

TECNOLOGIA EM SIST. COMPUTAÇÃO

Glauber de Souza Faria

17213050160

Angra dos Reis - RJ

PROGRAMAÇÃO COM INTERF. GRÁFICAS

API-X

## QUESTÃO 1:

```
class C(object):
```

```
    def __init__(self, a, b):
```

```
        self.a = a
```

```
        self.b = b
```

```
    def __add__(self, c):
```

```
        return C(c.b[::-1] + self.a[::-1], self.b + c.b)
```

```
    def __repr__(self):
```

```
        return self.a + self.b
```

```
print(C("ABC", "DEF") + C("FOO", "BAR"))
```

## QUESTÃO 2:

```
Def g(A, b):
```

```
    IF LEN(A) < b:
```

```
        RETURN [A]
```

```
    RETURN [A[:b]] + g(A[b:LEN(A)], b)
```

```
PRINT(g(List(Range(10)), 3))
```

```
PRINT(g("*" * 10, 4))
```

### QUESTÃO 3:

```
from operator import itemgetter
```

```
class Agencia(object):
```

```
    def __init__(self):
```

```
        self.Agencia = []
```

```
    def cadastradas(self):
```

```
        Contas-Ordenadas = sorted(self.Agencia, key=itemgetter('NConta'))
```

```
        return (Contas-Ordenadas)
```

```
    def saldo(self, NConta):
```

```
        for c in self.Agencia:
```

```
            if NConta == c['NConta']:
```

```
                return (c['saldo'])
```

```
    def limite(self, NConta):
```

```
        for c in self.Agencia:
```

```
            if NConta == c['NConta']:
```

```
                return (c['limite'])
```

```
    def AlteraLimite(self, NConta, VLimite):
```

```
        for c in self.Agencia:
```

```
            if NConta == c['NConta']:
```

```
                c['limite'] = VLimite
```

```
DEF MOVIMENTA (SELF, NCONTA, VALOR):
```

```
IF SELF.CRIACONTA (NCONTA) == False:
```

```
    IF VALOR > 0:
```

```
        SELF.DEPOSITO (NCONTA, VALOR)
```

```
    ELIF VALOR < 0:
```

```
        SELF.SAQUE (NCONTA, VALOR)
```

```
    ELSE:
```

```
        PRINT ("NÃO É POSSÍVEL REALIZAR ESTA OPERAÇÃO")
```

```
ELSE:
```

```
    IF VALOR > 0:
```

```
        CONTA = {"NCONTA": NCONTA, "LIMITE": 0, "EXTRATO": [VALOR],  
                  "SALDO": VALOR}
```

```
        SELF.AGENCIA.APPEND (CONTA)
```

```
    ELSE:
```

```
        PRINT ("CONTA NÃO PODE SER CRIADA COM MOV NEGATIVA")
```

```
DEF SAQUE (SELF, NCONTA, VALOR):
```

```
    FOR C IN SELF.AGENCIA:
```

```
        IF NCONTA == C["NCONTA"]:
```

```
            IF VALOR <= C["LIMITE"]:
```

```
                C["SALDO"] = C["SALDO"] - VALOR
```

```
                C["EXTRATO"].APPEND (VALOR)
```

```
                BREAK
```

```
            ELSE:
```

```
                PRINT ("CONTA SEM SALDO SUFICIENTE p SAQUE")
```

```
Def deposito (self, NConta, valor):
```

```
    For c in self. Agencia:
```

```
        IF NConta == c["NConta"]:
```

```
            c["Saldo"] = c["Saldo"] + valor
```

```
            c["EXtrato"]. Append(valor)
```

```
        Break
```

```
Def criaConta (self, NConta):
```

```
    criaConta = ""
```

```
    For c in self. Agencia:
```

```
        IF NConta == c["NConta"]:
```

```
            criaConta = False
```

```
        Break
```

```
    Return criaConta
```

```
Def extrato (self, NConta):
```

```
    For c in self. Agencia:
```

```
        IF NConta == c["NConta"]:
```

```
            Return (c["EXtrato"])
```

```
Def __str__(self):
```

```
    STR = ""
```

```
    For c in self. CadasTradas():
```

```
        STR += "Conta %0d, Limite = %0.2f" % (c["NConta"],
```

```
            self.Limite(c["NConta"]))
```

```
        STR += ", Extrato = %0s, Saldo = %0.2f \n" % (
```

```
            self.Extrato(c["NConta"]), self.Saldo(c["NConta"]))
```

```
    Return STR
```

```
A = Agencia( )  
A.movimenta(5, 100)  
A.movimenta(7, 200)  
A.movimenta(5, -50)  
A.movimenta(2, 400)  
PRINT(A)
```

```
TRY:  
    A.movimenta(7, -300)  
EXCEPT VALUE ERROR AS V:  
    PRINT(V)
```

```
TRY:  
    A.movimenta(8, -50)  
EXCEPT VALUE ERROR AS V:  
    PRINT(V)
```

```
A.alteralimite(7, -400)  
A.movimenta(7, -300)  
PRINT(A)
```



# OBSERVAÇÕES:

# Importei o ITEMGETTER PARA SETAR A KEY do Dicionário

# FUNÇÃO MOVIMENTA Foi dividida em SUBFUNÇÕES PARA MANTER AS BOAS PRÁTICAS de PROGRAMAÇÃO e TORNAR A MANUTENÇÃO FUTURA MAIS FÁCIL.