

# QUESTÃO 1 de 4 - Conteúdo até Aula 03

Enunciado: Imagina-se que você é um dos programadores responsáveis pela construção de app de vendas para uma determinada empresa X que aceita cartões de crédito. Uma das estratégias de vendas dessa empresa X é cobrar um Juros maior conforme a quantidade de parcelas que o cliente desejar, conforme a **listagem abaixo**:

- Se a quantidade de parcelas for **menor** que **4**, o Juros será de **0%** (0 / 100);
- Se a quantidade de parcelas for **igual ou maior** que **4** e **menor que 6**, o Juros será de **4%** (4 / 100);
- Se a quantidade de parcelas for **igual ou maior** que **6** e **menor que 9**, o Juros será de **8%** (8 / 100);
- Se a quantidade de parcelas for **igual ou maior** que **9** e **menor que 13**, o Juros será de **16%** (16 / 100);
- Se a quantidade de parcelas for **igual ou maior** que **13**, o Juros será de **32%** (32 / 100);

O valor da parcela é calculado da seguinte maneira:

$$\text{valorDaParcela} = \frac{\text{valorDoPedido} * (1 + \text{juros})}{\text{quantidadeParcelas}}$$

O valor total parcelado é calculado da seguinte maneira:

$$\text{valorTotalParcelado} = \text{valorDaParcela} * \text{quantidadeParcelas}$$

Elabore um programa em Python que:

- A. Deve-se implementar o **print** com o seu **nome completo** (somente print, não usar input aqui).  
Por exemplo: `print("Bem-vindos a loja do Bruno Kostiuk")` [EXIGÊNCIA DE CÓDIGO 1 de 6];
- B. Deve-se implementar o input do **valorDoPedido** e da **quantidadeParcelas** [EXIGÊNCIA DE CÓDIGO 2 de 6];
- C. Deve-se implementar o Juros **conforme a enunciado acima** (obs.: atente-se as condições de menor, igual e maior) [EXIGÊNCIA DE CÓDIGO 3 de 6];
- D. Deve-se implementar o **valorDaParcela** e **valorTotalParcelado** [EXIGÊNCIA DE CÓDIGO 4 de 6];
- E. Deve-se implementar as estruturas **if, elif e else (todas elas)** [EXIGÊNCIA DE CÓDIGO 5 de 6];
- F. Deve-se inserir comentários **relevantes** no código [EXIGÊNCIA DE CÓDIGO 6 de 6];
- G. Deve-se apresentar na saída de console uma mensagem com seu nome completo [EXIGÊNCIA DE SAÍDA DE CONSOLE 1 de 2];
- H. Deve-se apresentar na saída de console um parcelamento com Juros (**quantidadeParcelas** maior ou igual a 4) apresentando o valor da Parcela e o valor Total Parcelado [EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 2];

## EXEMPLO DE SAÍDA DE CONSOLE:

```
➡ Bem-vindo a Loja do Bruno Kostiuk
  Entre com o valor do pedido: 1000.00
  Entre com a quantidade de parcelas: 14
  O valor das parcelas é de:R$ 94.29
  O valor Total Parcelado é de:R$ 1320.00
```

Figura 1.1: Exemplo de saída de console que o aluno deve fazer. Em que se perguntar o valor do pedido (pode ser qualquer valor) a quantidade de parcelas (maior ou igual a 4 [EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 2]), e é apresentado o valorDaParcela e o valorTotalParcelado.

# Apresentação de Código da Questão 1:

```
print("Bem-vindo a Loja da Kathleen Soares")

# Solicita ao usuário o valor total do pedido e a quantidade de parcelas desejada.
valorDoPedido = float(input("Entre com o valor do pedido: "))
quantidadeParcelas = int(input("Entre com a quantidade de parcelas: "))

# Atribuição das entradas para variáveis simplificadas
qtd = quantidadeParcelas
pedido = valorDoPedido

# Cálculo da taxa de juros conforme número de parcelas solicitada.
if qtd < 4:
    juros = 0.0 # 0% de juros.

elif qtd < 6:
    juros = 0.04 # 4% de juros.

elif qtd < 9:
    juros = 0.08 # 8% de juros.

elif qtd < 13:
    juros = 0.16 # 16% de juros.

else:
    juros = 0.32 # 32% de juros.

valorParcela = pedido * (1 + juros) / qtd # Cálculo do valor das parcelas incluindo a taxa de juros.
valorTotalParcela = valorParcela * qtd # Cálculo do valor total do pedido parcelado com a taxa de juros.

# Output do valor de cada parcela e o valor final do pedido com juros.
print(f'O valor das Parcelas é de: R$ {valorParcela:.2f}')
print(f'O valor Total Parcelado é de: R$ {valorTotalParcela:.2f}')
```

Apresentação de **Saída do Console da Questão 1:**

```
➡ Bem-vindo a Loja da Kathleen Soares
  Entre com o valor do pedido: 1000.00
  Entre com a quantidade de parcelas: 14
  O valor das Parcelas é de: R$ 94.29
  O valor Total Parcelado é de: R$ 1320.00
```

# QUESTÃO 2 de 4 - Conteúdo até aula 04

Enunciado: Você e sua equipe de programadores foram contratados para desenvolver um app de vendas para uma loja que vende Marmitas de Bife Acebolado ou Filé de Frango. Você ficou com a parte de desenvolver a interface do cliente para retirada do produto.


