



RONAN MENEZES

Atividade 01

O que é o Git₁ e qual é a sua principal função no controle de versão de código-fonte?₂

resposta:

- ₁Git é um Sistema de Controle de Versão distribuído, o que significa que é uma ferramenta útil para rastrear facilmente alterações em seu código, colaborar e compartilhar o mesmo.
- ₂No Git, a cópia de trabalho de todo desenvolvedor do código também é um repositório que pode conter o histórico completo de todas as alterações.

Atividade 02

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

resposta:

Usando o comando `git init` .

(A execução desse comando cria um novo subdiretório `.git` no diretório de trabalho atual.)

Atividade 03

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

resposta:

Git add adiciona arquivos modificados a uma área temporária, a staging area, sem que haja um comprometimento com a fixação dessas modificações.

Git commit “se compromete” com as modificações que foram adicionadas e gera um commit, que é um conjunto das diferenças existentes em relação a um commit anterior na árvore de versionamento.

Atividade 04

Explique o propósito do comando "git push" e quando você o utilizaria.

resposta:

Ele vai colocar tudo o que você fez em um local onde qualquer pessoa colaboradora possa ver e pegar suas alterações, é um comando que possibilita que as alterações da sua máquina local sejam enviadas para uma máquina remota.

Atividade 05

O que são branches no Git₁ e por que são úteis em um projeto de desenvolvimento?₂

resposta:

₁ Branches no Git são semelhantes a um ramo de uma árvore, onde o tronco seria a base do código. Desse modo é possível criar diversos ramos e fazer alterações, enquanto a base permanece intacta.₂ O recurso é muito útil pois funciona como um controle de origem que permite separar a evolução da programação em diferentes subdivisões.

Atividade 06

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

resposta:

Através do comando `git status`.

Atividade 07

Explique o que é um conflito de merge no Git₁ e como você resolveria esse problema.₂

resposta:

₁ 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.₂ Pelo o que eu entendi, o “main” teria que verificar e alterar manualmente as linhas dos códigos onde é necessário fazer tais alterações.

Atividade 08

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

resposta:

O Git é uma ferramenta de controle de versão local que possibilita que os desenvolvedores salvem snapshots de seus projetos ao longo do tempo, sendo mais adequado para uso individual. Por outro lado, o GitHub é uma plataforma online que integra as funcionalidades de controle de versão do Git, permitindo sua utilização de maneira colaborativa.

Atividade 09

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

resposta:

Sendo público qualquer pessoa tem acesso, por outro lado, quando o repositório é privado, apenas as pessoas selecionadas podem acessá-lo. (É tipo uma conta no instagram aberta ou pvd).

Atividade 10

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

resposta:

É só digitar `git clone` e colar a URL já copiada.

Atividade 11

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

resposta:

Pull Request são solicitações feitas por um colaborador de um repositório para incorporar suas alterações propostas ao projeto principal. Quando alguém faz modificações em uma cópia de um repositório e deseja integrar essas alterações ao repositório original, essa pessoa envia um Pull Request.

Atividade 12

Como podemos adicionar um arquivo ao arquivo `.gitignore`₁ e qual é a sua finalidade?₂

resposta:

- ₁ Se você deseja que um arquivo seja ignorado após um commit, é necessário removê-lo do repositório e, em seguida, adicionar uma regra no arquivo `.gitignore` para excluí-lo. Ao utilizar a opção `--cached` com o comando `git rm`, o arquivo será removido do repositório, mas ainda permanecerá no seu diretório atual como um arquivo ignorado.
- ₂ Sua principal finalidade é evitar que arquivos desnecessários ou sensíveis ao contexto (como senhas, chaves de acesso, arquivos temporários) sejam incluídos acidentalmente no repositório Git

Atividade 13

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

resposta:

Fork é basicamente uma cópia pessoal do repositório, ele é muito útil pois permitem fazer alterações em um projeto sem afetar o repositório original.

Atividade 14

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

resposta:

A principal ferramenta para visualizar o histórico de commits em um repositório git é o comando git log. Quando usado sem argumentos, o git log exibe os commits em ordem cronológica inversa, mostrando o commit mais recente no topo da lista.

Atividade 15

Qual é a função do comando "git pull"₁ e quando ele deve ser utilizado no desenvolvimento colaborativo com o Git e GitHub?₂

resposta:

- ₁O comando git pull é usado para buscar e baixar conteúdo de repositórios remotos e fazer a atualização imediata ao repositório local para que os conteúdos sejam iguais.
- ₂Ele geralmente é utilizado quando você deseja atualizar seu repositório local com as últimas alterações feitas por outros colaboradores no repositório remoto, garantindo assim a sincronia com o estado mais recente do projeto.

FONTES:

[HTTPS://WWW.ATLASSIAN.COM/BR/GIT/TUTORIALS/WHAT-IS-GITFSA](https://www.atlassian.com/br/git/tutorials/what-is-gitfsa)

[HTTPS://WWW.CODAQUI.DEV/TRILHAS/GITHUB-STARTER/](https://www.codaqui.dev/trilhas/github-starter/)

[HTTPS://PT.STACKOVERFLOW.COM/QUESTIONS/572430/COMANDOS-GIT-GIT-ADD-E-GIT-COMMIT](https://pt.stackoverflow.com/questions/572430/comandos-git-git-add-e-git-commit)

[HTTPS://BLOG.BETRYBE.COM/GIT/GIT-PUSH/](https://blog.betrybe.com/git/git-push/)

[HTTPS://WWW.LOCAWEB.COM.BR/BLOG/TEMAS/CODIGO-ABERTO/GIT-BRANCH-COMO-USAR-E-GERAR-RAMIFICACOES/](https://www.locaweb.com.br/blog/temas/codigo-aberto/git-branch-como-usar-e-gerar-ramificacoes/)

[HTTPS://DOCS.GITHUB.COM/PT/PULL-REQUESTS](https://docs.github.com/pt/pull-requests)

FONTES:

[HTTPS://WWW.ALURA.COM.BR/ARTIGOS/CLONANDO-REPOSITORIO-GIT-GITHUB](https://www.alura.com.br/artigos/clonando-repositorio-git-github)

[HTTPS://WWW.ATLASSIAN.COM/BR/GIT/TUTORIALS/SAVING-CHANGES/GITIGNORE](https://www.atlassian.com/br/git/tutorials/saving-changes/gitignore)

[HTTPS://PT.STACKOVERFLOW.COM/QUESTIONS/143458/O-QUE-ESSAS-PALAVRAS-SIGNIFICAM-NO-GIT-GITHUB-FORK-CLONE-TRACK](https://pt.stackoverflow.com/questions/143458/o-que-essas-palavras-significam-no-git-github-fork-clone-track)

[HTTPS://GIT-SCM.COM/BOOK/PT-BR/V2/FUNDAMENTOS-DE-GIT-VENDO-O-HIST%C3%B3RICO-DE-COMMITS](https://git-scm.com/book/pt-br/v2/fundamentos-de-git-vendo-o-historico-de-commits)

F I M