

Programação Estruturada Linguagem Python

Professor Adjenor Cristiano
Queiroz
FAPAM - 2023

Aula 3



HORA DA
CORREÇÃO!



1 - Desenvolva um software que receba uma string do usuário e depois solicite a ele quantas letras da string ele quer que sejam separadas, depois imprima na tela as duas partes da string

```
varString = input("Digite uma String:\n")
varQtd = int(input("Digite a Quantidade:\n"))
print("String digitada: %s"%varString)
print("%d Primeiros Caracteres: %s"
      %
      (varQtd, varString[:varQtd])
      # (varQtd, varString[0:varQtd])
      )
print("Demais Caracteres: %s"
      %
      varString[varQtd:]
      # varString[varQtd:len(varString)]
      )
```

2 - Desenvolva um software que receba do usuário seu nome, seu sobrenome e uma quantidade. Depois solicite a ele quantas vezes ele deseja imprimir o conjunto (nome e sobrenome) e imprima na tela essa quantidade de vezes. Utilize \n para quebra de linha.

Ex_2_Concatenacao.py - F:/Fapam/Programação Estruturada/Aulas/Aula 8 - Aula 3 Python/Ex_2_Concatenacao.py (3.8.0)

File Edit Format Run Options Window Help

```
nome = input("Digite seu nome:\n")
sobreNome = input("Digite seu sobrenome:\n")
qtd = int(input("Digite uma Quantidade:\n"))
print((nome+" "+sobreNome+"\n")*qtd)
```

3 - Desenvolva um software que receba do usuário um valor e imprima na tela este valor formatado para reais (R\$) utilizando composição.

Ex_3_Reais.py - F:/Fapam/Programação Estruturada/Aulas/Aula 8 - Aula 3 Python/Ex_3_Reais.py (3.8.0)

File Edit Format Run Options Window Help

```
valor = float(input("Digite um valor:\n"))
valorDuasCasas = str("%.2f"%valor)
valorReal = valorDuasCasas.replace(".", ",")
print("Valor Digitado: %d"%valor)
print("Valor em Reais: %s"%valorReal)
```

4 - Desenvolva um software que receba do usuário um valor de compra, depois receba um valor em dinheiro e calcule o troco e imprima na tela este valor formatado para reais (R\$) utilizando composição.

```
Ex_4_TrocoReais.py - F:\Fapam/Programação Estruturada/Aulas/Aula 8 - Aula 3 Python/Ex_4_TrocoReais.py (3.8.0)
File Edit Format Run Options Window Help

valorProd = float(input("Digite o valor do Produto:\n"))
valorDin = float(input("Digite o valor Dinheiro:\n"))
Troco = valorDin - valorProd
TrocoDC = str("%.2f"%Troco)
TrocoReal = TrocoDC.replace(".", ",")
print("Valor do troco: %s"%TrocoReal)
```



```
1 valorProd = float(input("Digite o valor do Produto:\n"))
2 valorProdDC = str("%.2f"%valorProd)
3 valorProdReal = valorProdDC.replace(".",",")
4
5 valorDin = float(input("Digite o valor Dinheiro:\n"))
6 valorDinDC = str("%.2f"%valorDin)
7 valorDinReal = valorDinDC.replace(".",",")
8
9 Troco = valorDin - valorProd
10 TrocoDC = str("%.2f"%Troco)
11 TrocoReal = TrocoDC.replace(".",",")
12 print("Valor do Produto: R$ %s \n Valor em Dinheiro: R$
13 #Linha 12 (escrever tudo na mesma Linha)
14 #print("Valor do Produto: R$ %s \n
15         # Valor em Dinheiro: R$ %s \n
16         # Valor do troco: R$ %s"
17         #%(valorProdReal, valorDinReal, TrocoReal))
```



```
Python 3.8.0 (tags/v3.8.0:fa919fd, Oct 14 2019, 19:37:50) [
D64)] on win32
```

```
Type "help", "copyright", "credits" or "license()" for more
>>>
```

```
= RESTART: F:/Fapam/Programação Estruturada/Aulas/Aula 8 -
ocoReais.py
```

```
Digite o valor do Produto:
```

```
150
```

```
Digite o valor Dinheiro:
```

```
200
```

```
Valor do Produto: R$ 150,00
```

```
Valor em Dinheiro: R$ 200,00
```

```
Valor do troco: R$ 50,00
```

```
>>>
```