A Loja possui seguinte relação:

- Tamanho **P** de Bife Acebolado (**BA**) custa 16 reais e o Filé de Frango (**FF**) custa 15 reais;
- Tamanho **M** de Bife Acebolado (**BA**) custa 18 reais e o Filé de Frango (**FF**) custa 17 reais;
- Tamanho **G** de Bife Acebolado (**BA**) custa 22 reais e o Filé de Frango (**FF**) custa 21 reais;

Elabore um programa em Python que:

- A. Deve-se implementar o **print** com o seu **nome completo** (somente print, não usar input aqui).  
Por exemplo: **print(“Bem vindos a loja de Marmitas do Bruno Kostiuk”)**  
Além do seu nome completo, deve-se implementar um **print com um Menu** para o cliente. [EXIGÊNCIA DE CÓDIGO 1 de 8];
- B. Deve-se implementar o input do **sabor** (BA/FF) e o print “Sabor inválido. Tente novamente” se o usuário entra com valor diferente de BA e FF [EXIGÊNCIA DE CÓDIGO 2 de 8];
- C. Deve-se implementar o input do **tamanho** (P/M/G) e o print “Tamanho inválido. Tente novamente” se o usuário com entra valor diferente de P, M ou G [EXIGÊNCIA DE CÓDIGO 3 de 8];
- D. Deve-se implementar **if, elif e/ou else**, utilizando o modelo **aninhado** (aula 3 – Tema 4) com cada uma das combinações de **sabor** e **tamanho** [EXIGÊNCIA DE CÓDIGO 4 de 8];
- E. Deve-se implementar um **acumulador** para somar os valores dos pedidos [EXIGÊNCIA DE CÓDIGO 5 de 8];
- F. Deve-se implementar o input com a pergunta: “Deseja pedir mais alguma coisa?”. Se sim **repetir a partir do item B**, senão encerrar o programa executar o print do **acumulador** [EXIGÊNCIA DE CÓDIGO 6 de 8];
- G. Deve-se implementar as estruturas de **while, break, continue (todas elas)** [EXIGÊNCIA DE CÓDIGO 7 de 8];
- H. Deve-se inserir comentários **relevantes** no código [EXIGÊNCIA DE CÓDIGO 8 de 8];
- I. Deve-se apresentar na saída de console uma mensagem com o seu nome completo e o menu para o cliente conhecer as opções [EXIGÊNCIA DE SAÍDA DE CONSOLE 1 de 4];
- J. Deve-se apresentar na saída de console um pedido em que o usuário errou o **sabor** [EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 4];
- K. Deve-se apresentar na saída de console um pedido em que o usuário errou o **tamanho** [EXIGÊNCIA DE SAÍDA DE CONSOLE 3 de 4];
- L. Deve-se apresentar na saída de console um pedido com duas opções sabores diferentes e com tamanhos diferentes [EXIGÊNCIA DE SAÍDA DE CONSOLE 4 de 4];

## EXEMPLO DE SAÍDA DE CONSOLE:



```
----- Bem-vindo a Loja de Marmitas do Bruno Kostiuk -----
-----Cardápio-----
-----
---| Tamanho | Bife Acebolado(BA) | Filé de Frango(FF) | ---
---| P        | R$ 16.00           | R$ 15.00            | ---
---| M        | R$ 18.00           | R$ 17.00            | ---
---| G        | R$ 22.00           | R$ 21.00            | ---
-----

Entre com o sabor desejado (BA/FF): BF
Sabor inválido. Tente novamente

Entre com o sabor desejado (BA/FF): BA
Entre com o tamanho desejado (P/M/G): EXGG
Tamanho inválido. Tente novamente

Entre com o sabor desejado (BA/FF): BA
Entre com o tamanho desejado (P/M/G): M
Você pediu um Bife Acebolado no tamanho M: R$ 18.00

Deseja mais alguma coisa? (S/N): S
Entre com o sabor desejado (BA/FF): FF
Entre com o tamanho desejado (P/M/G): G
Você pediu um Filé de Frango no tamanho G: R$ 21.00

Deseja mais alguma coisa? (S/N): N

O valor total a ser pago: R$ 39.00
```

Mensagem com seu nome completo e Menu de opções (cardápio)

Usuário errou o sabor

Usuário errou o tamanho

Pedido com 2 itens de tamanhos e sabores diferentes

Figura 2.1: Exemplo de saída de console que o aluno deve fazer. Em que se perguntar o sabor e o tamanho. Há uma tentativa de pedido que se errou o sabor e outra que se errou o tamanho. Há também um pedido com dois itens com sabores e tamanhos diferentes.

# Apresentação de Código da Questão 2:

```
boasVindas = 'Bem-vindos a Loja de Marmitas da Kathleen Soares'
cardapio = 'Cardápio'

# Imprime o menu para o cliente.
print(f"{'-' * 9} {boasVindas} {'-' * 8}")
print(f"{'-' * 28} {cardapio} {'-' * 29}")
print({'-' * 67})
print(f"{'-' * 3}| Tamanho | Bife Acebolado (BA) | Filé de Frango (FF) | {'-' * 3}")
print(f"{'-' * 3}| P | R$ 16.00 | R$ 15.00 | {'-' * 3}")
print(f"{'-' * 3}| M | R$ 18.00 | R$ 17.00 | {'-' * 3}")
print(f"{'-' * 3}| G | R$ 22.00 | R$ 21.00 | {'-' * 3}")
print({'-' * 67})

# Atribui o tamanho e preço para cada opção do menu.
menu = {
    'BA': {'P': 16.00, 'M': 18.00, 'G': 22.00},
    'FF': {'P': 15.00, 'M': 17.00, 'G': 21.00}
}

total = 0.0 # Acumulador da soma dos valores pedidos pelo cliente.

while True:
    sabor = input('Entre com o sabor desejado (BA/FF): ') # Solicita o sabor ao cliente.
    if sabor not in menu: # Verifica se o sabor escolhido é válido.
        print('Sabor inválido. Tente novamente.\n')
        continue

    tamanho = input('Entre com o tamanho desejado (P/M/G): ') # Solicita o tamanho ao cliente.
    if tamanho not in menu[sabor]: # Verifica se o tamanho escolhido é válido.
        print('Tamanho inválido. Tente novamente.\n')
        continue

    # Atribui o preço conforme a escolha do cliente.
    preco = menu[sabor][tamanho]
    total += preco # Adiciona o preço ao total.

    # Conversão dos sabores em nome completo.
    if sabor == 'BA':
        sabor = 'Bife Acebolado'
    elif sabor == 'FF':
        sabor = 'Filé de Frango'

    # Imprime o sabor e tamanho escolhidos assim como o preço.
    print(f'Você pediu um {sabor} no tamanho {tamanho}: R$ {preco:.2f}\n')

    # Verifica se o cliente deseja algo mais.
    algoMais = input('Deseja mais alguma coisa? (S/N): ')
    if algoMais == 'N':
        break

# Imprime o valor total a ser pago.
print(f'\nO valor total a ser pago: R$ {total:.2f}')
```

