

O que aprendemos?

Nesta aula, aprendemos:

- Que o Git pode nos ajudar a desfazer alterações que não vamos utilizar;
- Que, para desfazer uma alteração antes de adicioná-la para `commit` (com `git add`), podemos utilizar o comando `git checkout -- <arquivos>`;
- Que, para desfazer uma alteração após adicioná-la para `commit`, antes precisamos executar o `git reset HEAD` e depois podemos desfazê-las com `git checkout --``;
- Que, para revertermos as alterações realizadas em um `commit`, o comando `git revert` pode ser a solução;
- Que o comando `git revert` gera um novo `commit` informando que alterações foram desfeitas;
- Que, para guardar um trabalho para retomá-lo posteriormente, podemos utilizar o `git stash`;
- Que, para visualizar quais alterações estão na `stash`, podemos utilizar o comando `git stash list`;
- Que, com o comando `git stash apply <numero>`, podemos aplicar uma alteração específica da `stash`;
- Que o comando `git stash drop <numero>` remove determinado item da `stash`;
- Que o comando `git stash pop` aplica e remove a última alteração que foi adicionada na `stash`;
- Que o `git checkout` serve para deixar a cópia do código da nossa aplicação no estado que desejarmos:
 - `git checkout <branch>` deixa o código no estado de uma `branch` com o nome `<branch>`;
 - `git checkout <hash>` deixa o código no estado do *commit* com o hash `<hash>`.