

CENTRO UNIVERSITÁRIO



# Fundamentos de Python – Parte 2

## Prof. Me. José Carlos Perini

# Contatos do professor

- José Carlos **Perini**
- E-mail: [jose.perini@unimetrocamp.edu.br](mailto:jose.perini@unimetrocamp.edu.br)
- Blog: <http://profperini.com>
- E-mail pessoal: [profjoseperini@gmail.com](mailto:profjoseperini@gmail.com)
- Canal do youtube: [www.youtube.com/profperini](http://www.youtube.com/profperini)
- Site: <http://about.me/profperini>

# Estrutura de Repetição Enquanto

- Uma estrutura de repetição **enquanto** pode ser utilizada quando o algoritmo precisa **testar determinada condição antes de executar um conjunto de comandos** repetidas vezes
- Se a condição avaliada for verdadeira, o conjunto de comandos dentro da estrutura de repetição **enquanto** é executado e após esta execução, a condição é novamente avaliada
- Se o resultado da avaliação for falso, este conjunto de comandos não será executado e o fluxo do algoritmo segue normalmente.
- Nesta estrutura de repetição, pode ocorrer do conjunto de comando não ser executado nenhuma vez.

# Estrutura de Repetição Enquanto - Python

- Sintaxe da Estrutura de Repetição enquanto

*<inicialização da variável de controle>;*

***while*** *<condição>* :

*<comando\_1>;*

*<comando\_2>;*

...

*<comando\_n>;*

*<atualização da variável de controle>;*

# Estrutura de Repetição Enquanto - Python

- Exemplo

```
#Exemplo while
```

```
x = 0
while x < 3 :
    print('O valor de x é: %d' % x)
    x = x + 1
print('Saiu do while')
```

# Exercícios

1. Desenvolva um programa que recebe números inteiros digitados pelo usuário e calcula a soma entre esses números e a média. Só parar de digitar os números quando o usuário digitar zero.

```
# Desenvolva um programa que recebe números inteiros digitados
# pelo usuário e calcula a soma entre esses números e a média.
# Só parar de digitar os números quando o usuário digitar zero.

numero = 5
cont = 0
soma = 0

while numero != 0 :
    numero = int(input('Digite um número: '))
    if numero != 0 :
        soma = soma + numero
        cont = cont + 1
# saiu do laço
media = soma / cont
print ('A soma é igual a %d e a média é igual a %.2f' % (soma, media))
```

# Exemplo de while com if

```
while True :  
    msg = input('Digite \'end\' para sair, qualquer tecla para continuar: ')  
    if 'end' in msg:  
        print ('Saindo...')  
        break  
    if 'continue':  
        print ('Continuando...')  
print('Fim do programa')
```



# Estrutura de Repetição Para

- Uma estrutura de repetição **para** pode ser utilizada quando o algoritmo precisa ter definido a quantidade de vezes que um conjunto de comandos deve ser executado
- Neste caso, a variável de controle, sua inicialização e finalização bem como sua atualização fazem parte do cabeçalho da estrutura de repetição **para** e o conjunto de comandos dentro da estrutura de repetição **para** é executado a quantidade de vezes determinado no cabeçalho desta estrutura
- Note que nesta estrutura de repetição, pode ocorrer do conjunto de comandos não ser executado nenhuma vez

# Estrutura de Repetição for em Python

- Exemplo

```
# Primeiro Exemplo de for
```

```
x = 0
```

```
for x in range (3) :    # x vai de 0 a 2
    print ('O valor de x é: %d' % x);
print ('Saiu do laço')
```

# Estrutura de Repetição for em Python

- Segundo exemplo

```
# Segundo exemplo de for

for x in range (50, 100) : # x vai de 50 a 99
    if x == 88 :
        break # se x for igual a 88, sai do laço
    print(x)
print('Saiu do for')
```

# Estrutura de Repetição for em Python

- Terceiro exemplo

```
# Terceiro exemplo com for - soma dos números ímpares
total = 0
numero = int(input('Digite um número: '))
if (numero % 2) == 0:
    numero = numero -1
for i in range(numero,0,-2):      # i vai de numero até 0, decrementando de 2
    total = total + i
    print('Valor de i %d:' % i)
print("A soma dos números ímpares é %d " % total)
```

# Exercícios

- 1 - Desenvolva um programa que calcule e o quadrado dos números inteiros compreendidos entre 10 e 150. **Utilizar for.**
- 2 - Desenvolva um programa que receba um número inteiro, calcule e mostre o seu fatorial. (Exemplo de Fatorial: se o número 4 for digitado, o programa deverá fazer  $1*2*3*4$  e mostrar como resultado 24, se o número digitado for 5 o programa deverá fazer  $1*2*3*4*5$  e mostrar como resultado 120). **Utilizar for.**

```
# Desenvolva um programa que calcule e o
# quadrado dos números inteiros compreendidos entre 10 e 150.
# Utilizar for.
```

```
for i in range (10, 151) :
    print ('%d ao quadrado = %d' % (i, i*i))
```

```
# Desenvolva um programa que receba um número inteiro,
# calcule e mostre o seu fatorial. (Exemplo de Fatorial:
# se o número 4 for digitado, o programa deverá fazer 1*2*3*4 e
# mostrar como resultado 24,
# se o número digitado for 5 o programa deverá fazer 1*2*3*4*5 e
# mostrar como resultado 120). Utilizar for.
```

```
numero = int(input('Digite um número: '))
f = 1
# indo de 1 até numero
for i in range (1,numero+1) :
    f = f * i
print('Fatorial de %d: %d' % (numero, f))
f = 1
# indo de numero até 1, decrementando em 1
for i in range (numero, 0, -1) :
    f = f * i
print('Fatorial de %d: %d' % (numero, f))
```

# FUNÇÕES

CENTRO UNIVERSITÁRIO

UNI  
METROCAMP

WYDEN

# Funções em Python

- Na programação, funções são blocos de código que realizam determinadas tarefas que normalmente precisam ser executadas diversas vezes dentro de uma aplicação.
- Quando surge essa necessidade, para que várias instruções não precisem ser repetidas, elas são agrupadas em uma função, à qual é dado um nome e que poderá ser chamada/executada em diferentes partes do programa.



