

Introdução: o funcionamento do programa de cálculo de IMC (Índice de Massa Corporal), desenvolvido em linguagem de programação Python e executado no Google Shell Editor. O IMC é uma medida que avalia a relação sobre peso e altura para classificar o estado nutricional de uma pessoa

Funcionamento do Programa: O programa solicita ao usuário o peso e a altura, em seguida calcula o IMC utilizando a fórmula:

$$\text{IMC} = \text{peso(kg)} \div \text{altura(m)}^2$$

Com base no valor calculado, o programa classifica o imc de acordo com o seguinte parâmetro:

IMC (kg/m^2)	Classificação
Menor que 16,9	Muito abaixo do Peso
17,0 a 18,4	Abaixo do peso
18,5 a 24,9	Peso normal
25,0 a 29,9	Acima do peso
30,0 a 34,9	Obesidade Grau I
35,0 a 40,0	Obesidade Grau II
Maior que 40,0	Obesidade Grau III

A estrutura do código do programa é dividido em três funções principais

1. Função “indice(kilograma, metros)”: Calcula o IMC com base no peso e altura fornecidos.

```
def indice(kilograma, metros):
    return round(kilograma/(metros**2))
```

2. Função “ifelse(indiceretornado)”: Classifica o IMC em uma das categorias predefinidas, utilizando comparações ”if - elif - else”.

```
def ifelse(indiceretornado):
    if indiceretornado in id1:
        return print('Muito abaixo do peso.')
    elif indiceretornado in id2:
        return print("Abaixo do peso.")
    elif indiceretornado in id3:
        return print("Peso normal.")
    elif indiceretornado in id4:
        return print("Acima do peso.")
    elif indiceretornado in id5:
        return print("Obesidade Grau I.")
    elif indiceretornado in id6:
        return print("Obesidade Grau II.")
    elif indiceretornado in id7:
        return print("Obesidade Grau III.")
    else:
        return print("ERRO!")
```

3. Bloco de execução principal: Recebe os inputs do usuário, executa a função de cálculo do IMC e chama a função de classificação “ifelse(indiceretornado)”.

```
kilograma = float(input('Quantos kilos você pesa?: '))
metros = float(input('Qual sua altura?: '))
indiceretornado = indice(kilograma, metros)

print('seu IMC é: ', indiceretornado)

classificacaoretornada = ifelse(indiceretornado)
```

Detalhes da implementação

1. Função de Cálculo: Utiliza a fórmula do IMC com a função “round()” para retornar o valor arredondado.

```
round(kilograma/(metros**2))
```

2. Função de Classificação: valia o IMC retornado e classifica-o com base nas faixas estabelecidas, listadas em variáveis (id1, id2, etc.), cada uma correspondendo a uma categoria.

```
id1 = list(range(0, 17))
id2 = list(range(18, 19))
id3 = list(range(20, 25))
id4 = list(range(26, 30))
id5 = list(range(31, 35))
id6 = list(range(36, 40))
id7 = list(range(51, 101))
```

Conclusão

O programa de cálculo IMC foi implementado com sucesso, fornecendo um cálculo e classificação confiáveis para o usuário. Ele serve como base para o entendimento de funções e estruturas condicionais em Python

```
Quantos kilos você pesa?: 90
Qual sua altura?: 1.80
seu IMC é: 28
Acima do peso.
```

```
def ifelse(indiceretornado):
    if indiceretornado in id1:
        return print('Muito abaixo do peso.')
    elif indiceretornado in id2:
        return print("Abaixo do peso.")
    elif indiceretornado in id3:
        return print("Peso normal.")
    elif indiceretornado in id4:
        return print("Acima do peso.")
    elif indiceretornado in id5:
        return print("Obesidade Grau I.")
    elif indiceretornado in id6:
        return print("Obesidade Grau II.")
    elif indiceretornado in id7:
        return print("Obesidade Grau III.")
    else:
        return print("ERRO!")

def indice(kilograma, metros):
    return round(kilograma/(metros**2))

id1 = list(range(0, 17))
id2 = list(range(18, 19))
id3 = list(range(20, 25))
id4 = list(range(26, 30))
id5 = list(range(31, 35))
id6 = list(range(36, 40))
id7 = list(range(51, 101))

kilograma = float(input('Quantos kilos você pesa?: '))
metros = float(input('Qual sua altura?: '))
indiceretornado = indice(kilograma, metros)

print('seu IMC é: ', indiceretornado)

classificacaoretornada = ifelse(indiceretornado)
```