# Apresentação de Saída do Console da Questão 2:

```

----- Bem-vindos a Loja de Marmitas da Kathleen Soares -----
----- Cardápio -----
-----
---| Tamanho | Bife Acebolado (BA) | Filé de Frango (FF) |---
---| P       | R$ 16.00            | R$ 15.00             |---
---| M       | R$ 18.00            | R$ 17.00             |---
---| G       | R$ 22.00            | R$ 21.00             |---
-----

Entre com o sabor desejado (BA/FF): BF
Sabor inválido. Tente novamente.

Entre com o sabor desejado (BA/FF): BA
Entre com o tamanho desejado (P/M/G): EXGG
Tamanho inválido. Tente novamente.

Entre com o sabor desejado (BA/FF): BA
Entre com o tamanho desejado (P/M/G): M
Você pediu um Bife Acebolado no tamanho M: R$ 18.00

Deseja mais alguma coisa? (S/N): S
Entre com o sabor desejado (BA/FF): FF
Entre com o tamanho desejado (P/M/G): G
Você pediu um Filé de Frango no tamanho G: R$ 21.00

Deseja mais alguma coisa? (S/N): N

0 valor total a ser pago: R$ 39.00

```



# QUESTÃO 3 de 4 - Conteúdo até aula 05

Enunciado: Você foi contratado para desenvolver um sistema de cobrança de serviços de uma fábrica que vende Camisetas em atacado. Você ficou com a parte de desenvolver a interface com o funcionário.

A Fábrica opera as vendas da seguinte maneira:

- Camiseta Manga Curta Simples (MCS), o valor unitário é de um real e oitenta centavos;
  - Camiseta Manga Longa Simples (MLS), o valor unitário é de dois reais e dez centavos;
  - Camiseta Manga Curta Com Estampa (MCE), o valor unitário é de dois reais e noventa centavos;
  - Camiseta Manga Longa Com Estampa (MLE), o valor unitário é de três reais e vinte centavos;
- 
- Se número de camisetas for **menor** que 20 não há desconto na venda;
  - Se número de camisetas for **igual ou maior** que 20 e **menor** que 200, o desconto será de 5%;
  - Se número de camisetas for **igual ou maior** que 200 e **menor** que 2000, o desconto será de 7%;
  - Se número de camisetas for **igual ou maior** que 2000 e **menor ou igual** que 20000, o desconto será de 12%;
  - Se número de camisetas for **maior** que 20000, não é aceito pedidos nessa quantidade de camisetas;
- 
- ♦ Para o **adicional** de frete por transportadora (1) é cobrado um valor **extra** de 100 reais;
  - ♦ Para o **adicional** de frete por Sedex (2) é cobrado um valor **extra** de 200 reais;
  - ♦ Para o **adicional** de retirar o pedido na fábrica (0) é cobrado um valor **extra** de 0 reais;

O valor final da conta é calculado da seguinte maneira:

total = (modelo \* num\_camisetas) + frete

Elabore um programa em Python que:

- A. Deve-se implementar o **print** com o seu **nome completo** (somente print, não usar input aqui).  
Por exemplo: **print(“Bem vindos a Fábrica de Camisetas do Bruno Kostiuk”)** [EXIGÊNCIA DE CÓDIGO 1 de 7];
- B. Deve-se implementar a função **escolha\_modelo()** em que: [EXIGÊNCIA DE CÓDIGO 2 de 7];
  - a. Pergunta o **modelo** desejado;
  - b. **Retorna o valor do modelo** com base na escolha do usuário (use **return**);
  - c. Repete a pergunta do item **B.a** se digitar uma opção diferente de: MCS/MLS/MCE/MLE;
- C. Deve-se implementar a função **num\_camisetas()** em que: [EXIGÊNCIA DE CÓDIGO 3 de 7];
  - a. Pergunta o **número de camisetas**;
  - b. **Retorna** (use **return**) o **número de camisetas** com desconto seguindo a regra do enunciado (desconto calculado em cima do número de camisetas);
  - c. Repete a pergunta do item **C.a** se digitar um valor acima de 20000 ou valor não numérico (use try/except para não numérico)
- D. Deve-se implementar a função **frete()** em que: [EXIGÊNCIA DE CÓDIGO 4 de 7];
  - a. Pergunta pelo serviço **adicional de frete**;
  - b. **Retorna** (use **return**) o **valor** de apenas uma das **opções** de **frete**
  - c. Repetir a pergunta item **D.a** se digitar uma opção diferente de: 1/2/0;
- E. Deve-se implementar o total a pagar no código principal (**main**), ou seja, não pode estar dentro de função, conforme o enunciado [EXIGÊNCIA DE CÓDIGO 5 de 7];
- F. Deve-se implementar **try/except** [EXIGÊNCIA DE CÓDIGO 6 de 7];
- G. Deve-se inserir comentários relevantes no código [EXIGÊNCIA DE CÓDIGO 7 de 7];
- H. Deve-se apresentar na saída de console uma mensagem com o seu nome completo [EXIGÊNCIA DE SAÍDA DE CONSOLE 1 de 4];
- I. Deve-se apresentar na saída de console um pedido no qual o usuário errou a opção de modelo [EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 4];
- J. Deve-se apresentar na saída de console um pedido no qual o usuário digitou ultrapassou no número de camisetas [EXIGÊNCIA DE SAÍDA DE CONSOLE 3 de 4];
- K. Deve-se apresentar na saída de console um pedido com opção de modelo, número de camisetas e frete válidos [EXIGÊNCIA DE SAÍDA DE CONSOLE 4 de 4];

