

Objetivos da prática

- Descrever a utilização das estruturas de condição if e else em Python;
- Descrever a utilização da estrutura de condição else if (elif) em Python;
- Descrever a utilização da estrutura de repetição while em Python;
- Descrever a utilização da estrutura de repetição for em Python;
- Descrever a utilização de funções em Python;
- Descrever a utilização argumentos de funções no Python;
- Refazer a calculadora utilizando estrutura condicional e funções;
-

Atividades práticas:

Microatividade 1: Descrever a utilização das estruturas de condição if e else em Python:

1. Crie uma variável chamada temperatura e atribua a ela o valor 29
2. Crie uma verificação, utilizando a condição if, para checar se o valor da variável temperatura é menor que 30
3. Caso positivo, imprima na tela o texto 'A temperatura hoje está amena'
4. Caso contrário, e utilizando a condição else, imprima na tela o texto 'Hoje está fazendo calor'

Nome do arquivo:estruturas_condicao1.py

```
temperatura = 29
if temperatura < 30:
    print('A temperatura hoje está amena')
else:
    print('Hoje está fazendo calor')
```

Microatividade 2: Descrever a utilização da estrutura de condição else if (elif) em Python:

1. Crie uma variável chamada tempoExperiencia e atribua a ela o valor 5
2. Crie uma verificação, utilizando a condição if, para checar se o valor da variável tempoExperiencia é menor que 2;
3. Caso positivo, imprima na tela o texto 'Nível de conhecimento júnior.';
4. Após as instruções acima, crie uma outra condição utilizando elif (else if) para verificar se o valor da variável tempoExperiencia é maior que 2 e menor que 5.
5. Em caso positivo, imprima o texto 'Nível de conhecimento pleno.'
- # 6. Por fim, crie uma condição else e imprima o texto 'Nível de conhecimento sênior.';

Nome do arquivo: estruturas_condicao2.py

```
tempoExperiencia =5
if tempoExperiencia < 2:
    print('Nível de conhecimento júnior.')
elif tempoExperiencia >= 2 and tempoExperiencia < 5:
    print('Nível de conhecimento pleno.')
else:
    print('Nível de conhecimento sênior.')
```

Microatividade 3: Descrever a utilização da estrutura de repetição while em Python:

1. Crie uma variável chamada entrada_idade e atribua a ela o valor “;
 2. Crie uma instrução while que verifique se o valor atribuído à variável entrada_idade é diferente de 0 (como o valor inicial atribuído à variável é ”, isso a definiu como tipo string. Logo, a verificação no While deve ser feita com auxílio da instrução str);
 3. No escopo da instrução while, atribua à variável entrada_idade um input de entrada de dados com o texto ‘Digite um número qualquer ou 0 para sair: ’;
 4. Imprima, na tela, o número digitado pelo usuário precedido do texto ‘Número digitado: ’;
- A impressão deve ocorrer APENAS se o valor não for '0', para evitar a impressão final.

Nome do arquivo:estruturas_repeticao1.py

```
entrada_idade = ""  
while entrada_idade != '0':  
    entrada_idade = input('Digite um número qualquer ou 0 para sair: ')  
    if entrada_idade != '0':  
        print(f'Número digitado: {entrada_idade}')
```

Microatividade 4: Descrever a utilização da estrutura de repetição for em Python:

1. Crie uma variável chamada texto e atribua a ela o valor ‘Olá, laço for.’;
2. Crie uma instrução for que itere sobre a variável texto atribuindo cada um de seus caracteres a uma variável chamada item;
3. Imprima, na tela, dentro do escopo do laço for, o valor da variável item precedido do texto ‘Caractere: ’;
4. Crie, no mesmo script, uma nova instrução for que: Itere sobre um intervalo numérico entre 1 e 10 (dica: use a instrução range); range(1, 11) gera números de 1 até 10
5. Imprima, na tela, dentro do escopo do laço for, o valor de cada número no intervalo acima precedido do texto ‘Número do intervalo: ’ ;

Lembre-se de utilizar a instrução str para concatenar o valor inteiro com a string no momento de imprimir o valor pedido na tela.

Nome do arquivo: estruturas_repeticao2.py

```
texto = 'Olá, laço for.'  
print('--- Primeiro Loop (Iterando sobre a string) ---')  
for item in texto:  
    print('Caractere: ' + item)  
print('\n--- Segundo Loop (Iterando sobre o intervalo numérico) ---')  
for numero in range(1, 11):  
    print('Número do intervalo: ' + str(numero))
```

Microatividade 5: Descrever a utilização de funções em Python:

1. Defina uma função chamada “imprimir_variavel”. Tal instrução não receberá parâmetros;
2. No escopo da função acima, crie uma variável chamada texto e atribua a ela o valor ‘Olá, funções em Python’;
3. A seguir, ainda no escopo da função, imprima na tela o valor da variável texto;
4. Por último, fora do escopo da função, faça a chamada da mesma.

