%A3o%20no%20GitHub>.

12. Como podemos adicionar um arquivo ao arquivo .gitignore e qual é a sua finalidade?

O arquivo .gitignore é um arquivo de texto que diz ao Git quais arquivos ou pastas ele deve ignorar em um projeto.Um arquivo .gitignore local geralmente é colocado no diretório raiz de um projeto. Você também pode criar um arquivo .gitignore global e todas as entradas que estiverem naquele arquivo serão ignoradas em todos os seus repositórios do Git.Para criar um arquivo .gitignore local, crie um arquivo de texto e dê a ele o nome de .gitignore (lembre-se de incluir o . no começo). Em seguida, edite esse arquivo conforme necessário. Cada nova linha deve listar um arquivo ou pasta adicional que você quer que o Git ignore.

font:

https://www.freecodecamp.org/portuguese/news/gitignore-explicado-o-que-e-o-gitignore-e-como-adiciona-lo-ao-seu-repositorio/

13. Explique o que é um fork no GitHub e em que situações você pode querer realizar um fork de um repositório.

Os forks permitem fazer alterações em um projeto sem afetar o repositório original, também conhecido como repositório "upstream".

Os forks geralmente são usados para iterar ideias ou alterações antes de serem propostas de volta para o repositório upstream, como em projetos código aberto ou quando um usuário não tem acesso de gravação ao repositório upstream. Para saber mais, confira "Trabalhar com bifurcações".

font:

<https://docs.github.com/pt/pull-requests/collaborating-with-pull-requests/working-with-forks/fork-a-repo>

https://docs.github.com/pt/pull-requests/collaborating-with-pull-requests/working-with-forks/about-forks

14. Como podemos visualizar o histórico de commits em um repositório Git?

A ferramenta mais básica e poderosa para fazer isso é o comando git log . Por padrão, sem argumentos, git log lista os commits feitos neste repositório em ordem cronológica inversa; isto é, o commit mais recente aparece primeiro.Dando o retorno verátodos os commits feitos até o momento.

font:

https://dev.to/womakerscode/tutorial-git-ver-o-historico-de-commits-git-log-3ep7

15.Qual é a função do comando "git pull" e quando ele deve ser utilizado no desenvolvimento colaborativo com o Git e GitHub?

O comando git pull serve para recuperar e baixar o conteúdo de um repositório remoto e atualizar o repositório local assim que ele for baixado. Isso é muito útil em fluxos de trabalho de colaboração que precisam mesclar alterações upstream remotas no repositório local.

font:

https://www.locaweb.com.br/blog/temas/codigo-aberto/git-pull-aprenda-a-usar-o-comando/#:~:text=O%20comando%20git%20pull%20serve,upstream%20remotas%20no%20reposit%C3%B3rio%20local.