

## Trabalho 2: Git e GitHub

Professor: Diego da Silva de Medeiros

diegomedeiros@ifsc.edu.br

---

Este trabalho consiste na execução das atividades a seguir, e entrega dos resultados em forma de relatório. Indique no relatório o maior número de detalhes possível, incluindo os comandos utilizados, a ideia, dificuldades encontradas, erros solucionados, etc. O trabalho deve ser feito individualmente, contando com a colaboração de um colega para realizar partes específicas indicadas na atividade. Como sugestão, colabore com o mesmo colega como gesto de retribuição.

---

**Atividade 1 (2 pontos):** Com relação às perguntas abaixo, assinale a alternativa correta. Justifique sua resposta.

**I.** O que é um repositório Git?

- a) Um arquivo de configuração do Git.
- b) Uma pasta local onde os arquivos do projeto são armazenados e gerenciados pelo Git.
- c) Um serviço online para compartilhar arquivos.
- d) Um comando usado para criar novos branches.

**II.** Qual a diferença entre commit e push em Git?

- a) Commit é usado para enviar alterações para o repositório remoto, enquanto push é usado para salvar alterações localmente.
- b) Commit é usado para salvar alterações localmente, enquanto push é usado para enviar as alterações para o repositório remoto.
- c) Ambos são usados para enviar alterações para o repositório remoto.
- d) Ambos são usados para salvar alterações localmente.

**III.** Qual comando Git é usado para alternar para uma outra branch?

- a) git checkout
- b) git branch
- c) git commit
- d) git push

**IV.** O que é um conflito de merge em Git?

- a) Um erro fatal que impede o merge de ser concluído.
- b) Uma situação em que o Git não consegue determinar automaticamente como mesclar duas versões de um arquivo.
- c) Um problema de segurança no repositório.
- d) Uma mensagem de erro exibida pelo Git.

**V.** Qual o objetivo de um pull request?

- a) Solicitar que outro colaborador do projeto remova alterações.
- b) Unir as alterações de um branch para outro.
- c) Revisar e discutir alterações antes de mesclá-las no branch principal.
- d) Excluir um branch do repositório.

**VI.** Qual comando Git é usado para revisar o histórico de commits?

- a) git log
- b) git status
- c) git diff
- d) git merge

**VII.** Qual a diferença entre git pull e git fetch?

- a) git pull traz as alterações do repositório remoto e as mescla automaticamente, enquanto git fetch apenas traz as alterações.
- b) git fetch traz as alterações do repositório remoto e as mescla automaticamente, enquanto git pull apenas traz as alterações.
- c) Ambos são usados para trazer as alterações do repositório remoto.
- d) Ambos são usados para enviar as alterações para o repositório remoto.

**VIII.** O que é o arquivo .gitignore?

- a) Um arquivo que lista os usuários autorizados a acessar o repositório.
- b) Um arquivo que armazena informações de configuração do Git.
- c) Um arquivo usado para especificar quais arquivos e diretórios devem ser ignorados pelo Git.
- d) Um arquivo que registra as alterações feitas no repositório.

**IX.** O que é um fork em um repositório GitHub?

- a) Uma atualização automática dos arquivos do repositório.
- b) Uma cópia independente de um repositório existente, permitindo que os colaboradores trabalhem em alterações sem afetar o repositório original.
- c) Uma operação de exclusão permanente de um repositório.
- d) Uma ação que remove um branch do repositório.

**Questão 2 (1 ponto):** Criar um profile README para a sua conta GitHub. Inclua no relatório um print da página e o link para o seu perfil no GitHub.

**Questão 3 (7 pontos):** Execute as atividades práticas abaixo.

**Configuração Inicial:**

- Crie um repositório no GitHub e clone-o para sua máquina local.

**Branching e Merging:**

- Crie uma nova branch chamada `feature-x`.
- Adicione um arquivo chamado `feature.txt` com algum conteúdo relevante.
- Faça commit da sua mudança e depois faça merge da branch `feature-x` para a branch principal (main ou master).
- Envie as alterações para o repositório central no GitHub.

**Simulação de Colaboração:**

- Peça a um colega para fazer um fork do seu repositório.
- Faça uma mudança significativa no arquivo `feature.txt` e realize o commit dessa mudança na sua branch principal.
- Peça ao seu colega para fazer uma mudança diferente no mesmo arquivo e submeter um pull request para você.
- Revise o pull request e faça o merge dele no seu repositório principal.