



7/7

De volta à lição

Pergunta 1

Através de qual comando adicionamos uma modificação ao stage do Git?

```
git add meu_arquivo
```

- ! Para adicionar uma modificação ao stage do Git, ou seja, prepará-la para ser commitada, é necessário utilizar o comando "git add". A opção correta, "git add meu_arquivo", está de acordo com a sintaxe correta para adicionar um arquivo específico ao stage.

Agora, vamos analisar as demais opções:

git -a meu_arquivo: Essa opção está incorreta porque o parâmetro "-a" no comando "git" é usado para adicionar todas as modificações existentes ao stage, não apenas



git add --to-stage meu_arquivo: Essa opção está incorreta porque não existe um parâmetro "--to-stage" no comando "git add". O parâmetro correto para adicionar um arquivo ao stage é apenas o nome do arquivo, como mostrado na opção C.

git insert meu_arquivo: Essa opção está incorreta porque o comando "git insert" não existe no Git. Não é possível adicionar uma modificação ao stage usando esse comando."

Pergunta 2

Qual é o comando correto para salvar uma alteração que está no stage?

```
git commit -m "mensagem da mudança"
```

- ! No Git, o comando utilizado para salvar uma alteração que está no stage e criar um novo commit é o comando `"git commit"`.

Para adicionar uma mensagem ao commit, utiliza-se o parâmetro `"-m"` seguido da mensagem entre aspas. A opção correta, `"git commit -m "mensagem da mudança"`, está de acordo com a sintaxe correta para realizar um commit com uma mensagem descritiva da alteração.

Agora, vamos analisar as demais opções:

`git save`: Essa opção está incorreta porque o comando `"git save"` não existe no Git. O comando correto para salvar uma alteração no stage e criar um novo commit é `"git commit"`.

`git save -m "mensagem da mudança"`: Essa opção está incorreta porque, novamente, o comando `"git save"` não existe no Git. O correto é utilizar o comando `"git commit"` para salvar a alteração no stage. Além disso, o parâmetro correto para adicionar uma mensagem ao commit é `"-m"`, não `"-m"`.

`git commit --save "mensagem da mudança"`: Essa opção está incorreta porque o parâmetro `"--save"` não faz parte do comando

Pergunta 3

Qual comando usamos para solicitar o log dos últimos 3 commits?

```
git log -n 3
```

- ! No Git, o comando utilizado para visualizar o log dos commits é o comando `"git log"`. Para especificar o número de commits a serem exibidos, utiliza-se o parâmetro `"-n"` seguido do número desejado. A opção correta, C) `git log -n 3`, está de acordo com a sintaxe correta para solicitar o log dos últimos 3 commits.

Agora, vamos analisar as opções incorretas e por que elas estão erradas:

`git log --count=3`: Essa opção está incorreta porque o parâmetro "`--count`" não é utilizado no comando "`git log`". Para especificar o número de commits, deve-se utilizar o parâmetro "`-n`" seguido do número desejado, como mostrado na opção correta.

`git log --max=3`: Essa opção está incorreta porque o parâmetro "`--max`" não é utilizado no comando "`git log`" para especificar o número de commits a serem exibidos. O parâmetro correto é "`-n`", como mostrado na opção correta.

`git log top 3`: Essa opção está incorreta porque o comando "`git log`" não possui um subcomando "`top`". Além disso, não é possível especificar o número de commits diretamente após o comando "`git log`". O parâmetro correto para especificar o número de commits é "`-n`", como mostrado na opção correta.

Pergunta 4

Com qual comando alteramos o estado do repositório para o estado de determinado commit?

`git checkout commit_hash`

- ! No Git, o comando utilizado para alterar o estado do repositório para o estado de um determinado commit é o comando "`git checkout`". É necessário fornecer o hash do commit desejado para especificar qual commit será utilizado. A opção correta, D) `git checkout commit_hash`, está de acordo com a sintaxe correta para alterar o estado do repositório para um commit específico.

Agora, vamos analisar as opções incorretas e por que elas estão erradas:

`git change commit_hash`: Essa opção está incorreta porque o comando "`git change`" não existe no Git. Não há um comando específico para "`change`" no Git. O comando correto para alterar o estado do repositório para um commit específico é "`git checkout`", como mostrado na opção correta.

`git rollback commit_hash`: Essa opção está incorreta porque o comando "`git rollback`" não existe no Git. O comando correto para alterar o estado do repositório para um commit específico é "`git checkout`".

`git move commit_hash`: Essa opção está incorreta porque o comando "`git move`" não é utilizado para alterar o estado do repositório para um commit específico. O comando correto para essa finalidade é "`git checkout`".

Pergunta 5

Qual comando utilizamos para deletar uma branch?

```
git branch -D minha_branch
```

Pergunta 6

Qual comando utilizamos para trazer o conteúdo de uma branch para a branch atual?

```
git merge branch_de_origem
```

- ! No Git, o comando utilizado para trazer o conteúdo de uma branch para a branch atual é o comando "git merge". O "merge" combina as alterações da branch de origem com a branch atual, aplicando as modificações necessárias. A opção correta, D) git merge branch_de_origem, está de acordo com a sintaxe correta para realizar a mesclagem de uma branch na branch atual.

Agora, vamos analisar as opções incorretas e por que elas estão erradas:

git union branch_de_origem: Essa opção está incorreta porque o comando "git union" não existe no Git. Não há um comando específico chamado "union" para trazer o conteúdo de uma branch para a branch atual. O comando correto é "git merge", como mostrado na opção correta.

git fusion branch_de_origem: Essa opção está incorreta porque o comando "git fusion" não é um comando válido no Git. Não há um comando chamado "fusion" para trazer o conteúdo de uma branch para a branch atual. O comando correto é "git merge".

git merge branch_de_origem branch_atual: Essa opção está incorreta porque a sintaxe não está correta. No comando "git merge", a branch de origem é especificada como argumento, e não é necessário especificar a branch atual separadamente. A opção correta, D) git merge branch_de_origem, utiliza a sintaxe correta para realizar a mesclagem.

Pergunta 7

Através de qual comando atualizamos o repositório do Github com o conteúdo local?

git push

- ! No Git, o comando utilizado para atualizar o repositório do GitHub com o conteúdo local é o comando "git push". Esse comando envia as alterações locais para o repositório remoto no GitHub. A opção correta, A) git push, está de acordo com a sintaxe correta para realizar essa operação.

Agora, vamos analisar as opções incorretas e por que elas estão erradas:

git upload: Essa opção está incorreta porque o comando "git upload" não existe no Git. Não há um comando específico chamado "upload" para atualizar o repositório do GitHub com o conteúdo local. O comando correto é "git push".

git update: Essa opção está incorreta porque o comando "git update" não é utilizado para atualizar o repositório do GitHub com o conteúdo local. O comando "git update" é usado para atualizar as referências locais com commits remotos, mas não para enviar alterações para o repositório remoto. A opção correta é "git push".

git pull: Essa opção está incorreta porque o comando "git pull" é utilizado para obter as alterações mais recentes do repositório remoto e mesclá-las com o conteúdo local. Ele não é usado para atualizar o repositório do GitHub com o conteúdo local. A opção correta para enviar as alterações locais para o repositório remoto é "git push".

De volta à lição



WhatsApp

+55 (11) 4200-2991

Telefone

+55 (11) 3030-3200

Politica de privacidade



This site is protected by reCAPTCHA and the Google Privacy Policy and Terms of Service apply.