

01)

```
q01.py x
q01.py > ...
1  data = int(input('Digite o dia: '))
2  mes = int(input('Digite o mes: '))
3  ano = int(input('Digite o dia: '))
4
5  data_atual = int(input('Informe o dia atual: '))
6  mes_atual = int(input('Informe o mes atual: '))
7  ano_atual = int(input('Digite o ano atual: '))
8
9  def CalculaDias(data, mes, ano):
10     anos = ano_atual - ano
11     if mes_atual == mes:
12         idade_anos = ano_atual - ano
13         idade_mes = 0
14     elif mes_atual < mes:
15         idade_anos = anos - 1
16         idade_mes = (mes_atual - mes) + 12
17     elif mes_atual > mes:
18         idade_anos = anos
19         idade_mes = mes_atual - mes
20     idade_mes *= 30
21     idade_anos *= 365
22     dias = idade_mes + idade_anos
23     return dias
24
25 print(CalculaDias(data, mes, ano))
26
```

```
PS C:\Users\Júlia\Documents\IFCE\Fundamentos de Programação\Lista 07> & C:\Python310\python.exe "c:/Users/Júlia/Documents/IFCE/Fundamentos de Programação/Lista 07/q01.py"
Digite o dia: 12
Digite o mes: 06
Digite o dia: 2010
Informe o dia atual: 14
Informe o mes atual: 06
Digite o ano atual: 2023
4745
PS C:\Users\Júlia\Documents\IFCE\Fundamentos de Programação\Lista 07> |
```

02)

```
q02.py > ...
1  def Notas(nota):
2      if 0 <= nota <= 4.9:
3          conceito = 'D'
4      elif 5 <= nota <= 6.9:
5          conceito = 'C'
6      elif 7 <= nota <= 8.9:
7          conceito = 'B'
8      elif 9 <= nota <= 10:
9          conceito = 'A'
10     else:
11         print('Nota Inválida')
12     return conceito
13
14     nota = float(input('Informe a média final: '))
15     print(f'O conceito da nota informada é {Notas(nota)}')
16
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS C:\Users\Júlia\Documents\IFCE\Fundamentos de Programação\Lista 07> & C:/Users/Júlia/
undamentos de Programação/Lista 07/q02.py"
Informe a média final: 9.5
O conceito da nota informada é A
undamentos de Programação/Lista 07/q02.py"
Informe a média final: 8.9
O conceito da nota informada é B
PS C:\Users\Júlia\Documents\IFCE\Fundamentos de Programação\Lista 07> & C:/Users/Júlia/
undamentos de Programação/Lista 07/q02.py"
Informe a média final: 5
O conceito da nota informada é C
PS C:\Users\Júlia\Documents\IFCE\Fundamentos de Programação\Lista 07> & C:/Users/Júlia/
undamentos de Programação/Lista 07/q02.py"
Informe a média final: 3.3
O conceito da nota informada é D
PS C:\Users\Júlia\Documents\IFCE\Fundamentos de Programação\Lista 07> |
```

03)

```
q03.py x
q03.py > Saque
1 def Saque(valor):
2
3     cedula_200 = cedula_100 = cedula_50 = cedula_20 = cedula_10 = cedula_5 = cedula_2 = 0
4
5     while valor != 0:
6
7         if valor % 200 == 0:
8             cedula_200 += 1
9             valor -= 200
10            continue
11
12        if valor % 100 == 0:
13            cedula_100 += 1
14            valor -= 100
15            continue
16
17        if valor % 50 == 0:
18            cedula_50 += 1
19            valor -= 50
20            continue
21
22        if valor % 20 == 0:
23            cedula_20 += 1
24            valor -= 20
25            continue
26
27        if valor % 10 == 0:
28            cedula_10 += 1
29            valor -= 10
30            continue
31
32        if valor % 5 == 0:
33            cedula_5 += 1
```

```
q03.py  X
q03.py > Saque

25         continue
26
27     if valor % 10 == 0:
28         cedula_10 += 1
29         valor -= 10
30         continue
31
32     if valor % 5 == 0:
33         cedula_5 += 1
34         valor -= 5
35         continue
36
37     if valor % 2 == 0:
38         cedula_2 += 1
39         valor -= 2
40         continue
41
42
43     if valor < 4:
44         print("Operação inválida.")
45         break
46
47     if valor % valor == 0 and valor % 1 == 0:
48         if valor > 5:
49             cedula_5 += 1
50             valor = valor - 5
51         if valor > 2:
52             cedula_2 += 1
53             valor = valor - 2
54
55
56     print(f'{cedula_200} de 200')
```

```
q03.py x
q03.py > ...
55     print(f'{cedula_200} de 200')
56     print(f'{cedula_100} de 100')
57     print(f'{cedula_50} de 50')
58     print(f'{cedula_20} de 20')
59     print(f'{cedula_10} de 10')
60     print(f'{cedula_5} de 5')
61     print(f'{cedula_2} de 2')
62     print()
63
64
65 Saque(6)
66 Saque(8)
67 Saque(11)
68 Saque(13)
69 Saque(4)
70 Saque(3)
71

PS C:\Users\Júlia\Documents\IFCE\Fundamentos de Programação\Lista 07\q03
0 de 200
0 de 100
0 de 50
0 de 20
0 de 10
0 de 5
3 de 2

0 de 200
0 de 100
0 de 50
0 de 20
0 de 10
0 de 5
4 de 2

0 de 200
0 de 100
0 de 50
0 de 20
0 de 10
1 de 5
3 de 2

0 de 200
0 de 100
0 de 50
0 de 20
0 de 10
1 de 5
4 de 2

0 de 200
0 de 100
```

```
Python + v [ ] [ ] ...
1 de 5
3 de 2

0 de 200
0 de 100
0 de 50
0 de 20
0 de 10
1 de 5
4 de 2

0 de 200
0 de 100
0 de 50
0 de 20
0 de 10
0 de 5
2 de 2

Operação inválida.
0 de 200
0 de 100
0 de 50
0 de 20
0 de 10
0 de 5
0 de 2

PS C:\Users\Júlia\Documents\IFCE\Fundamentos de Programação\Lista 07>
```

