Fatec
Rio Claro

Linguagem de Programação I

Prof. Orlando Saraiva Júnior
orlando.nascimento@fatec.sp.gov.br

A abstração é a chave para lidar com a complexidade dos sistemas.

Edsger W. Dijkstra

Funções

Funções

Fatec Rio Claro

Podemos criar nossas próprias funções em Python. Já sabemos algumas funções, como print, input, float, int, dentre outras.

Para definir uma nova função, se usa a instrução **def**

```
def soma(a, b):
    print(a + b)
soma(2, 9)
soma(7, 8)
```

Funções

Fatec
Rio Claro

Funções são especialmente interessantes para isolar uma tarefa específica em um trecho do programa. Isso pemite que a solução de um problema seja reutilizada em outras partes do programa, sem precisar repetir as mesmas linhas.

Funções

Fatec Rio Claro

A instrução **return** serve para indicar o que deve ser retornado da função. Veja o exemplo:

```
def soma(a, b):
    return (a + b)
print(soma(2, 9))
print(soma(7, 8))
```

Atividade

Fatec
Rio Claro

Realizar as três atividades do **Exercício de Fixação.**

1) Maior valor

2) Múltiplo

3) Quadrado

4) Triângulo

Variáveis locais e globais

Fatec
Rio Claro

Quando usamos funções, começamos a trabalhar com variáveis internas ou locais e variáveis externas ou globais. A diferença entre elas é a visibilidade ou escopo.

Uma variável local a uma função existe apenas dentro dela, sendo inicializada a cada chamada. Assim, não podemos acessar o valor de uma variável local fora da função que a criou.

Variáveis locais

Fatec
Rio Claro

Observe as variáveis locais:

```
def soma(a, b):
    c = a + b
    return c
print(soma(2, 9))
print(soma(7, 8))
```

Variáveis locais

Fatec Rio Claro

Observe as variáveis locais:

```
def soma(a, b):
    c = a + b
    return c
print(soma(2, 9))
print(soma(7, 8))
```

Variáveis globais

Fatec Rio Claro

Uma variável global é definida fora de uma função, podendo ser vista por todas as funções do módulo (programa) e por todos os módulos

```
EMPRESA = 'Fatec Rio Claro'
def imprime_cabecalho():
    print(EMPRESA)
    print("=" * 15)
imprime_cabecalho()
```

Variáveis globais

Fatec Rio Claro

Uma variável global é definida fora de uma função, podendo ser vista por todas as funções do módulo (programa) e por todos os módulos

```
EMPRESA = 'Fatec Rio Claro'
def imprime_cabecalho():
    print(EMPRESA)
    print("=" * 15)
imprime_cabecalho()
```

DICA

Fatec
Rio Claro

Variáveis globais devem ser utilizadas o mínimo possível em seus programas, pois dificultam a leitura e violam o encapsulamento da função.

Encapsulamento é o fato da função conter ou esconder os detalhes de sua operação, de forma que seu funcionamento seja entendido analisando-se apenas os parâmetros de entrada e o código da própria função.

Variáveis globais

Fatec
Rio Claro

Se quisermos modificar uma variável global dentro de uma função, devemos informar que estamos usando uma variável global antes de inicializá-la, na primeira linha de nossa função. Essa modificação é feita com a instrução **global**.

Variáveis locais e globais

Fatec Rio Claro

```
a = 5
def muda_e_imprime()
    global a
    a = 7
    print(f" a dentro da função:  {a} ")
print(f" a antes de mudar:  {a} ")
muda_e_imprime()
print(f" a depois de mudar:  {a} ")
```

Variáveis locais e globais

Fatec Rio Claro

```
a = 5
def muda_e_imprime()
    global a
    a = 7
    print(f" a dentro da função:  {a} ")
print(f" a antes de mudar:  {a} ")
muda_e_imprime()
print(f" a depois de mudar:  {a} ")
```

Funções Recursivas

Fatec Rio Claro

Uma função pode chamar a si mesma. Quando isso ocorre, temos uma função recursiva. O problema fatorial pode demonstrar sua aplicação:

```
def fatorial(n):
    if n == 0 or n == 1:
        return 1
    else:
        return n * fatorial(n - 1)
```

Atividade

Fatec
Rio Claro

O problema de Fibonacci refere-se à sequência de Fibonacci, uma sucessão de números onde cada termo é a soma dos dois anteriores. A sequência começa assim:

0,1,1,2,3,5,8,13,21,34,...

Formalmente, ela é definida como:

$$F(n) = \begin{cases} 0, & \text{se } n = 0 \\ 1, & \text{se } n = 1 \\ F(n-1) + F(n-2), & \text{se } n \geq 2 \end{cases}$$

Atividade

Fatec
Rio Claro

```
def fibonacci(n):
    if n == 0:
        return 0  # Caso base: Fibonacci(0) = 0
    elif n == 1:
        return 1  # Caso base: Fibonacci(1) = 1
    fib_1 = 0  # F(0)
    fib_2 = 1  # F(1)
    for i in range(2, n + 1): # Calculamos a sequência do 2 até n
        proximo_fib = fib_1 + fib_2  # Soma dos dois anteriores
        fib_1 = fib_2  # Atualiza o primeiro número
        fib_2 = proximo_fib  # Atualiza o segundo número
    return fib_2  # Retorna o resultado final
```

Atividade

Fatec
Rio Claro

Reescreva o problema de Fibonacci com uso de recursividade.

Parâmetros opcionais

Fatec Rio Claro

Nem sempre precisamos passar todos os parâmetros para uma função, mas deixando a possibilidade de alterá-lo.

```
def barra(n=40, caracter=”*”)
    print(caracter * n)
barra()
barra(10)
barra(10, ’-’)
```

Parâmetros nomeados

Fatec Rio Claro

Quando especificamos o nome dos parâmetros, podemos passá-lo em qualquer ordem:

```
def retangulo(largula, altura, caracter='*')
    linha = caracter * largura
    for i in range(altura):
        print(linha)
```

Parâmetros nomeados

Fatec Rio Claro

Quando especificamos o nome dos parâmetros, podemos passá-lo em qualquer ordem:

```
def retangulo(largula, altura, caracter='*')
    linha = caracter * largura
    for i in range(altura):
        print(linha)

                retangulo(3, 4)

                retangulo(largura=4, altura=3)

                retangulo(caracter='#',largura=4, altura=3)
```

Fatec
Rio Claro

Dúvidas

Prof. Orlando Saraiva Júnior
orlando.nascimento@fatec.sp.gov.br