

ESTRUTURA DE DADOS: PILHAS

MÓDULO 2

Agora, você terá a possibilidade de testar isso direto no VSCode, como foi feito anteriormente ao estudarmos Filas. Para isso, precisamos utilizar uma classe da biblioteca Collections de python. Essa classe se chama **deque** e vamos criar uma pilha. Siga o passo a passo:

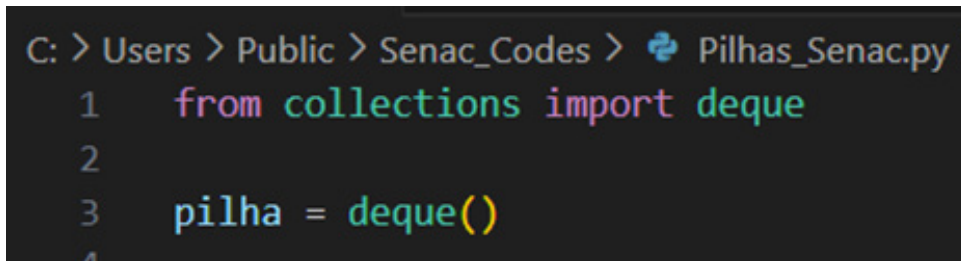
1. Crie um outro arquivo no VSCode, ele pode se chamar Pilhas.py (py é a extensão de arquivo python. Sua interface de desenvolvimento, no caso VSCode, irá entender que se trata de um arquivo escrito com características da linguagem Python). Para isso, precisaremos que inicialmente você insira a linha de código:

- from collections import deque

Importante: Essa linha fará com que você utilize uma biblioteca do Python, a fim de que não seja necessário criarmos métodos básicos para execução da pilha.

2. Logo após é necessário criarmos uma pilha em memória. Ou seja, declararmos que “pilha” é realmente uma pilha em sua essência. Ou seja:

- pilha = deque()



```
C: > Users > Public > Senac_Codes > Pilhas_Senac.py
1  from collections import deque
2
3  pilha = deque()
4
```

Agora você utilizará um conceito da vida real. Como? Pilha de **pratos**. Pense que você tem 3 pratos.

3. O primeiro que você inserirá será de cor rosa, o segundo branco e o último azul – um de cada vez, ok? Então continue o código. Insira um prato da cor rosa, utilizando a função **append()**.

- pilha.append("Prato Rosa")

4. <4> A linha de código acima insere um Prato Rosa na nossa pilha assim, realizamos o print (a impressão) do resultado na tela:

- print("Topo da pilha:", pilha[-1])

```
C: > Users > Public > Senac_Codes > Pilhas_Senac.py > ...
1  from collections import deque
2
3  pilha = deque()
4
5  pilha.append("Prato Rosa")
6  print("Topo da pilha:", pilha[-1])
```

5. Quem você terá no topo da pilha? Execute o código (F5):

```
Topo da pilha: Prato Rosa
PS C:\Users\Public\Senac_Codes> 
```

Importante: Utiliza-se o índice -1 para sabermos qual o item que está no topo da pilha!

A pilha tem um prato, logo esse prato inserido se torna o topo.

6. Insira mais um prato, agora da cor branca:

- pilha.append("Prato Branco")
- print("Topo da pilha:", pilha[-1])

```
C: > Users > Public > Senac_Codes > Pilhas_Senac.py > ...
1  from collections import deque
2
3  pilha = deque()
4
5  pilha.append("Prato Rosa")
6  print("Topo da pilha:", pilha[-1])
7
8  pilha.append("Prato Branco")
9  print("Topo da pilha:", pilha[-1])
10
```

7. Execute.

Temos dois pratos, um rosa e outro branco. O topo será o branco. Observe a linha abaixo:

```
print("Topo da pilha:", pilha[-1])
```

Percebeu que ela aparece duas vezes? Ou seja, sempre que você executar essa linha agora imprime

mirá o Prato Branco. É importante que você tenha em mente que isso NÃO acontece pelo fato dele estar embaixo da linha: `pilha.append("Prato Branco")`

8. Agora vamos inserir mais um prato, da cor azul:

- `pilha.append("Prato Azul")`

```
C: > Users > Public > Senac_Codes > Pilhas_Senac.py > ...
1  from collections import deque
2
3  pilha = deque()
4
5  pilha.append("Prato Rosa")
6
7  pilha.append("Prato Branco")
8
9  pilha.append("Prato Azul")
10 print("Topo da pilha:", pilha[-1])
```

9. Executando será impresso:

Topo da pilha: Prato Azul

Qual o motivo? Você inseriu o rosa, logo acima dele o branco e logo acima o azul.

E caso você queira tirar o prato azul de cima de pilha? Digite a linha de código a seguir e execute o código (F5).

- `pilha.pop()`

- `print("Topo da pilha após desempilhar:", pilha[-1])`

```
C: > Users > Public > Senac_Codes > Pilhas_Senac.py > ...
1  from collections import deque
2
3  pilha = deque()
4
5  pilha.append("Prato Rosa")
6
7  pilha.append("Prato Branco")
8
9  pilha.append("Prato Azul")
10 print("Topo da pilha:", pilha[-1])
11
12 pilha.pop()
13
14 print("Topo da pilha após desempilhar:", pilha[-1])
```

10. Após executar você obterá o seguinte resultado:

```
Topo da pilha: Prato Azul  
Topo da pilha após desempilhar: Prato Branco
```

Explicando melhor o que aconteceu:

- foi inserido o prato rosa;
- logo após foi inserido o prato branco;
- por fim foi inserido o prato azul;
- após o prato azul ter sido empilhado, ele se tornou o último da fila;
- executando a função `pop()`, retiramos o último prato da pilha (azul);
- sendo assim, o prato branco voltou a se tornar o topo da pilha.