Execício Git

Pergunta 1: Explica a finalidade dos 8 comandos git que aprendeste (init, status, log, add, commit, push, clone, pull)

Pergunta 2: Quem criou o sistema GIT e para que serve?

Pergunta 3: Explica por palavras tuas os principais conceitos da gestão de projetos de software com o GIT. (Working Directory, Staging, Repository, Remote Repo, Etc...)

Pergunta 4: Pesquisa na Internet 3 sistemas de armazenamento de código tipo GitHub e indica o seu endereço online (URL).

Pergunta 5: Dá a tua opinião pessoal sobre a utilidade desta tecnologia e como pensas vir a usá-la.

# Respostas:

1. **Git init:** cria, começa/cria os ficheiros pra “instalar” o git.

**Git Status:** mostra como está o estado projeto, se ainda tem commits e adds por fazer, também mostra o que foi ou não foi modificado

**Git Log:** O git log é um pouco pareceido com o git status mas mostra apenas o que foi feito, mostra o histórico.

**Git add:** Envia os ficheiros para a área de staging para o próximo commit

**Git Commit:** Cria um commit com as alterações que foram feitas no add, **Git**

**Git Push:** Envia as alterações locais para um repositório remoto

**Git Clone:** Faz um clone dos ficheiros que estão em outro repositório pelo URL

**Git pull:** Busca as alterações de um repositório remoto e adapta ao seu repositório local.

1. Quem Criou o Git foi Linus Torvalds em 2005. O Git é um sistema de controle de versão que guarda o histórico do projeto, permite voltar atrás, criar ramificações (**branches**) e trabalhar em equipe sem perder alterações. É muito usado junto com plataformas como GitHub e GitLab para colaboração.
2. Conceitos importantes da gestão de projetos de software com o GIT:

Repositório: É onde o projeto está guardado. Pode exixtir tanto o local quanto o remoto

Commits: commit: É o que envia as modificações feitas no projeto para o staging, possibilitando por mensagens dizendo o que foi feito.

Staging: É onde os ficheiros vão antes de ir para o repositório.

Working directory: É onde o projeto está sendo trabalhado, é o diretório de trabalho

Remote Repo: Repositório Remoto, em onde os ficheiros são guardados em outro local, como ex: Github.

**🔹 GitLab**

* Código aberto (pode instalar no seu servidor).
* Muito usado em empresas.
* Forte em **CI/CD integrado**.
* Mais controle de **permissões e pipelines** que o GitHub.

## **🔹 Bitbucket**

* Integrado com ferramentas Atlassian (Jira, Confluence, Trello).
* Foco maior em **times corporativos**.
* Suporte a **Git** (e antes também Mercurial, descontinuado).
* Bom para quem já usa ferramentas Atlassian.

## **🔹 Gitea**

* Leve, **self-hosted** (você pode instalar no seu servidor).
* Totalmente gratuito e de código aberto.
* Indicado para quem quer **independência de serviços externos**.
* Menos recursos prontos que GitHub/GitLab, mas bem simples e rápido.

1. Esse tipo de tecnologia é muito útil e importante, porque facilita a gestão de projetos que têm muitas pessoas a trabalharem juntas, pois todos tem acesso à sua versão mais atualizada, também facilita na questão de quem trabalha em múltiplos computadores como tudo é guardado remotamente, podemos só fazer um simples comando e já temos tudo pra continuar a trabalhar de qualquer lugar, assim evita muitos problemas e dores de cabeça.