EXEMPLO DE SAÍDA DE CONSOLE:

```
➡ Bem vindo a Fábrica de Camisetas do do Bruno Kostiuk

Entre com o modelo desejado
MCS - Manga Curta Simples
MLS - Manga Longa Simples
MCE - Manga Curta Com Estampa
MLE - Manga Longa Com Estampa
>>MLL                                     Errou o modelo
Escolha inválida, entre com o modelo novamente

Entre com o modelo desejado
MCS - Manga Curta Simples
MLS - Manga Longa Simples
MCE - Manga Curta Com Estampa
MLE - Manga Longa Com Estampa
>>MCS
Entre com o número de camisetas: 300000    Errou o número de camisetas
Não aceitamos tantas camisetas de uma vez.
Por favor, entre com o número de camisetas novamente.

Entre com o número de camisetas: 10000

Escolha o tipo de frete:
1 - Frete por transportadora - R$ 100.00
2 - Frete por Sedex - R$ 200.00
0 - Retirar pedido na fábrica - R$ 0.00    Pedido com modelo, número de
>>2                                         camisetas e frete válidos
Total: R$ 16040.00 (Modelo: 1.80 * Quantidade(com desconto): 8800 + frete: 200.00)
```

**Figura 3.1: Exemplo de saída de console que o aluno deve fazer. Em que se pergunta pelo modelo e se erra opção inicialmente, que se passa do número de camisetas acima do aceito. Na sequência, o usuário digitou um modelo, número de camisetas e frete válidos.**



# Apresentação de Código da Questão 3:

```
print("Bem vindo a Fábrica de Camisetas da Kathleen Soares\n")

# Controla o fluxo das etapas.
validacao = 1

# Dicionário para definição dos modelos e valores de camisetas.
modelo_camisetas = {
    "MCS": 1.80,
    "MLS": 2.10,
    "MCE": 2.90,
    "MLE": 3.20
}

# Função para solicitar ao cliente a escolha do modelo de camisetas.
def escolha_modelo():
    global validacao # Controla o fluxo dentro da função.
    while validacao == 1:
        # Imprime na tela as opções de camisetas para o cliente.
        print('Entre com o modelo desejado ')
        print('MCS - Manga Curta Simples')
        print('MLS - Manga Longa Simples')
        print('MCE - Manga Curta com Estampa')
        print('MLE - Manga Longa com Estampa')

        modelo_desejado = input('>> ')

        if modelo_desejado not in modelo_camisetas:
            print('Escolha inválida, entre com o modelo novamente.\n')
        else:
            validacao = 2 # Atualiza a validação para a próxima etapa.
            return modelo_desejado

# Chama a função e armazena o resultado.
modeloEscolhido = escolha_modelo()

# Função para solicitar ao cliente a quantidade de camisetas.
def num_camisetas():
    global validacao
    while validacao == 2:
        try:
            quantidade = int(input('Entre com o número de camisetas: '))

            if quantidade > 20000: # Verifica se a quantidade excede o número permitido.
                print('Não aceitamos tantas camisetas de uma vez.')
                print('Por favor, entre com o número de camisetas novamente.\n')
            elif quantidade < 1:
                print('Por favor, entre com o número de camisetas novamente.\n')
            else:
                validacao = 3
                return quantidade

        except ValueError:
            print('Entrada Inválida.')

# Chama a função e armazena o resultado.
numero_camisas = num_camisetas()

# Dicionário para definição dos valores do frete.
valoresFrete = {
    "1": 100.00,
    "2": 200.00,
    "0": 0.00
}

# Função que solicita o tipo de frete a ser escolhido.
def frete():
```

```
global validacao

while validacao == 3:

    print("\nEscolha o tipo de frete:")
    print('1 - Frete por transportadora - R$ 100.00')
    print('2 - Frete por Sedex - R$ 200.00')
    print('0 - Retirar pedido na fábrica - R$ 0.00')

    frete_escolhido = input('>> ')

    if frete_escolhido not in valoresFrete:

        frete() # Chama a função novamente caso a escolha seja inválida.
    else:

        validacao = 4

    return frete_escolhido


# Chama a função e armazena o resultado.
tipoFrete = frete()


# Função para calcular o desconto baseado na quantidade de camisetas escolhida.
def desconto():

    if numero_camisas < 20:

        percentagem_desconto = 0.0 # 0%
        return percentagem_desconto

    if 20 <= numero_camisas < 200:

        percentagem_desconto = 0.05 # 5%
        return percentagem_desconto

    elif 200 <= numero_camisas < 2000:

        percentagem_desconto = 0.07 # 7%
        return percentagem_desconto

    elif 2000 <= numero_camisas <= 20000:

        percentagem_desconto = 0.12 # 12%
        return percentagem_desconto


# Chama a função e calcula o desconto.
totalDesconto = desconto()


# Recupera o preço do modelo escolhido e o valor do frete
x = modelo_camisetas[modeloEscolhido]
y = valoresFrete[tipoFrete]


# Calcula a quantidade de camisetas com desconto aplicado
qtd_com_desc = numero_camisas - (numero_camisas * totalDesconto)


# Calcula o valor total das camisetas.
valor_camisas = numero_camisas * modelo_camisetas[modeloEscolhido]


# Calcula o valor do desconto aplicado.
desconto_valor = valor_camisas * totalDesconto


# Calcula o valor final da conta, subtraindo o desconto e somando o frete.
valorTotal = (valor_camisas - desconto_valor) + valoresFrete[tipoFrete]


# Exibe o total formatado
print(f"Total R$ {valorTotal: .2f} (Modelo: {x: .2f} * Quantidade(com desconto): {qtd_com_desc: .0f} + frete: {y: .2f}) ")
```