Nome do arquivo: funcoes1.py

```
def imprimir_variavel():  
    texto = 'Olá, funções em Python'  
    print(texto)  
imprimir_variavel()
```

Microatividade 6: Descrever a utilização argumentos de funções no Python:

1. Defina uma função chamada “loginUsuario”. Tal instrução receberá como parâmetro a variável perfil;
2. No escopo da função, verifique se o valor do parâmetro perfil é igual a ‘admin’.

Dica: utilize a instrução lower no momento de fazer a verificação;

Caso o valor do parâmetro seja igual a ‘admin’, imprima na tela: ‘Bem-vindo, Administrador’. Do contrário, imprima: ‘Bem-vindo, Usuário’;

3. Fora do escopo da função, faça a chamada da mesma passando diferentes valores como parâmetro.
- 4 Primeira chamada: ‘Admin’

Nome do arquivo: funcoes2.py

```
def loginUsuario(perfil):
    perfil='admin'
    if perfil.lower() == 'admin':
        print('Bem-vindo, Administrador')
    else:
        print('Bem-vindo, Usuário')
loginUsuario('usuário')
```

Atividade da Calculadora

Essa atividade não fiz a primeira versão. Fiz a mudança de curso e minhas matérias demorou para ser liberadas. O primeiro trabalho não fiz por que o prazo já havia expirado.

1. Crie uma variável chamada saída e atribua a ela o valor ”;
2. Crie uma função chamada adicao. Tal função deverá receber dois parâmetros e retornar a soma entre ambos;
3. Crie uma função chamada subracao. Tal função deverá receber dois parâmetros e retornar a subtração entre ambos;
4. Crie uma função chamada multiplicacao. Tal função deverá receber dois parâmetros e retornar a multiplicação entre ambos;
5. Crie uma função chamada divisao. Tal função deverá receber dois parâmetros, verificar se um deles é igual a 0. Em caso positivo, deverá retornar a mensagem “Não foi possível realizar a divisão por 0”. Em caso negativo, deverá retornar a divisão entre ambos;
6. Crie uma função chamada calculadora. Tal função deverá receber três parâmetros, sendo eles: os dois números que serão usados para os cálculos e a operação matemática que se deseja realizar. No corpo da função calculadora você deverá verificar qual a operação desejada pelo usuário, checando o valor do parâmetro correspondente. Utilize estruturas de condição para isso e, dependendo da operação desejada, você deverá chamar a função relativa a ela, passando as variáveis contendo os dois números para serem utilizados no cálculo.

Armazene o resultado da chamada às funções de cálculo numa variável chamada resultado. Ao final da função calculadora você deverá retornar a variável resultado;

O loop while será implementado na próxima fase.

O script será salvo agora.

7. Crie um laço while e, como condição do mesmo, verifique se o valor da variável saída é diferente de n. Lembre-se de que o usuário poderá inserir tanto N quanto n;

8. No escopo do laço while peça ao usuário para digitar o primeiro número e armazene seu valor numa variável. Faça o mesmo para o segundo número e para a operação matemática.

Passe essas três variáveis para o método calculadora, armazenando o retorno dessa chamada numa variável também chamada resultado.

Imprima na tela o valor da variável resultado precedido pelo texto ‘Resultado da operação: ’.

Por fim, pergunte ao usuário se ele deseja continuar ou não executando o programa.

Armazene tal input na variável saida;

9. Tome cuidado com a condição de verificação do laço for em relação à entrada do usuário armazenada na variável saida. Use, por exemplo, S/N.

Nome do arquivo:calculadora_v2.py

```
saida = ''  
def adicao(num1, num2):  
    return num1 + num2  
def subtracao(num1, num2):  
    return num1 - num2  
def multiplicacao(num1, num2):  
    return num1 * num2  
def divisao(num1, num2):  
    if num2 == 0:  
        return "Não foi possível realizar a divisão por 0"  
    return num1 / num2  
def calculadora(num1, num2, operacao):  
    operacao = operacao.lower()  
    if operacao == '+' or operacao == 'adicao':  
        resultado = adicao(num1, num2)  
    elif operacao == '-' or operacao == 'subtracao':  
        resultado = subtracao(num1, num2)  
    elif operacao == '*' or operacao == 'multiplicacao':  
        resultado = multiplicacao(num1, num2)  
    elif operacao == '/' or operacao == 'divisao':  
        resultado = divisao(num1, num2)  
    else:  
        resultado = "Operação inválida"  
    return resultado  
while saida.lower() != 'n':  
    print('\n--- CALCULADORA INTERATIVA ---')  
    try:  
        num1 = float(input('Digite o primeiro número: '))  
        num2 = float(input('Digite o segundo número: '))  
    except ValueError:  
        print("Entrada inválida. Por favor, digite apenas números.")  
        continue  
    operacao = input('Digite a operação (+, -, *, / ou nome): ')  
    resultado = calculadora(num1, num2, operacao)  
    print(f'Resultado da operação: {resultado}')  
    saida = input('Deseja continuar? (S/N): ')  
print('Programa encerrado.')
```