

Construção de software

Git (exercícios)

1. Qual o comando para obter a versão instalada do Git?
git --version
2. Qual o efeito da execução de cada um dos comandos abaixo?
 - a. **git config -l**
Mostra as configurações de credenciais
 - b. **git mv a.txt b.txt**
Move um determinado arquivo, no caso do exemplo, renomeia
 - c. **git reset --hard**
Reseta a branch, ignorando os arquivos locais e puxando a versão da HEAD.
 - d. **git log -27**
Mostra o log do em até 27 bytes
 - e. **git help**
Ajuda do git, mostrando a lista de comandos
 - f. **git help reset**
Mostra o manual do comando reset
 - g. **git add --all**
Adiciona todas as mudanças
 - h. **git add -u**
Adiciona arquivos modificados que já estão adicionados ao repositório
3. O fluxo “clássico” de interação com o Git é algo como “alterar um ou mais arquivos”, “acrescentar essas mudanças para serem contemplados no próximo commit” e, finalmente, executar um “commit”. Quais os comandos necessários para realizar os dois últimos “passos” desse fluxo?
git add -all; git commit;
4. Qual o comando deve ser executado para identificar o que foi alterado desde o último “commit”?
git diff
5. Em um dado repositório, arquivos simplesmente copiados para lá, ou seja, *untracked*, podem ser exibidos/identificados com que comando?
Git ls -files
6. Qual o comando para efetuar um *commit*?

git commit

7. Qual o comando que devemos empregar para descartar mudanças ocorridas no arquivo teste.txt, por exemplo?

git checkout HEAD --teste.txt

8. O que deve ser feito para que um determinado diretório do seu repositório seja ignorado pelo Git? Faça uma busca por **.gitignore**.

Adicionar ao arquivo gitignore o nome do diretório que se deseja ignorar

9. O que acontece se o seu repositório local for acidentalmente cd removido?

Pode ser restaurado da versão do repositório remoto

10. Como clonar um repositório remoto?

git clone <url-repositorio>

11. Em alguns cenários **git log** pode produzir extensos resultados. Se houver interesse em visualizar o histórico de um repositório, onde cada mudança é fornecida exatamente em uma única linha, qual o comando que deve ser empregado?

git log --pretty=oneline

12. Em qual arquivo o Git armazena informações de configuração empregadas por usuário?

No arquivo config

13. Qual o comando para criar um repositório local?

git init

14. Qual o nome do diretório criado pelo Git quando se executa o comando **git init**?

Ele cria uma pasta oculta chamada .git

15. Qual o comando para adicionar todos os arquivos modificados? (Aqueles para os quais **git status** identificam como **modified**?)

git add -u

16. O Git faz uso do valor de *hash* conhecido por SHA1. O que isto significa? Qual o propósito? O que é SHA1?

Gerar um hash único que sirva como chave para cada conjunto de mudanças

17. Qual a palavra para indicar o último *commit* em vez do valor de *hash* SHA1 correspondente?

git log -n

18. Quando se cria dois arquivos usando um editor de texto qualquer e, na sequência, executamos o comando **git add -u**, os dois arquivos criados passam de *untracked* para *new file*?

Não, apenas se forem adicionados ao repositório

19. Qual o efeito da execução dos dois comandos abaixo, nesta ordem, em um dado repositório?

```
git reset --soft HEAD~1
```

O primeiro comando realiza um reset para versão da HEAD, deixando as mudanças da árvore local, no segundo comando, ele reseta os arquivos para a HEAD ignorando qualquer mudança realizada.

20. Após o emprego de um ambiente integrado de desenvolvimento (IDE), é comum a criação de arquivos e diretórios. Qual o comando que podemos empregar para remover arquivos e diretórios *untracked*?

```
git clean -f
```

21. Qual o nome do arquivo no qual podemos inserir a indicação para o Git de arquivos e diretórios a serem ignorados?

.gitignore

22. Quando se cria o arquivo `MinhaClasse.class` em um dado diretório e desejamos que o Git ignore não apenas este, mas arquivos `.class` em geral, por todos os membros de uma equipe que estão contribuindo com um dado projeto?

Basta adicionar ao arquivo .gitignore o seguinte comando: *.class

23. jQuery é uma famosa biblioteca em JavaScript. Consulte detalhes em <http://jquery.com>. O repositório Git correspondente encontra-se disponível em <https://github.com/jquery/jquery.git>. Faça o clone deste repositório **jqueryrepo**.

24. No repositório **jqueryrepo**, criado no passo anterior, qual o efeito do comando **git shortlog -sne**?

Mostra os logs de commits com usuários e e-mails

25. No repositório **jqueryrepo**, qual o efeito de **git remote -v**?

Mostra o caminho do repositório remoto do jqueryrepo

26. Um repositório Git pode ser etiquetado ao longo do tempo. Ou seja, *commits* específicos podem ser “marcados” ou “etiquetados” para facilitar referências posteriores. Para listar todas as “etiquetas” (*tags*) estabelecidas para um dado repositório, qual comando deve ser executado?

```
git tag <tag>
```

27. Caso um dato repositório retorne muitas “marcas” ou “etiquetas” para o comando **git tag**, como retornar apenas aquelas que atendem a determinado padrão, por exemplo, iniciadas por 2.0?

```
git tag -l 2.0
```

28. Qual o efeito do comando **git tag -a 3.4-gold -m "minha versão ouro"**?

Vai adicionar a anotação minha versão ouro na tag 3.4-gold

29. Após executado o comando acima, qual o efeito de **git show 3.4-gold**?

Serão exibidas as mensagens da tab 3.4-gold

30. O que o comando **git push origin 3.4-gold** teria como efeito?

Realizaria o envio para a branch remota

31. Após executar um *commit*, qual o efeito de **git commit –amend**?

Realiza um commit levando em consideração a versão mais atual daquela branch.

32. Após executar **git add x.txt**, qual o efeito de **git reset HEAD x.txt**?

O arquivo será sinalizado como untracked

33. Após alterar o conteúdo de um arquivo *committed* em passo anterior, qual o efeito do comando **git checkout – a.txt**?

Ele realizar o checkout do arquivo com o que está no repositório.

34. Qual a diferença entre os comandos **git reset HEAD x.txt** e **git checkout – a.txt**?

Os dois comandos terão os mesmo resultados.