Dicas do Programador



Use o Menu Options do Editor de Textos IDLE para personalizar seu ambiente de Trabalho!

Mostrar Número da Linha
Clique em Options -> Hide Line Numbers

FormataSalario.py - F:\Fapam\Programação Estruturada\Aulas\Aula 7 - Aula 2 Python\FormataSalario.py (3.8.0)

File Edit Format Run Options Window Help

Configure IDLE

Show Code Context

Hide Line Numbers

Zoom Height Alt+2

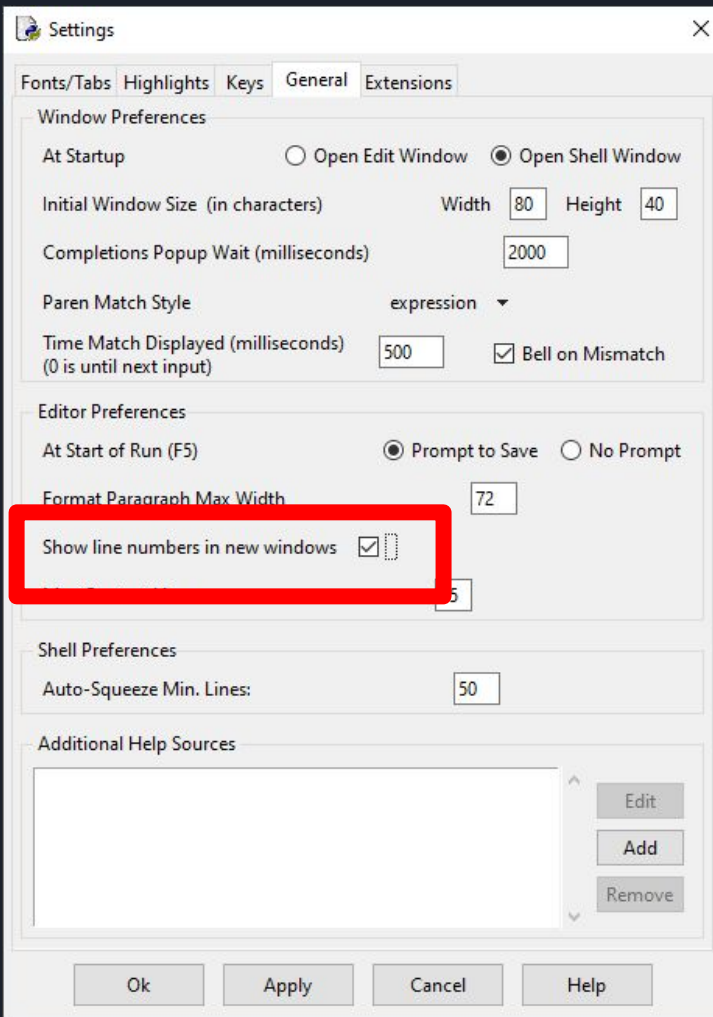
```
1 nome = input("Digite seu no  
2 idade = input("Digite s  
3 salario = float(input("Digi  
4 horas = int(input("Digite s  
5  
6 salarioStr = str("%.2f"%sal  
7 salarioReal = salarioStr.re
```

Dicas do Programador



**Você definir como padrão
Sempre Exibir Número de Linhas
(RECOMENDADO)**

Sempre Exibir Número de Linhas
Clique em Options -> Configure IDLE
->General->Show line numbers in new
windows



