

Exceções

Lista de Exercícios II

- 1) Implemente uma função `definir_idade(idade)` que receba um número inteiro representando a idade de uma pessoa. Se a idade for negativa, a função deve **lançar** uma exceção `ValueError` com a mensagem `"Idade não pode ser negativa"`. Caso contrário, apenas retorne a idade recebida.

```
# Exemplo rápido de uso:
try:
    print(definir_idade(20))    # 20
    print(definir_idade(-3))   # lança ValueError
except ValueError as e:
    print('Erro:', e)
```

- 2) Crie uma função `dividir(a, b)` que devolva o resultado da divisão `a / b`. Caso `b` seja zero, a função deve **lançar** a exceção `ZeroDivisionError` com a mensagem `"Denominador não pode ser zero"`.

```
# Exemplo rápido de uso:
try:
    print(dividir(10, 2))    # 5.0
    print(dividir(7, 0))     # lança ZeroDivisionError
except ZeroDivisionError as e:
    print('Erro:', e)
```

- 3) Escreva uma função `atribuir_nota(nota)` que receba um valor de 0 a 10. Se a nota estiver fora desse intervalo, a função deve **lançar** um `ValueError` com a mensagem `"Nota deve estar entre 0 e 10"`. Caso contrário, retorne a nota.

```
# Exemplo rápido de uso:
try:
    print(atribuir_nota(8.5))    # 8.5
    print(atribuir_nota(11.0))  # lança ValueError
except ValueError as e:
    print('Erro:', e)
```

- 4) Implemente duas exceções personalizadas: `SaldoInsuficienteError` e `ValorInvalidoError`, ambas herdando de `Exception`. Em seguida, crie a função `sacar(saldo, valor)` que recebe o saldo atual de uma conta e o valor de um saque. A função deve retornar o saldo atualizado quando o saque for válido. Se o valor do saque for **menor que zero**, a função deve **lançar** `ValorInvalidoError` com a mensagem "Valor do saque não pode ser negativo". Se o valor do saque for **maior que o saldo disponível**, a função deve **lançar** `SaldoInsuficienteError` com a mensagem "Saldo insuficiente para realizar o saque".

```
# Exemplo rápido de uso:
try:
    print(sacar(100.0, 40.0))    # 60.0
    print(sacar(50.0, 80.0))    # levanta SaldoInsuficienteError
    print(sacar(100.0, -5.0))   # levanta ValorInvalidoError
except SaldoInsuficienteError as e:
    print('Erro de saldo:', e)
except ValorInvalidoError as e:
    print('Erro de entrada:', e)
```

- 5) Implemente uma exceção personalizada `FilaVaziaError` (herdando de `Exception`) e duas funções: `entrar(fila, nome)`, que adiciona um nome ao final da lista `fila` validando que o nome seja não vazio; e `chamar(fila)`, que remove e retorna o primeiro nome da fila, **lançando** `FilaVaziaError` quando a fila estiver vazia.

```
# Exemplo rápido de uso:
try:
    fila = []
    entrar(fila, 'Ana')
    entrar(fila, 'Bruno')
    print(chamar(fila)) # Ana
    print(chamar(fila)) # Bruno
    print(chamar(fila)) # lança FilaVaziaError
except FilaVaziaError as e:
    print('Erro de fila:', e)
except ValueError as e:
    print('Erro de entrada:', e)
```