04)

```
q04.py  X
q04.py > ...
1  matriz = [[], [], []]
2
3  def Soma_Linha(matriz):
4      soma1 = soma2 = soma3 = menor = 0
5      for l in range(3):
6          soma1 += matriz[0][l]
7      for l in range(3):
8          soma2 += matriz[1][l]
9      for l in range(3):
10         soma3 += matriz[2][l]
11     for i in range(3):
12         if i == 0:
13             menor = soma1
14         elif soma2 < menor:
15             menor = soma2
16         elif soma3 < menor:
17             menor = soma3
18     resp = print(f'A soma da menor linha é: {menor}')
19     return resp
20
21 def Soma_Coluna(matriz):
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL**

```
Digite um número para [1, 2]: 6
Digite um número para [2, 0]: 7
Digite um número para [2, 1]: 8
Digite um número para [2, 2]: 9
 1  2  3
 4  5  6
 7  8  9
A soma da menor linha é: 6
A soma da maior coluna é: 18
PS C:\Users\Júlia\Documents\IFCE\Fundamentos de Programação\Lista 07> 
```

```
q04.py X
q04.py > ...
20
21 def Soma_Coluna(matriz):
22     soma1 = soma2 = soma3 = maior = 0
23     for c in range(3):
24         soma1 += matriz[c][0]
25     for c in range(3):
26         soma2 += matriz[c][1]
27     for c in range(3):
28         soma3 += matriz[c][2]
29     for i in range(3):
30         if i == 0:
31             maior = soma1
32         elif soma2 > maior:
33             maior = soma2
34         elif soma3 > maior:
35             maior = soma3
36     resp = print(f'A soma da maior coluna é: {maior}')
37     return resp
38
39
40 for i in range(0, 3):
41     for r in range(0, 3):
42         n = int(input(f'Digite um número para [{i}, {r}]: '))
43         matriz[i].append(n)
44
45 for l in range(3):
46     for c in range(3):
47         print(f'{matriz[l][c]:^5}', end='')
48     print()
49
50 linha = Soma_Linha(matriz)
51 coluna = Soma_Coluna(matriz)
52
```

05)

```
q05.py  X
q05.py > ...
1  def CalculaIMC(peso, altura):
2      imc = peso / (altura ** 2)
3      return imc
4
5  def ClassificaIMC(imc):
6      if imc < 18.5:
7          resultado = 'magreza'
8      elif 18.5 <= imc < 24.9:
9          resultado = 'normal'
10     elif 25 <= imc < 29.9:
11         resultado = 'sobrepeso com grau de obesidade 1'
12     elif 30 <= imc < 39.9:
13         resultado = 'obesidade com grau de obesidade 2'
14     elif imc >= 40:
15         resultado = 'obesidade grave com grau de obesidade 3'
16     return resultado
17
18 peso = float(input('Informe o peso: '))
19 altura = float(input('Informe a altura: '))
20 print(f'O valor do IMC é {CalculaIMC(peso,altura):.1f} e tem como classificação {ClassificaIMC(CalculaIMC(peso,altura))}')
21
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Python

Fundamentos de Programação/Lista 07/q05.py
Informe o peso: 48.5
Informe a altura: 1.55
O valor do IMC é 20.2 e tem como classificação normal
PS C:\Users\Júlia\Documents\IFCE\Fundamentos de Programação\Lista 07> & C:/Users/Júlia/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/
Fundamentos de Programação/Lista 07/q05.py"
Informe o peso: 150.7
Informe a altura: 1.73
O valor do IMC é 50.4 e tem como classificação obesidade grave com grau de obesidade 3
PS C:\Users\Júlia\Documents\IFCE\Fundamentos de Programação\Lista 07>