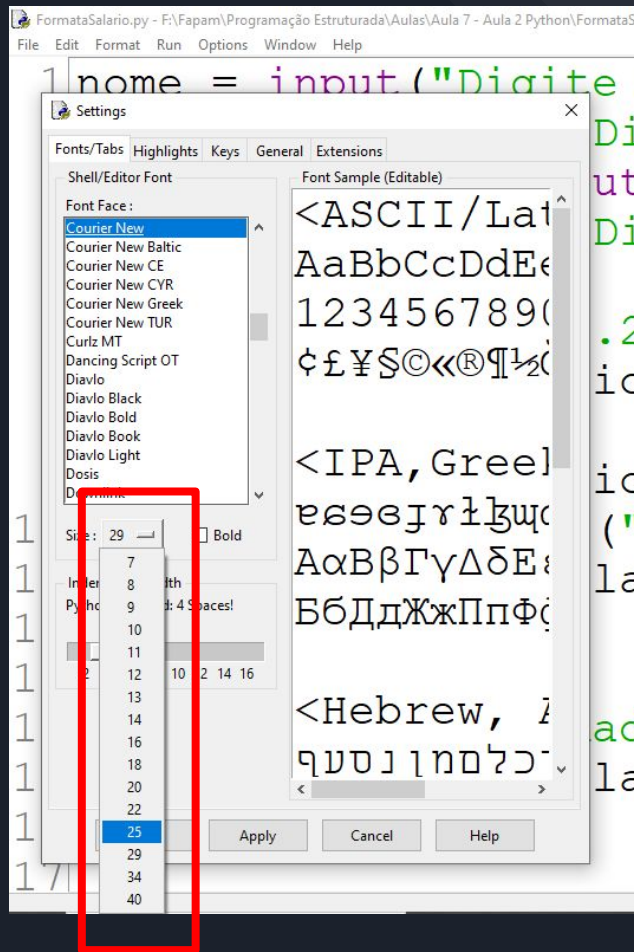
Dicas do Programador



Você pode aumentar a fonte do seu editor (para que seu professor meio ceguinho enxergue seu código)

Aumentar a Fonte

Clique em Options -> Configure IDLE
->Size

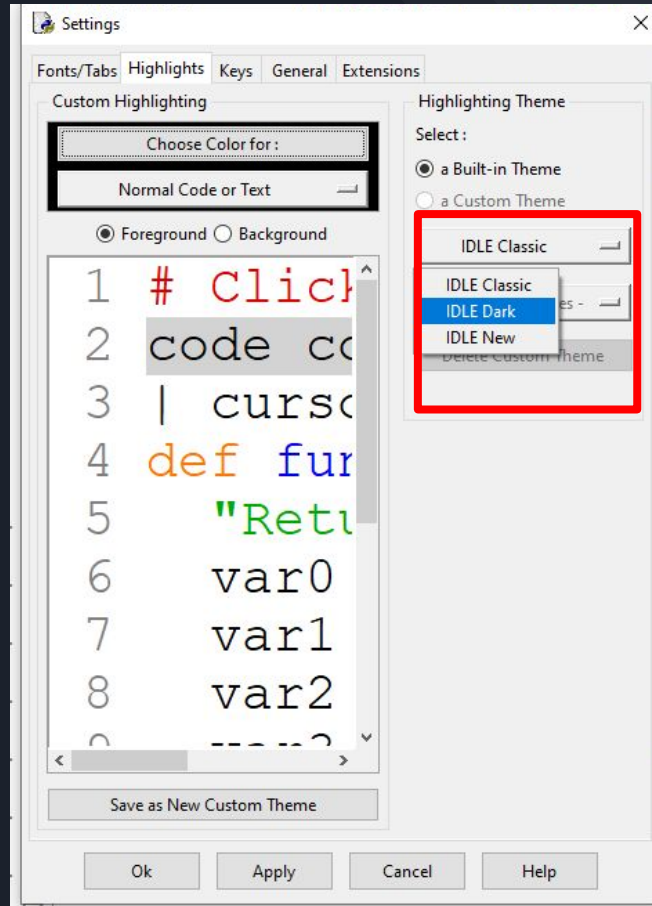


Dicas do Programador



Você pode alterar as aparências do seu editor pra que ele fique chique igual aqueles vídeos do YouTube!