# Apresentação de Saída do Console da Questão 3

```
🔄 Bem vindo a Fábrica de Camisetas da Kathleen Soares

Entre com o modelo desejado
MCS - Manga Curta Simples
MLS - Manga Longa Simples
MCE - Manga Curta com Estampa
MLE - Manga Longa com Estampa
>> MLL
Escolha inválida, entre com o modelo novamente.

Entre com o modelo desejado
MCS - Manga Curta Simples
MLS - Manga Longa Simples
MCE - Manga Curta com Estampa
MLE - Manga Longa com Estampa
>> MCS
Entre com o número de camisetas: 300000
Não aceitamos tantas camisetas de uma vez.
Por favor, entre com o número de camisetas novamente.

Entre com o número de camisetas: 10000

Escolha o tipo de frete:
1 - Frete por transportadora - R$ 100.00
2 - Frete por Sedex - R$ 200.00
0 - Retirar pedido na fábrica - R$ 0.00
>> 2
Total R$ 16040.00 (Modelo: 1.80 * Quantidade(com desconto): 8800 + frete: 200.00)
```

# QUESTÃO 4 de 4 - Conteúdo até aula 06

Enunciado: Você e sua equipe de programadores foram contratados por pequena empresa para desenvolver o software de gerenciamento de funcionários. Este software deve ter o seguinte menu e opções:

- 1) Cadastrar Funcionário
- 2) Consultar Funcionário
  - 1. Consultar Todos
  - 2. Consultar por Id
  - 3. Consultar por setor
  - 4. Retornar ao menu
- 3) Remover Funcionário
- 4) Encerrar Programa

Elabore um programa em Python que:

- A. Deve-se implementar o **print** com o seu **nome completo** (somente print, não usar input aqui).  
Por exemplo: **print(“Bem vindos a empresa do Bruno Kostiuk”)** [EXIGÊNCIA DE CÓDIGO 1 de 8];
- B. Deve-se implementar uma lista com o nome de **lista\_funcionarios** e a variável **id\_global** com valor inicial igual ao número de seu RU [EXIGÊNCIA DE CÓDIGO 2 de 8];
- C. Deve-se implementar uma função chamada **cadastrar\_funcionario(id)** em que: [EXIGÊNCIA DE CÓDIGO 3 de 8];
  - a. Pergunta **nome, setor, salario** do funcionário;
  - b. Armazena o **id** (este é fornecido via parâmetro da função), **nome, setor, salario** dentro de um dicionário;
  - c. **Copiar** o dicionário para dentro da **lista\_funcionarios** (utilizar o **copy**);
- D. Deve-se implementar uma função chamada **consultar\_funcionarios()** em que: [EXIGÊNCIA DE CÓDIGO 4 de 8];
  - a. Deve-se perguntar qual opção deseja (1. Consultar Todos / 2. Consultar por Id / 3. Consultar por Setor / 4. Retornar ao menu):
    - i. Se Consultar Todos, apresentar todos os funcionários com todos os seus dados cadastrados;
    - ii. Se Consultar por Id, solicitar ao usuário que informe um id, e apresentar o funcionário específico com todos os seus dados cadastrados;
    - iii. Se Consultar por Setor, solicitar ao usuário que informe o setor, e apresentar o(s) funcionário(s) do setor com todos os seus dados cadastrados;
    - iv. Se Retornar ao menu, deve-se **retornar** ao menu principal (return);
    - v. Se Entrar com um valor diferente de 1, 2, 3 ou 4, printar “Opção inválida” e repetir a pergunta **D.a.**
    - vi. Enquanto o usuário não escolher a opção 4, o menu consultar funcionários deve se repetir.
- E. Deve-se implementar uma função chamada **remover\_funcionario()** em que: [EXIGÊNCIA DE CÓDIGO 5 de 8];
  - a. Deve-se pergunta pelo **id** do funcionário a ser removido;
  - b. Remover o funcionário da **lista\_funcionarios**;
  - c. Se o id fornecido não for de um funcionário da lista, printar “Id inválido” e repetir a pergunta **E.a.**
- F. Deve-se implementar uma estrutura de menu no código principal (**main**), ou seja, **não pode estar dentro de função**, em que: [EXIGÊNCIA DE CÓDIGO 6 de 8];
  - a. Deve-se pergunta qual opção deseja (1. Cadastrar Funcionário / 2. Consultar Funcionário / 3. Remover Funcionário / 4. Encerrar Programa):
    - i. Se Cadastrar Funcionário, **incrementar** em um **id\_ global** e chamar a função **cadastrar\_funcionario(id\_ global)**;
    - ii. Se Consultar Funcionário, chamar função **consultar\_funcionario ()**;
    - iii. Se Remover Funcionário, chamar função **remover\_funcionario()**;
    - iv. Se Encerrar Programa, sair do menu (e com isso acabar a execução do código);
    - v. Se Entrar com um valor diferente de 1, 2, 3 ou 4, printar “Opção inválida” e repetir a pergunta **F.a.**
    - vi. Enquanto o usuário não escolher a opção 4, o menu deve se repetir.
- G. Deve-se implementar uma **lista de dicionários** (uma lista contento dicionários dentro) [EXIGÊNCIA DE CÓDIGO 7 de 8];
- H. Deve-se inserir comentários **relevantes** no código [EXIGÊNCIA DE CÓDIGO 8 de 8];
- I. Deve-se apresentar na saída de console uma mensagem com o seu nome completo [EXIGÊNCIA DE SAÍDA DE CONSOLE 1 de 6];
- J. Deve-se apresentar na saída de console um cadastro de 3 funcionários (sendo **2** deles no mesmo setor) [EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 6];
- K. Deve-se apresentar na saída de console uma consulta de todos os funcionários [EXIGÊNCIA DE SAÍDA DE CONSOLE 3 de 6];
- L. Deve-se apresentar na saída de console uma consulta por código (id) de um dos funcionários [EXIGÊNCIA DE SAÍDA DE CONSOLE 4 de 6];
- M. Deve-se apresentar na saída de console uma consulta por setor em que **2** funcionários sejam do mesmo setor [EXIGÊNCIA DE SAÍDA DE CONSOLE 5 de 6];
- N. Deve-se apresentar na saída de console uma remoção de um dos funcionários seguida de uma consulta de todos os funcionários [EXIGÊNCIA DE SAÍDA DE CONSOLE 6 de 6];

