



Instruções: *Todos os programas devem ser resolvidos utilizando os conceitos de Programação Orientada a Objetos, a linguagem Java™ e os conceitos de Java Persistence API.*

02 - INTRODUÇÃO AO GIT

Durante o desenvolvimento das aplicações, o código passa por diversas mudanças e muitas vezes é preciso realizar testes e voltar a versões anteriores. Quando não usamos um sistema de controle de versão esse gerenciamento de mudanças é feito por meio de cópias das pastas e arquivos do projeto, processo que pode se mostrar inseguro e levar a erros no projeto.

Nesse contexto entra o Git, um Sistema de Controle de Versão que nos permite controlar as mudanças ocorridas nos arquivos de forma segura. Com ele podemos persistir mudanças no código e voltar atrás quando for necessário sem precisar fazer cópias manuais.

Foi criado em 2005 por Linus Torvalds, o mesmo criador do Linux, que estava descontente com o BitKeeper, o sistema de controle de versão utilizado no desenvolvimento do kernel do Linux.

Atualmente, conhecer bem como utilizar o Git é uma habilidade importante para uma carreira bem-sucedida no desenvolvimento de software.

1. Versionando código com Git

- 1.1.** Acesse a unidade **C:** e crie uma pasta chamada **repositorios** e, dentro desta, crie a pasta **citacoes**.
- 1.2.** Abra o Bloco de Notas e salve, dentro da pasta **citacoes**, o arquivo **jobs.txt** com o seguinte conteúdo:

```
"Cada sonho que você deixa pra trás, é um pedaço do seu futuro que deixa de existir"
```
- 1.3.** Transforme o diretório **citacoes** em um repositório do Git.
- 1.4.** Verifique a situação do arquivo no repositório Git.
- 1.5.** Faça com que o arquivo **jobs.txt** seja rastreado pelo Git.
- 1.6.** Verifique a situação do arquivo no repositório Git novamente.
- 1.7.** Execute o comando para gravar as mudanças no repositório com a mensagem **"Arquivo inicial de citações"**.
- 1.8.** Verifique a situação do arquivo no repositório Git novamente.
- 1.9.** Insira mais uma linha no arquivo **jobs.txt** com o conteúdo abaixo e salve-o:

```
"Eu tenho tanto orgulho das coisas que não fizemos quanto das coisas que fizemos. Inovação é dizer 'não' a milhares de coisas"
```
- 1.10.** Depois disso, execute `git status` novamente.
- 1.11.** Adicione as alterações ao repositório local e grave as mudanças com a mensagem **"Inserindo nova citação"**.



Instruções: *Todos os programas devem ser resolvidos utilizando os conceitos de Programação Orientada a Objetos, a linguagem Java™ e os conceitos de Java Persistence API.*

1.12. Execute o comando para verificar o histórico (log) das alterações gravadas no repositório.

2. GitHub

O GitHub é um serviço web de hospedagem de código que possibilita o controle de versão de projetos de software. Construído sobre o Git, esse serviço abstrai do programador questões de infraestrutura, disponibilizando em um servidor web os seus projetos, de modo que você ou uma equipe (centralizada ou distribuída), possa trabalhar no mesmo projeto a partir de um controle de versão rápido e simples.

2.1. Acesse o github (<https://github.com/>) e crie uma conta.

2.2. Crie um repositório chamado **etec-aula2**.

2.3. Copie o endereço do repositório criado.

2.4. Acesse a pasta **repositorios** e efetue um `git clone <repositorio>`.

2.5. Abra o Bloco de Notas e salve, dentro da pasta **etec-aula2**, o arquivo **jobs.txt** com o seguinte conteúdo:

"Cada sonho que você deixa pra trás, é um pedaço do seu futuro que deixa de existir"

2.6. Verifique a situação do arquivo no repositório Git.

2.7. Faça com que o arquivo **jobs.txt** seja rastreado pelo Git.

2.8. Verifique a situação do arquivo no repositório Git novamente.

2.9. Execute o comando para gravar as mudanças no repositório com a mensagem "**Arquivo inicial de citações**".

2.10. Efetue um `git push` para enviar as alterações ao repositório remoto do **Github**.

2.11. Abra a página web do **Github** e verifique se o arquivo foi inserido.

2.12. Insira mais uma linha no arquivo **jobs.txt** com o conteúdo abaixo e salve-o:

"Eu tenho tanto orgulho das coisas que não fizemos quanto das coisas que fizemos. Inovação é dizer 'não' a milhares de coisas"

2.13. Depois disso, execute `git status` novamente.

2.14. Adicione as alterações ao repositório local e grave as mudanças com a mensagem "**Inserindo nova citação**".

2.15. Efetue um `git push` para enviar as alterações ao repositório remoto do **Github**.

2.16. Abra a página web do **Github** e verifique se o arquivo foi atualizado.