Colocar editor Chique:
Clique em Options -> Configure IDLE
->Highlights->IDLE Classic->IDLE Dark



Agora parece aqueles cara bão do YouTube!

```
FormataSalario.py - F:\Fapam\Programação Estruturada\Aulas\Aula 7 - Aula 2 Python\FormataSalario.py (3.8.0)
File Edit Format Run Options Window Help
1 nome = input("Digite seu nome")
2 idade = int(input("Digite sua Idade"))
3 salario = float(input("Digite seu salario"))#1000
4 horas = int(input("Digite sua Carga Horária"))#1000
5
6 salarioStr = str("%.2f"%salario)#1000.00
7 salarioReal = salarioStr.replace(".", ",")#1000,00
8
9 salariodia = (salario/30)/horas
10 salarioDiaStr = str("%.2f"%salariodia)#1000.00
11 salarioDiaReal = salarioDiaStr.replace(".", ",")#1000,00
12
13 print(
14     "Nome: %s \n Idade: %03d \n Salario: %s \n Salario D
15     %(nome,idade,salarioReal, salarioDiaReal)
16 )
17
```

Ln: 17 Col: 0

Dicas do Programador



Agora é só clicar no Joinha, NÃO
CONVERSAR COM O
COLEGUINHA e PRESTAR
ATENÇÃO NA AULA!



Estruturas Condicionais

Executar ou não executar? Eis a questão...



if

Em VisuAlg nossa estrutura condicional era a **SE**, em Python, utilizaremos o **if**:

► Listagem 4.1 – Formato da estrutura de condicional if

```
if <condição>:  
    bloco verdadeiro
```



