



100%

Transcrição

Já criamos nosso primeiro repositório, então, se executarmos `git status` dentro da pasta em que trabalhamos anteriormente, veremos que trata-se de um repositório Git, porém, seu arquivo ainda não está sendo monitorado, ou seja, ele não está salvo no histórico do Git. Para salvarmos uma alteração, ou um arquivo nele, precisaremos que ele monitore o arquivo, e suas mudanças.

Como o arquivo `index.html` ainda não está sendo monitorado, e nunca foi editado e salvo pelo Git, utilizaremos o comando `git add index.html`. Se tivéssemos vários arquivos, não precisaríamos colocar seus nomes um a um, bastando `git add .` para que todos os arquivos da pasta atual sejam monitorados.

Com isso, se rodarmos `git status`, desta vez teremos um retorno diferente, incluindo `Changes to be committed`, isto é, "mudanças a serem commitadas", ou salvas, enviadas. Inclusive, é indicado que poderíamos executar `git rm` para remover o arquivo e para que o mesmo deixe de ser monitorado, o que não queremos fazer.

Queremos salvar as alterações, e o que poderemos entender como sendo um *check point* para indicar que houve mudança, seria o **commit**, que precisa ter modificações, que já adicionamos, mas também precisa ter uma mensagem, o que criaremos agora. Por já termos adicionado as modificações a serem enviadas, executaremos simplesmente `git commit -m "Criando arquivo index.html com lista de cursos"`, em que o parâmetro `-m` serve para passarmos uma mensagem de commit, que será incluído entre aspas.

A boa prática pede para colocarmos mensagens **descritivas**, evitando que fiquem muito grandes.

Quando dermos "Enter", o Git Bash nos informa que este é o `root-commit`, commit base dentro de um master, e exibe a mensagem que configuramos. Também é mostrado quais foram as alterações, no caso, apenas 1, com 15 inserções (linhas). Se executarmos `git status` novamente, teremos que não há nada a ser commitado, entretanto ele não mostra mais que não há commits ainda.

Vamos fazer uma modificação simples, como colocar um acento agudo em "Contínua" de `Integração Contínua`. Salvaremos e reexecutaremos `git status`, e obteremos a indicação de que há uma modificação não salva. Para isso, executaremos `git add index.html`. Com outro `git status`, teremos a mensagem de que há mudanças a serem commitadas. Usaremos `clear` para limparmos a tela, e então `git commit -m "Acento adicionado no curso de Integração Contínua"` e pressionaremos "Enter".

Conseguiremos acessar uma espécie de lista de commits realizados de forma muito simples, por meio de um comando que veremos a seguir.

304.0k xp





100%

ATIVIDADES
1 DE 9

FÓRUM DO
CURSO

VOLTAR
PARA
DASHBOARD

MODO
NOTURNO

ABRIR
CADERNO



304.0k xp

a