EXEMPLO DE SAÍDA DE CONSOLE:

```
➡ Bem vindo a Empresa do Bruno Kostiuk      Mensagem com seu nome completo
-----
----- MENU PRINCIPAL -----
Escolha a opção desejada:
1 - Cadastrar Funcionários
2 - Consultar Funcionário(s)
3 - Remover Funcionário
4 - Sair
>>1
-----
----- MENU CADASTRAR FUNCIONÁRIO -----
Id do Funcionário: 4297914
Por favor entre com o nome do Funcionário: Bruno      Cadastro do primeiro funcionário
Por favor entre com o setor do Funcionário: Configurações
Por favor entre com o salário do Funcionário: 1000.00
-----
```

Figura 4.1: Exemplo de saída de console que o aluno deve fazer. Apresenta o print com seu nome completo e é realizado o cadastro do primeiro funcionário, note que o ID do funcionário inicia em 4297914 pois o id\_global é inicializado em 4297913.

```
➡ ----- MENU PRINCIPAL -----
Escolha a opção desejada:
1 - Cadastrar Funcionários
2 - Consultar Funcionário(s)
3 - Remover Funcionário
4 - Sair
>>1
-----
----- MENU CADASTRAR FUNCIONÁRIO -----
Id do Funcionário: 4297915
Por favor entre com o nome do Funcionário: Tamy
Por favor entre com o setor do Funcionário: Tutorias
Por favor entre com o salário do Funcionário: 1000
-----
----- MENU PRINCIPAL -----
Escolha a opção desejada:
1 - Cadastrar Funcionários
2 - Consultar Funcionário(s)
3 - Remover Funcionário
4 - Sair
>>1
-----
----- MENU CADASTRAR FUNCIONÁRIO -----
Id do Funcionário: 4297916
Por favor entre com o nome do Funcionário: Osmar
Por favor entre com o setor do Funcionário: Tutorias
Por favor entre com o salário do Funcionário: 1000.00
-----
```

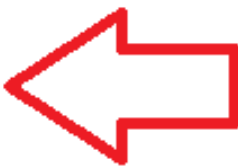


Figura 4.2: Exemplo de saída de console que o aluno deve fazer. São cadastrados mais dois funcionários com mesmo setor.

```
-----  
----- MENU PRINCIPAL -----  
Escolha a opção desejada:  
1 - Cadastrar Funcionários  
2 - Consultar Funcionário(s)  
3 - Remover Funcionário  
4 - Sair  
>>2  
-----  
----- MENU CONSULTAR FUNCIONÁRIO -----  
Escolha a opção desejada:  
1 - Consultar Todos os Funcionários  
2 - Consultar Funcionário por id  
3 - Consultar Funcionário(s) por setor  
4 - Retornar  
>>1  
-----  
id: 4297914  
nome: Bruno  
setor: Configurações  
salário: 1000.0  
  
id: 4297915  
nome: Tamy  
setor: Tutorias  
salário: 1000.0  
  
id: 4297916  
nome: Osmar  
setor: Tutorias  
salário: 1000.0  
-----
```

**Consulta todos os  
funcionários cadastrados**

Figura 4.3: Exemplo de saída de console que o aluno deve fazer. Em que se consulta Todos os funcionários cadastrados.

```
-----  
----- MENU CONSULTAR FUNCIONÁRIO -----  
Escolha a opção desejada:  
1 - Consultar Todos os Funcionários  
2 - Consultar Funcionário por id  
3 - Consultar Funcionário(s) por setor  
4 - Retornar  
>>2  
Digite o id do funcionário: 4297914  
-----  
id: 4297914  
nome: Bruno  
setor: Configurações  
salario: 1000.0  
  
-----  
-----  
----- MENU CONSULTAR FUNCIONÁRIO -----  
Escolha a opção desejada:  
1 - Consultar Todos os Funcionários  
2 - Consultar Funcionário por id  
3 - Consultar Funcionário(s) por setor  
4 - Retornar  
>>3  
Digite o setor do(s) funcionário(s): Tutorias  
-----  
id: 4297915  
nome: Tamy  
setor: Tutorias  
salario: 1000.0  
  
id: 4297916  
nome: Osmar  
setor: Tutorias  
salario: 1000.0
```

**Consulta por ID**

**Consulta por Setor**

Figura 4.4: Exemplo de saída de console que o aluno deve fazer. Em que se consulta o funcionário com id número 4297914 e consulta pelo nome do setor (Tutorias).





```
----- MENU PRINCIPAL -----
Escolha a opção desejada:
1 - Cadastrar Funcionários
2 - Consultar Funcionário(s)
3 - Remover Funcionário
4 - Sair
>>3
-----
----- MENU REMOVER FUNCIONÁRIO -----
Digite o id do funcionario a ser removido: 4297914
Funcionário removido com sucesso!
-----
----- MENU PRINCIPAL -----
Escolha a opção desejada:
1 - Cadastrar Funcionários
2 - Consultar Funcionário(s)
3 - Remover Funcionário
4 - Sair
>>2
-----
----- MENU CONSULTAR FUNCIONÁRIO -----
Escolha a opção desejada:
1 - Consultar Todos os Funcionários
2 - Consultar Funcionário por id
3 - Consultar Funcionário(s) por setor
4 - Retornar
>>1
-----
id: 4297915
nome: Tamy
setor: Tutorias
salário: 1000.0

id: 4297916
nome: Osmar
setor: Tutorias
salário: 1000.0
```