Podemos utilizar os operadores relacionais

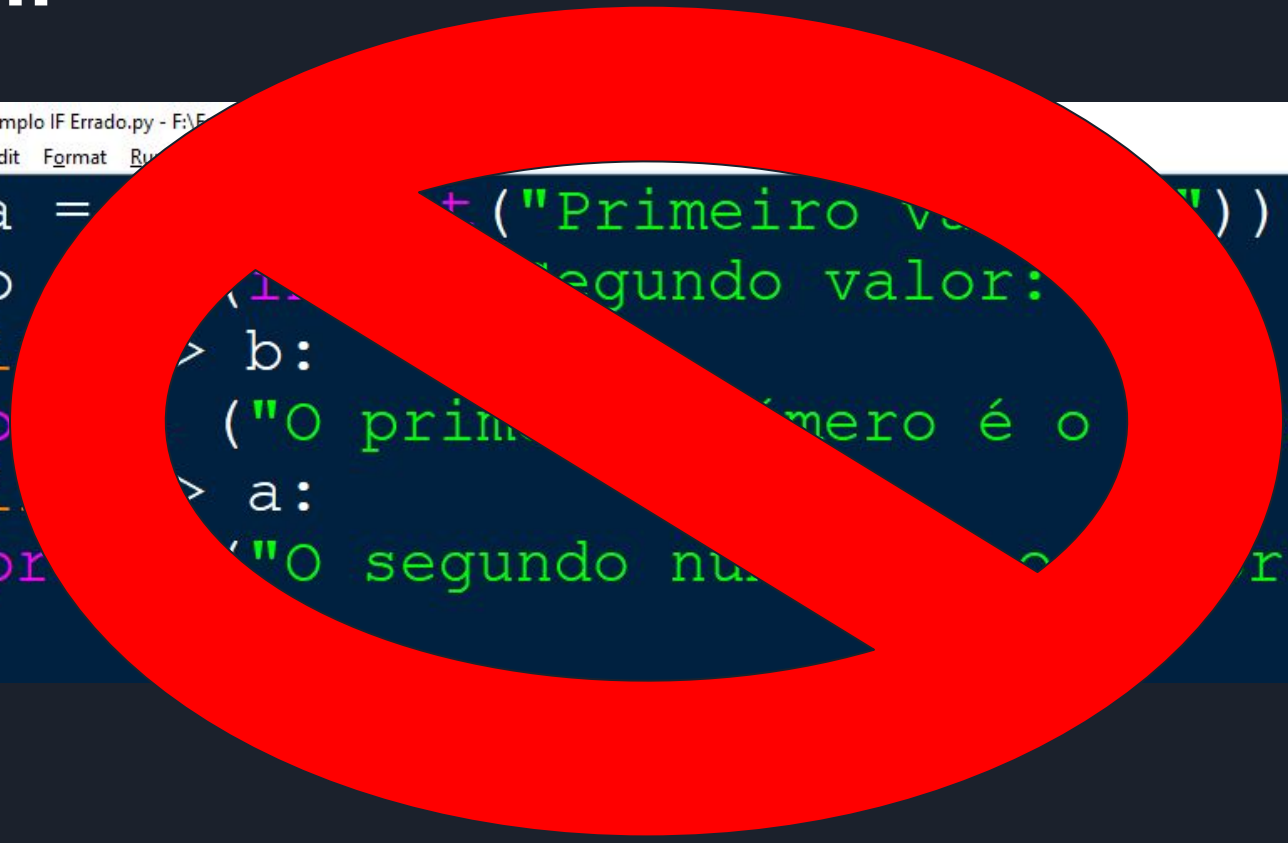
Operador	Operação	Símbolo matemático
==	igualdade	=
>	maior que	>
<	menor que	<
!=	diferente	≠
>=	maior ou igual	≥
<=	menor ou igual	≤



Também os operadores lógicos:

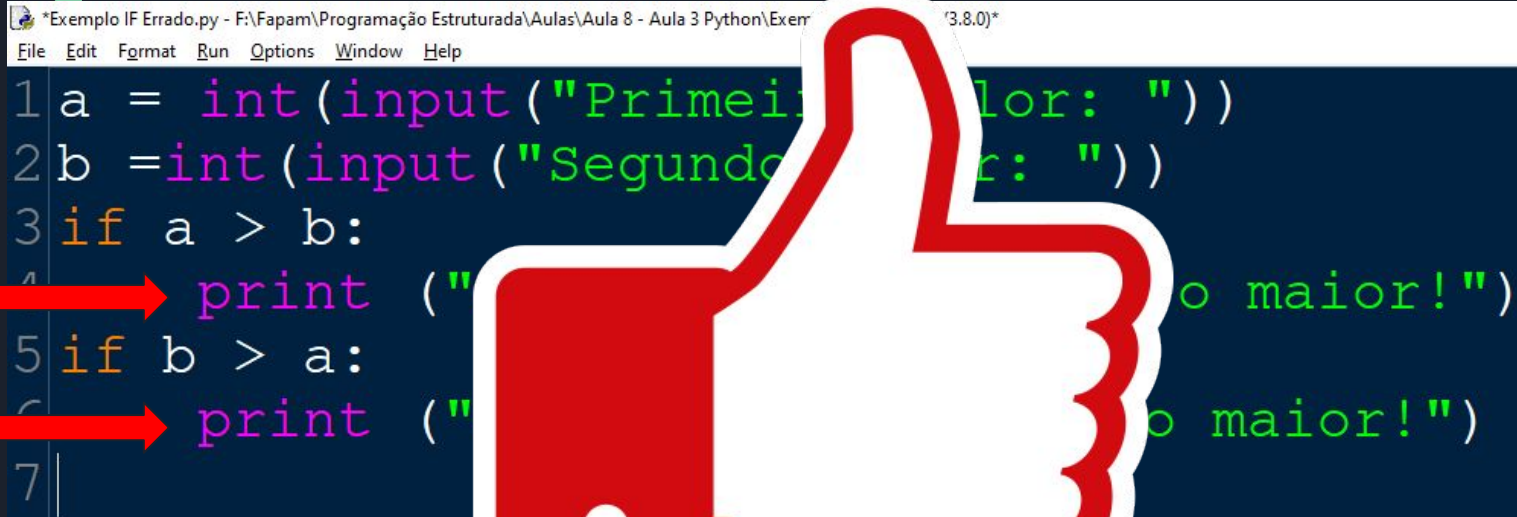
Operador Python	Operação
not	não
and	e
or	ou

