

TUTORIAL: ESTRUTURA DE DADOS - PILHAS

MÓDULO 2

Olá, Caro aluno!

Agora nós vamos aprender sobre o conceito de estrutura de dados **Pilhas**. Então, diferente das listas, a pilha tem uma complexidade de que a ideia é que o último que entra é o primeiro a sair. Então vamos pensar no conceito do mundo real, por exemplo, nós podemos pensar em **Pilhas** de tijolos, **Pilhas** de livros, **Pilhas** de pratos, como está aqui.

Então, por exemplo, pensar aqui no tijolo. Então, se eu coloco um tijolo, inicio uma pilha, então não tinha nada, então a pilha estava zerada, coloco um tijolo, ele é o último a entrar, mas se eu quiser tirar esse tijolo, a pilha vai ficar vazia, perfeito. E se eu coloco mais um tijolo em cima?

Coloco um tijolo em cima. Se o último que entrou foi o tijolo 2, certo? Quem vai ser o primeiro a sair? O tijolo 2. Ah, mas Marlon, se eu quiser tirar o tijolo 1 de lado, eu tiro. Mas pensa o seguinte, você está numa situação de pilha estruturada, que você não pode, você está num buraco, era um buraco e você foi colocando tijolo, então você não tem como tirar o 1 de lado, você colocou 1 lá por baixo, você não tem mais como tirar.

Então, para você tirar o 1, você vai ter que tirar o 2 primeiro, então, essa é a ideia de pilha, você não consegue tratar o que está abaixo de você. Então, perfeito. Então, por exemplo, temos aqui a pilha 1, 2, 3. O 3 foi o último a entrar.

Para que eu possa manipular o 2 eu tenho que remover o 3. Então, o último a entrar é sempre o primeiro a sair. Então, vamos testar isso aqui no código. Existe como nós criarmos toda essa funcionalidade de pilha, criando algumas funções, mas, como nós temos essa biblioteca de Queue, dentro da collections nós vamos utilizá-la.

Então, from collections importar a funcionalidade de Queue. Essa é a primeira situação, está o import, mas o que quer dizer esse import? Que você está importando alguma situação, uma biblioteca, algum módulo que você vai utilizar. E, assim, dentro desse código, você pode utilizar os métodos que temos dentro desse objeto importado. Ok? Então, inicialmente, nós vamos instanciar a pilha.

Como assim? Nós vamos fazer o quê? Como eu citei no outro vídeo de listas, a variável recebe o tipo a partir do momento que você inseriu o valor. Então, se eu colocar pilha recebe 1, ele vai fazer o quê? Número literal 1. No caso, se a pilha recebe de Queue, ele vai fazer o quê? Vai ser um tipo de Queue, que vai ser pilha. Ok? Então, vamos lá. Temos uma pilha, se eu imprimir essa pilha, vai aparecer o quê? Vai aparecer vazio.

Então o que nós vamos fazer? Nós vamos inserir objetos dentro dessa pilha. essa pilha vai receber objetos. Então inserimos uma String Prato Rosa.

Ok? Então se, no caso, eu pegar esse daqui, vamos fazer isso aqui. Vamos imprimir o que nós temos

por último na nossa pilha. Então, por exemplo, pilha, print, topo da pilha, isso aqui é uma String, simplesmente é um texto, vírgula o quê? Vírgula qual objeto você quer mostrar.

Então, pilha, -1, quer dizer o quê? O -1 é uma função desse de Queue, e o que ela mostra? O último objeto da pilha. Então, sempre que você quiser utilizar esse módulo de Queue, da Biblioteca Collections, e quiser saber qual é o último da pilha, você vai colocar o nome da pilha que você criou, nesse caso é pilha realmente. Qual que é a posição? -1. Ok? Então, vamos ver quem que é o último da pilha.

Vou executar Prato Rosa. Olha lá. Ok? Beleza.

Então, agora, eu vou fazer o quê? Eu vou adicionar mais dois pratos na pilha. Então, adicionei Prato Rosa, que será lá embaixo da pilha, o zero da pilha, o Prato Branco, que será em cima do Prato Rosa, e o Prato Azul, que será em cima do Prato Branco. Então, utilizei para adicionar o Append também, Então, pilha.append, qual que é o objeto.

Qual que será agora? Então, print topo da pilha, pilha -1. Pilha -1 é o último item da pilha. O último quer dizer o quê? O último a entrar, não lá embaixo. É o último a entrar.

Ok? Então, vamos dar um F5. Quem que ele deve imprimir? Prato Azul. Ok.

Então, vocês se perguntam.

Marlon, mas se eu quiser saber qual é o último item lá na pilha? O que está lá embaixo. Então, você sabe qual é o que está lá embaixo, o índice, pelo menos, lá embaixo. Vai ser qual índice? No mesmo conceito de listas, vai ser 0. Então, qual que é o primeiro que entrou lá embaixo? Foi o prato rosa.

Então, quem tem que imprimir aqui? Ele imprimiu o prato azul, porque é esse print que mostra qual é o último que entrou. E imprimiu o pilha 0, que é o prato rosa, que foi o primeiro que entrou, ok? Perfeito. E agora, o que nós vamos fazer? Nós vamos fazer o seguinte.

Da mesma forma que existe a função append, que adiciona mais um item na lista, é possível desempilhar, né? Então, para fazer esse desempilhamento, me perdoem se eu estiver errando na gramática, mas a hora que eu estiver necessitando tirar esse último item que entrou, eu utilizo o pop. Então, pilha.pop. Preciso setar, preciso definir quem que ele vai retirar? Não, então, o pop não vai ter nenhum parâmetro aqui dentro, porque o pop você não tem na pilha, ou você tira o último que entrou ou você não tira ninguém, você não pode mexer no meio da pilha. Existe essa imutabilidade na situação.

Então, nesse caso, eu adicionei por primeiro rosa, por segundo branco, por terceiro azul, certo? Então, rosa é índice 0, branco índice 1, azul índice 2. Eu não tenho como tirar o branco se eu não tirar o azul primeiro. Então, a partir do momento que eu coloco pilha.pop, eu vou ter o quê? Topo da pilha após desempilhar. Se eu estou tirando o azul, quem ele vai ter de resultado? Ele mostrou aqui primeiro o topo da pilha antes de fazer o comando pop e depois o topo da pilha após desempilhar.

Então prato branco, ok? Espero que você tenha conseguido. Sugiro testar e colocar outros parâmetros, testem com livros, com tijolos, com mais parâmetros, testem funções de colocar índice e fazer as impressões. Valeu?