Remove um  
funcionário

Realiza o consultar Todos mostrando  
que o funcionário foi removido

**Figura 4.5: Exemplo de saída de console que o aluno deve fazer. Em que se remove o funcionário de Id número 4297914 e depois se faz uma consulta de todos os funcionários.**

# Apresentação de Código da Questão 4:

```
print("Bem vindos a Empresa da Kathleen Soares")
print('-' * 60)

id_global = 4648267 # Declaração de uma variável global para gerar IDs únicos a cada iteração.
lista_funcionarios = [] # Declaração de uma lista vazia que irá armazenar os dados dos funcionários cadastrados.

def menu_principal(id): # Definição da função que receberá o ID como argumento.
    while True: # Loop infinito para exibir a estrutura de menu e processar as opções selecionadas pelo usuário.
        print(f'{'-' * 22} MENU PRINCIPAL {'-' * 22}')
        print('Escolha a opção desejada: ')
        print('1 - Cadastrar Funcionários')
        print('2 - Consultar Funcionário(s)')
        print('3 - Remover Funcionário')
        print('4 - Sair')

        try:
            menu = int(input('>> ')) # Entrada do usuário para a opção do menu.
        except ValueError: # Verificação do dado de entrada caso não seja um número.
            print('Entrada inválida. Por favor, escolha uma opção numérica.')
            continue

        if menu == 1: # Incrementa o ID e chama a função para cadastrar funcionário.
            id += 1
            cadastrar_funcionario(id)

        elif menu == 2: # Chama a função para consultar funcionário.
            consultar_funcionario()

        elif menu == 3: # Chama a função para remover funcionário.
            remover_funcionario()

        elif menu == 4: # Encerra o programa.
            print('Programa Encerrado.')
            break

        else: # Retorna 'opção inválida' caso a entrada seja diferente das opções fornecidas.
            print('Opção Inválida.')

def cadastrar_funcionario(id): # Função para cadastrar funcionário.
    while True:
        print('-' * 60)
        print(f'{'-' * 16} MENU CADASTRAR FUNCIONÁRIO {'-' * 16}')
        print(f'Id do Funcionário: {id}')

        # Entrada de dados para cadastro do funcionário.
        nome = input('Por favor entre com o nome do Funcionário: ')
        setor = input('Por favor entre com o setor do Funcionário: ')

        try: # Entrada do salário e verificação do dado de entrada caso não seja numérico.
            salario = float(input(f'Por favor entre com o salário do Funcionário: '))
        except ValueError:
            print('Entrada inválida. Por favor, insira um valor numérico para o salário.')
            continue

        # Criação de um dicionário para armazenar os dados do funcionário.
        cadastro = {
            'id': id,
            'nome': nome,
            'setor': setor,
            'salario': salario
        }

        # Copia o dicionário para dentro da lista funcionários.
        lista_funcionarios.append(cadastro.copy())
        menu_principal(id) # Retorna ao menu principal.
        break

def consultar_funcionario(): # Função para consultar funcionários cadastrados.
    print('-' * 60)
    print(f'{'-' * 16} MENU CONSULTAR FUNCIONÁRIO {'-' * 16}')
    print('Escolha a opção desejada:')
    print('1 - Consultar Todos os Funcionários')
    print('2 - Consultar Funcionário por id')
    print('3 - Consultar Funcionário(s) por setor')
    print('4 - Retornar')

    try: # Entrada de dado e verificação de erro caso não seja numérico.
        consultar = int(input('>> '))
    except ValueError:
        print('Entrada inválida. Por favor, escolha uma opção numérica.')
        return

    print('-' * 20)
    if consultar == 1: # Processamento da opção escolhida pelo usuário
        for funcionario in lista_funcionarios: # Retorna todas os funcionários cadastrados.
            print(f'id: {funcionario["id"]}')
            print(f'nome: {funcionario["nome"]}')
            print(f'setor: {funcionario["setor"]}')
            print(f'salário: {funcionario["salario"]}')
            print()

    elif consultar == 2: # Processamento da entrada com base no ID fornecido.
        digite_id = int(input('Digite o id do funcionário: '))
```

```
print('-' * 20)

for funcionario in lista_funcionarios:
    if digite_id == funcionario['id']:
        print(f'id: {funcionario["id"]}')
        print(f'nome: {funcionario["nome"]}')
        print(f'setor: {funcionario["setor"]}')
        print(f'salário: {funcionario["salario"]}')
        print('-' * 60)
        break

    else: # Verificação de erro.
        print('Funcionário não encontrado.')

elif consultar == 3: # Processamento da entrada por setor fornecido.
    digite_setor = input('Digite o setor do(s) funcionário(s): ')
    print('-' * 20)

    encontrado = False # Variável para determinar se o funcionário foi encontrado ou não.
    for funcionario in lista_funcionarios:
        if digite_setor == funcionario['setor']: # Verifica se o setor do funcionário corresponde ao fornecido.
            print(f'id: {funcionario["id"]}')
            print(f'nome: {funcionario["nome"]}')
            print(f'setor: {funcionario["setor"]}')
            print(f'salário: {funcionario["salario"]}')
            print()
            encontrado = True

        if not encontrado: # Verificação de erro.
            print('Funcionário não encontrado.')

elif consultar == 4: # Retorna ao menu principal.
    return

else: # Verificação de erro e retorna ao menu consultar funcionário.
    print('Opção Inválida.')
    consultar_funcionario()

def remover_funcionario(): # Função para remover um funcionário.
    print('-' * 60)
    print(f'{'-' * 17} MENU REMOVER FUNCIONÁRIO {'-' * 17}')

    try: # Entrada de dado via ID e verificação de erro caso não seja válido.
        rmv_funcionario = int(input('Digite o id do funcionário a ser removido: '))
    except ValueError:
        print('Entrada inválida. Por favor, insira um id numérico.')
        return

    for funcionario in lista_funcionarios: # Remove funcionário com ID fornecido.
        if rmv_funcionario == funcionario['id']:
            lista_funcionarios.remove(funcionario)
            print('Funcionário removido com sucesso!')
            print('-' * 60)
            return

    print('Id inválido.') # Verificação de erro caso não seja correspondente.
    remover_funcionario() # Retorna ao menu remover funcionário.

menu_principal(id_global) # Chama a função menu principal com o ID Global.
```