if



```
*Exemplo IF Errado.py - F:\E
File Edit Format Run
1 a = 1 + ("Primeiro valor: ")
2 b = 1 + ("segundo valor: ")
3 if b:
4     print("O primeiro número é o ", a, "r!")
5 if a:
6     print("O segundo número é o ", b, "r!")
7
```

if



```
*Exemplo IF Errado.py - F:\Fapam\Programação Estruturada\Aulas\Aula 8 - Aula 3 Python\Exem (3.8.0)*
File Edit Format Run Options Window Help
1 a = int(input("Primeiro valor: "))
2 b = int(input("Segundo valor: "))
3 if a > b:
4     print("O primeiro valor é o maior!")
5 if b > a:
6     print("O segundo valor é o maior!")
7
```

INDENTAÇÃO É OBRIGATÓRIA!



TRÍVIA

Python é uma das poucas linguagens de programação que utiliza o deslocamento do texto à direita (recuo) para marcar o início e o fim de um bloco. Outras linguagens contam com palavras especiais para isso, como BEGIN e END, em Pascal; ou as famosas chaves ({ e }), em C, Java, PHP.



Exercício


Altere o software do exemplo acima e de modo que imprima uma mensagem caso os números sejam iguais.

Exercício

Exemplo IF 2.py - F:/Fapam/Programação Estruturada/Aulas/Aula 8 - Aula 3 Python/Exemplo IF 2.py (3.8.0)

File Edit Format Run Options Window Help

```
a = int(input("Primeiro valor: "))
b = int(input("Segundo valor: "))
if a > b:
    print ("O primeiro número é o maior!")
if b > a:
    print ("O segundo número é o maior!")
if b == a:
    print ("Os Números são Iguais!")
```



Podemos utilizar condições dentro de condições ou vários testes em uma mesma condicional.

Imaginem um software que receba duas palavras de um usuário e depois imprima na tela se elas tem a mesma quantidade de letras e se elas são iguais.


```
a = input("Digite a Primeira String:\n")
b = input("Digite a Segunda String:\n")

if(len(a)==len(b)) :
    print("%s e %s tem o mesmo Tamanho: %d"%(a,b,len(a)))

    if(a==b) :
        print("%s = %s"%(a,b))

if(len(a) !=len(b)) :
    print("%s e %s não tem o mesmo Tamanho."%(a,b))
```



Exercício

Desenvolva um software que leia três números e que imprima o maior deles.



Exercício

```
a = int(input("Primeiro valor: "))
b = int(input("Segundo valor: "))
c = int(input("Terceiro valor: "))
if a > b:
    if a > c:
        print ("O número %d é o maior!"%a)
if b > a:
    if b > c:
        print ("O número %d é o maior!"%b)
if c > b:
    if c > a:
        print ("O número %d é o maior!"%c)
```



Utilizando operadores Lógicos:

```
a = int(input("Primeiro valor: "))
b = int(input("Segundo valor: "))
c = int(input("Terceiro valor: "))
if (a > b) and (a > c):
    print ("O número %d é o maior!"%a)
if (b > a) and (b > c):
    print ("O número %d é o maior!"%b)
if (c > a) and (c > b):
    print ("O número %d é o maior!"%c)
```



