

# Universidade Federal do Rio de Janeiro

## Centro de Ciências da Saúde

### Instituto de Biofísica Carlos Chagas Filho

**Disciplina:** CFB017 - Programação para Biociências

**Professor:** Dr. Vitor Lima Coelho

## TAC 5 - Programação Básica

Listas e dicionários

Funções

1 - Escreva um programa chamado “controleAquisicaoBioMol.py” que leia uma relação de materiais de laboratório de biologia molecular digitada pelo usuário que serão consultados para aquisição.

- a) O programa vai ler produto, preço e quantidade até que o usuário digite “-1” para parar.
- b) O programa armazenará em um dicionário onde o produto será a chave referente à uma lista contendo o preço e quantidade deste produto, ou seja, a chave será “produto” e o valor será uma lista [preço, quantidade]. Exemplo:

estoque = {“caixa de ponteira” : [20, 3], “caixa de par de luvas”: [35.5, 4], “máscara” : [2, 200], “pipeta” : [1500, 3]}

- c) Escreva um módulo em Python chamado “operacoesCompra.py” que conterá as funções:
  - i) imprimeProdutos: imprime o nome dos produtos a serem comprados;
  - ii) imprimeQuantidades: imprime a quantidade de produtos a serem comprados;
  - iii) calculaTotalCompra: retorna o valor total das compras;
- d) O seu programa principal “controleAquisicaoBioMol.py” deverá importar “operacoesCompra.py” para mostrar ao usuário o nome dos produtos, a quantidade e o valor total de compras.



Código 1: **ControleAquisiçãoBioMol.py**:

```
# importação de módulos criados:
# é necessário adicionar a localização deste módulo ao path do IDE!!
import OperaçõesCompra
#flag
f = 1
# criação de dicionário:
estoque = {}
# lista:
gasto = []
#loop:
while f==1:
    produto=str(input("produto:"))
    if produto == str(-1):
        break
    else:
# gravando input do usuário em variáveis:
        preço=float(input("preço:"))
        quant=int(input("quantidade:"))
# gravando elemento de dicionário:
        e = {produto : [preço,str(quant)]}
# atualizando o dicionário principal com os elementos acima:
        estoque.update(e)
# a lista gasto terá o valor total de cada item (preço x quantidade):
        gasto.append(float(preço*quant))
# utilização das funções criadas no módulo importado:
OperaçõesCompra.ImprimeProdutos(estoque)
OperaçõesCompra.ImprimeQuantidade(estoque)
print("Gastos Totais:",OperaçõesCompra.CalculaTotalCompra(gasto),sep=" ")
```



## Código 2: OperaçõesCompra.py

```
# Este módulo será importado no código principal ControleAquisiçãoBioMol.py

# recebe como entrada o dicionário criado com itens do estoque mais a lista
# como valor
def ImprimeProdutos(dic):
# formatando para sair todos os nomes numa mesma linha sem 'dic.keys'
# aparecer no output
    print(', '.join("{}".format(k) for k, v in dic.items()))

# recebe de entrada o dicionário estoque
def ImprimeQuantidade(dic):
# criação de lista vazia pra receber as quantidades - segundo item da lista
# em dic.values
    l=[]
    for v in dic.values():
        l.append(v[-1])
# só consegui imprimir tudo na mesma linha realizando a formatação e
# realizando o append em uma nova lista
    print(', '.join(map(str, l)))

# recebe como entrada uma lista chamada gastos.
# linhas 23 e 24 do código principal!!!
def CalculaTotalCompra(lista):
# realizando somatório de todos os itens da lista:
    return sum(lista)
```

Output:

```
In [30]: runfile('/home/carol/Documents/CFB017/TAC 5/TAC5_ControlAquisiçãoBioMol.py', wdir='/home/carol/Documents/CFB017/TAC 5')
Reloaded modules: OperaçõesCompra

produto:luvas
preço:25.50
quantidade:3
produto:jalecos
preço:57.56
quantidade:5
produto:pipetas p1000
preço:250.0
quantidade:2
produto:ponteiras p1000
preço:60.15
quantidade:50
produto:-1
luvas, jalecos, pipetas p1000, ponteiras p1000
3, 5, 2, 50
Gastos Totais: 3871.8
```