# Funções em Python

A sintaxe de uma função é definida por três partes: nome, parâmetros e corpo, o qual agrupa uma sequência de linhas que representa algum comportamento.

# Funções em Python

No código abaixo, temos um exemplo de **declaração de função em Python**:

```
#função hello:
def hello(meu_nome):
    print('Olá', meu_nome)
nome = input('Digite seu nome: ')

#chamada da função hello:
hello(nome)
```

# Funções em Python

No código abaixo, temos um exemplo de **declaração de função em Python com 2 parâmetros**:

```
#função hello:  
def hello(seu_nome, sua_idade):  
    print('Olá', seu_nome, '\nvocê tem', sua_idade, 'anos')  
nome = input('Digite seu nome: ')  
idade = int(input('Digite sua idade: '))
```

```
#chamada da função hello:  
hello(nome, idade)
```

CENTRO UNIVERSITARIO

UNI  
METROCAMP

WYDEN

UNI  
METROCAMP

WYDEN

# Funções em Python

Exemplo de função com retorno de valor:

```
def calcular_pagamento(horas, valor_hora):  
    if horas <= 40:  
        salario=horas*valor_hora  
    else:  
        h_excd = horas - 40  
        salario = 40*valor_hora+(h_excd*(1.5*valor_hora))  
    return salario  
qtd_horas = float(input('Digite a quantidade de horas: '))  
valor = float(input('Digite o valor da hora trabalhada: '))  
#chamada da função:  
salario = calcular_pagamento(qtd_horas, valor)  
print('Salário: %.2f ' % salario)
```

# Exemplo

- Fazer um programa que entra com dois números inteiros e apresenta os resultados da soma, subtração, multiplicação e divisão entre eles. Fazer uma função para cada operação.

# Funções

```
#funções:  
def soma(n1, n2):  
    return n1 + n2  
def subtracao(n1, n2):  
    return n1 - n2  
def multiplicacao(n1, n2):  
    return n1 * n2  
def divisao(n1, n2):  
    return n1 / n2
```

# Chamadas das funções

```
#chamadas das funções:
print('\n%d + %d = %d' % (num1, num2, soma(num1, num2)))
print('%d - %d = %d' % (num1, num2, subtracao(num1, num2)))
print('%d x %d = %d' % (num1, num2, multiplicacao(num1, num2)))
print('%d / %d = %.2f' % (num1, num2, divisao(num1, num2)))

#utilizando .format
print('\nRespostas com format')
print('{:d} + {:d} = {:d}'.format(num1, num2, soma(num1,num2)))
print('{:d} - {:d} = {:d}'.format(num1, num2, subtracao(num1,num2)))
print('{:d} x {:d} = {:d}'.format(num1, num2, multiplicacao(num1,num2)))
print('{:d} / {:d} = {:.2f}'.format(num1, num2, divisao(num1,num2)))
```

# Exercícios

1. Faça um programa que recebe um número inteiro e através de um módulo função calcula e mostra se o número é primo ou não.
2. Faça um programa que receba um número inteiro e através de um módulo função calcule e mostre se este número é par ou ímpar.
3. Faça um programa que recebe dois números reais e através de um módulo procedimento calcula e mostra a potenciação do primeiro número pelo segundo e a divisão do primeiro número pelo segundo.
4. Desenvolver em Python um programa que tenha um módulo função que apresenta um menu de opções e retorna a opção escolhida. Opções: 1 - Fatorial do número; 2 - Tabuada do número; 3 - Mostrar se o número é par ou ímpar; 4 - Sair do programa. Fazer os módulos correspondentes aos cálculos e as chamadas dos mesmos.



# Exercício 1

#1. Faça um programa que recebe um número inteiro e através de um módulo função calcula e mostra se o número é primo ou não.

```
def primo (num) :
    cont=0
    for x in range(2, num) :
        if num % x == 0 :
            cont+=1
            break

    if cont==0 :
        return True
    else:
        return False

numero = int(input('Digite um número: '))
prim = primo (numero)
if prim :
    print ('{} é primo' .format(numero))
else :
    print ('{} não é primo' .format(numero))
```

# Exercício 4 (cont.)

```
num1 = int(input('Digite o primeiro número: '))
opcao = -1
while opcao != 4 :
    opcao = menu()
    if opcao == 1 :
        fatorial(num1)
    elif opcao == 2 :
        tabuada(num1)
    elif opcao == 3 :
        par_impar(num1)
    elif opcao == 4 :
        print('Saindo do programa. obrigado!')
    else :
        print('Opção inválida')
```

# Canal do Professor

- Você já se inscreveu no canal do professor?
- Lá você encontra muitos assuntos importantes sobre programação.
- <http://youtube.com/profperini>



**PROGRAMANDO COM O PROF. PERINI**

Vídeo novo toda semana.

Instagram Facebook

 Programando com o Prof. Perini  
252 inscritos

PERSONALIZAR O CANAL

YOUTUBE STUDIO (BETA)



Faci facid

FACIMP

F&V

fmg  
Fundação  
Martha Falcão

ISL

UNIFAVIP

UNI  
METROCAMP

RUY  
BARBOSA

AREA1

UniF&V

UniFanor