Exercício

Desenvolva um software que pergunte o salário do funcionário e calcule o valor do aumento. Para salários superiores a R\$ 1250,00, calcule um aumento de 10%. Para os inferiores ou iguais, de 15%

```
salario = float(input("Digite o Salário: "))
if (salario > 1250):
    salarioNovo = salario*1.1
    aumento = 10

if (salario <= 1250):
    salarioNovo = salario*1.15
    aumento = 15

salarioNovoStr = str("%.2f"%salarioNovo)
salarioNovoReal = salarioNovoStr.replace(".", ",")

print ("Você teve um aumento de %d%% "%aumento)
print ("Seu novo salário é R$ %s "%salarioNovoReal)
```



else

Conforme estudamos em VisuAlg, as estruturas SE poderiam ter as estruturas SENAO, que basicamente são executadas sempre que a estrutura SE não for executada. Em Python temos a estrutura **else** que funciona similar ao nosso SENAO.



else

If-Else_Maior-Idade.py - F:/Fapam/Programação Estruturada/Aulas/Aula 8 - Aula 3 Python/If-Else_Maior-Idade.py (3.8.0)

File Edit Format Run Options Window Help

```
idade = int(input("Digite sua Idade:\n"))
if(idade >= 18):
    print("Maior de Idade")
else:
    print("Menor de Idade")
```



Exercício


Desenvolva um software que pergunte a distância que um passageiro deseja percorrer em km e calcule o preço da passagem, cobrando R\$ 0,50 por km para viagens de até de 200 km, e R\$ 0,45 para viagens mais longas.

```
1 distancia = float(input("Digite a Distância: "))
2 if (distancia > 200):
3     valor = distancia*0.45
4
5 else:
6     valor = distancia*0.5
7
8 valorStr = str("%.2f"%valor)
9 valorReal = valorStr.replace(".",",")
10
11 print ("O Valor da Corrida é %s "%valorReal)
12
```



Estruturas aninhadas:

Nem sempre nossos programas serão tão simples. Muitas vezes, precisaremos aninhar vários if para obter o comportamento desejado do programa. Aninhar, nesse caso, é utilizar um if dentro de outro.



Vejamos o exemplo de calcular a conta de um telefone celular da empresa Tchau. Os planos da empresa Tchau são bem interessantes e oferecem preços diferenciados de acordo com a quantidade de minutos usados por mês. Abaixo de 200 minutos, a empresa cobra R\$ 0,20 por minuto. Entre 200 e 400 minutos, o preço é de R\$ 0,18. Acima de 400 minutos, o preço por minuto é de R\$ 0,15. O programa da listagem 4.6 resolve esse problema.

► Listagem 4.6 – Conta de telefone com três faixas de preço

```
minutos=int(input("Quantos minutos você utilizou este mês:"))  
if minutos < 200: ❶  
    preço = 0.20 ❷  
else:  
    if minutos < 400: ❸  
        preço = 0.18 ❹  
    else: ❺  
        preço = 0.15 ❻  
print("Você vai pagar este mês: R$%6.2f" % (minutos * preço))
```

Vejamos, por exemplo a situação em que cinco categorias são necessárias. Façamos um programa que leia a categoria de um produto e determine o preço pela tabela 4.1.

Tabela 4.1 – Categorias de produto e preço

Categoria	Preço
1	10,00
2	18,00
3	23,00
4	26,00
5	31,00

► Listagem 4.7 – Categoria x preço

```
1 categoria = int(input("Digite a categoria do produto:"))
2 if categoria == 1:
3     preço = 10
4 else:
5     if categoria == 2:
6         preço = 18
7     else:
8         if categoria == 3:
9             preço = 23
10        else:
11            if categoria == 4:
12                preço = 26
13            else:
14                if categoria == 5:
15                    preço = 31
16                else:
17                    print("Categoria inválida, digite um valor entre 1 e 5!")
18                    preço = 0
19 print("O preço do produto é: R${:.2f}" % preço)
```




elif

Python apresenta uma solução muito interessante ao problema de múltiplos ifs aninhados. A cláusula elif substitui um par else if, mas sem criar outro nível de estrutura, evitando problemas de deslocamentos desnecessários à direita.