# Apresentação de Saída do Console da Questão 4:

```

▶ Bem vindos a Empresa da Kathleen Soares
[⇌] ----- MENU PRINCIPAL -----
Escolha a opção desejada:
1 - Cadastrar Funcionários
2 - Consultar Funcionário(s)
3 - Remover Funcionário
4 - Sair
>> 1

----- MENU CADASTRAR FUNCIONÁRIO -----
Id do Funcionário: 4648268
Por favor entre com o nome do Funcionário: Kathleen
Por favor entre com o setor do Funcionário: Configurações
Por favor entre com o salário do Funcionário: 1000.00

----- MENU PRINCIPAL -----
Escolha a opção desejada:
1 - Cadastrar Funcionários
2 - Consultar Funcionário(s)
3 - Remover Funcionário
4 - Sair
>> 1

----- MENU CADASTRAR FUNCIONÁRIO -----
Id do Funcionário: 4648269
Por favor entre com o nome do Funcionário: Tamy
Por favor entre com o setor do Funcionário: Tutorias
Por favor entre com o salário do Funcionário: 1000.00

----- MENU PRINCIPAL -----
Escolha a opção desejada:
1 - Cadastrar Funcionários
2 - Consultar Funcionário(s)
3 - Remover Funcionário
4 - Sair
>> 1
-----

▶ ----- MENU CADASTRAR FUNCIONÁRIO -----
[⇌] Id do Funcionário: 4648270
Por favor entre com o nome do Funcionário: Osmar
Por favor entre com o setor do Funcionário: Tutorias
Por favor entre com o salário do Funcionário: 1000.00

----- MENU PRINCIPAL -----
Escolha a opção desejada:
1 - Cadastrar Funcionários
2 - Consultar Funcionário(s)
3 - Remover Funcionário
4 - Sair
>> 2

----- MENU CONSULTAR FUNCIONÁRIO -----
Escolha a opção desejada:
1 - Consultar Todos os Funcionários
2 - Consultar Funcionário por id
3 - Consultar Funcionário(s) por setor
4 - Retornar
>> 1

-----
id: 4648268
nome: Kathleen
setor: Configurações
salário: 1000.0

id: 4648269
nome: Tamy
setor: Tutorias
salário: 1000.0

id: 4648270
nome: Osmar
setor: Tutorias
salário: 1000.0
```

```
----- MENU CONSULTAR FUNCIONÁRIO -----
Escolha a opção desejada:
1 - Consultar Todos os Funcionários
2 - Consultar Funcionário por id
3 - Consultar Funcionário(s) por setor
4 - Retornar
>> 2

-----
Digite o id do funcionário: 4648268

-----
id: 4648268
nome: Kathleen
setor: Configurações
salário: 1000.0

-----

----- MENU CONSULTAR FUNCIONÁRIO -----
Escolha a opção desejada:
1 - Consultar Todos os Funcionários
2 - Consultar Funcionário por id
3 - Consultar Funcionário(s) por setor
4 - Retornar
>> 3

-----
Digite o setor do(s) funcionário(s): Tutorias

-----
id: 4648269
nome: Tamy
setor: Tutorias
salário: 1000.0

-----
id: 4648270
nome: Osmar
setor: Tutorias
salário: 1000.0
```

```
----- MENU CONSULTAR FUNCIONÁRIO -----
Escolha a opção desejada:
1 - Consultar Todos os Funcionários
2 - Consultar Funcionário por id
3 - Consultar Funcionário(s) por setor
4 - Retornar
>> 4

-----

----- MENU PRINCIPAL -----
Escolha a opção desejada:
1 - Cadastrar Funcionários
2 - Consultar Funcionário(s)
3 - Remover Funcionário
4 - Sair
>> 3

-----

----- MENU REMOVER FUNCIONÁRIO -----
Digite o id do funcionário a ser removido: 4648268
Funcionário removido com sucesso!
```

```
----- MENU PRINCIPAL -----
Escolha a opção desejada:
1 - Cadastrar Funcionários
2 - Consultar Funcionário(s)
3 - Remover Funcionário
4 - Sair
>> 2

-----

----- MENU CONSULTAR FUNCIONÁRIO -----
Escolha a opção desejada:
1 - Consultar Todos os Funcionários
2 - Consultar Funcionário por id
3 - Consultar Funcionário(s) por setor
4 - Retornar
>> 1

-----
id: 4648269
nome: Tamy
setor: Tutorias
salário: 1000.0

-----
id: 4648270
nome: Osmar
setor: Tutorias
salário: 1000.0

-----

----- MENU CONSULTAR FUNCIONÁRIO -----
Escolha a opção desejada:
1 - Consultar Todos os Funcionários
2 - Consultar Funcionário por id
3 - Consultar Funcionário(s) por setor
4 - Retornar
>> 4
```