Vamos revisar o problema da listagem 4.7, dessa vez usando elif. Veja o resultado no programa da listagem 4.8.

► Listagem 4.8 – Categoria x preço, usando elif

```
categoria = int(input("Digite a categoria do produto:"))  
  
if categoria == 1:  
    preço = 10  
  
elif categoria == 2:  
    preço = 18  
  
elif categoria == 3:  
    preço = 23  
  
elif categoria == 4:  
    preço = 26  
  
elif categoria == 5:  
    preço = 31  
  
else:  
    print("Categoria inválida, digite um valor entre 1 e 5!")  
    preço = 0  
  
print("O preço do produto é: R${:.2f}" % preço)
```



Exercício

Desenvolva um software que leia dois números e que pergunte qual operação você deseja realizar. Você deve poder calcular a soma (+), subtração (-), multiplicação (*) e divisão (/). Exiba o resultado da operação solicitada.



Exercício

Desenvolva um software para aprovar o empréstimo bancário para compra de uma casa. O programa deve perguntar o valor da casa a comprar, o salário e a quantidade de anos a pagar. O valor da prestação mensal não pode ser superior a 30% do salário. Calcule o valor da prestação como sendo o valor da casa a comprar dividido pelo número de meses a pagar.



Exercício


Desenvolva um software de lanchonete que apresente para o cliente um menu com ao menos 4 opções de salgado para que ele escolha, depois apresente ao menos 4 opções de bebidas. Ao final, o software deve mostrar na tela o valor total da compra e os itens comprados.



Exercício

Desenvolva um software para analisar se um aluno foi aprovado. O software deve receber 4 notas (entre 0 e 25) e ao final atestar as seguintes possibilidades:

- Se o total for maior ou igual 80 - ALUNO APROVADO - EXCELENTE
- Se o total for entre 60 e 79 - ALUNO APROVADO
- Se o total for entre 40 e 59 - ALUNO EM RECUPERAÇÃO
- Se o total for menor que 40 - ALUNO REPROVADO
- Se alguma das notas estiver fora do intervalo (entre 0 e 25) o software deve exibir a mensagem de “Valor inválido” e encerrar.



Exercício 4.10 Escreva um programa que calcule o preço a pagar pelo fornecimento de energia elétrica. Pergunte a quantidade de kWh consumida e o tipo de instalação: R para residências, I para indústrias e C para comércios. Calcule o preço a pagar de acordo com a tabela a seguir.

Preço por tipo e faixa de consumo		
Tipo	Faixa (kWh)	Preço
Residencial	Até 500	R\$ 0,40
	Acima de 500	R\$ 0,65
Comercial	Até 1000	R\$ 0,55
	Acima de 1000	R\$ 0,60
Industrial	Até 5000	R\$ 0,55
	Acima de 5000	R\$ 0,60



Bibliografia

- MENEZES, Nilo Ney Coutinho - Introdução à Programação com Python: Algoritmos e Lógica de Programação Para Iniciantes, 3ª Edição – 2019, Editora: Novatec Editora, ISBN-10: 8575227181
- SHAW, Zed A – Aprenda Python 3 do jeito certo, 1ª Edição – 2019, Editora: Alta Books, ISBN: 978-85-508-0473-6.
- <https://docs.python.org/pt-br/3/>
- https://www.ime.usp.br/~leo/mac2166/2017-1/introducao_estrutura_basica_c_python.html
- <http://python42.com.br/?p=176>
- <https://www.youtube.com/@CursoemVideo>
- <https://panda.ime.usp.br/cc110/static/cc110/>
-