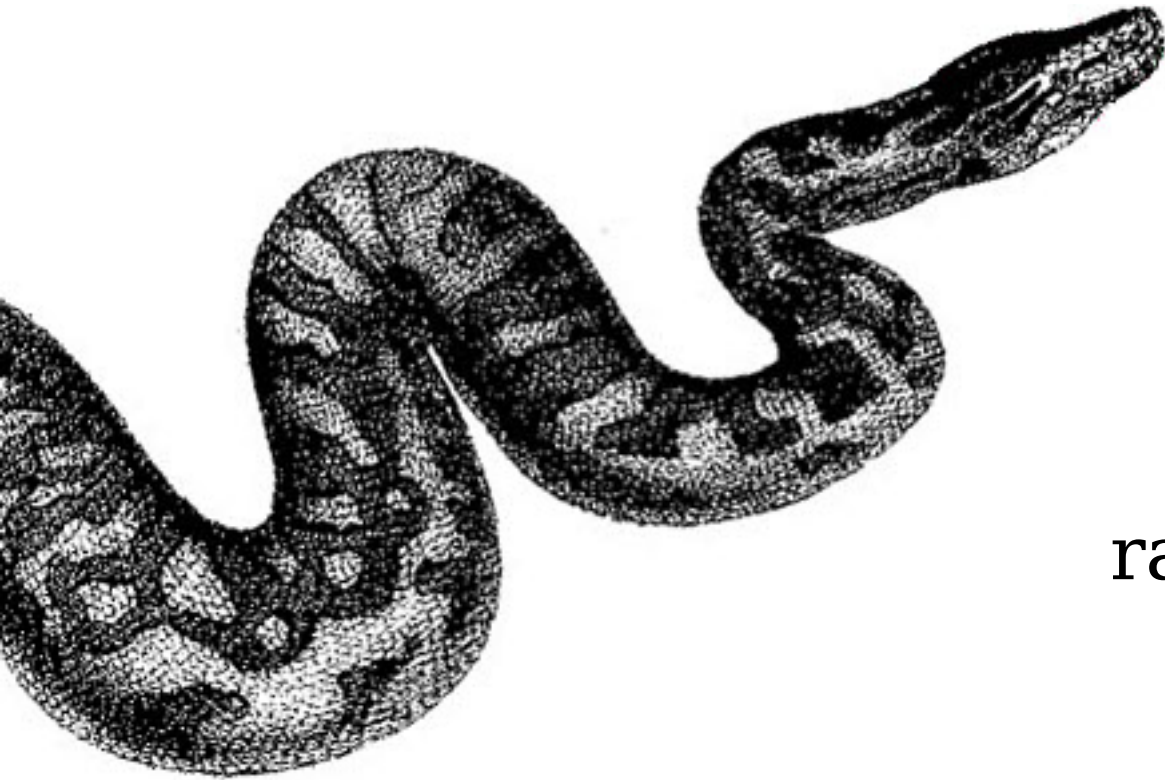


# Python: a primeira mordida



Luciano Ramalho  
[ramalho@python.pro.br](mailto:ramalho@python.pro.br)

# Definição de funções

- Comando **def** inicia a definição
- Comando **return** marca o fim da execução da função e define o resultado a ser retornado

```
def inverter(texto):  
    if len(texto) <= 1:  
        return texto  
    lista = list(texto)  
    lista.reverse()  
    return ' '.join(lista)
```

# Argumentos de funções

- Valores default indicam args. opcionais
  - argumentos obrigatórios devem vir antes de argumentos opcionais
    - `def exibir(texto, estilo=None, cor='preto'):`
- Palavras-chave podem ser usadas para fornecer argumentos fora de ordem
- Como a função acima pode ser invocada:
  - `exibir('abacaxi')`
  - `exibir('abacaxi', 'negrito', 'amarelo')`
  - `exibir('abacaxi', cor='azul')`

# Argumentos arbitrários

- Use `*args` para aceitar uma lista de argumentos posicionais
- Use `**args` para aceitar um dicionário de argumentos identificados por palavras-chave
- Exemplo:
  - `def tag(nome, *linhas, **atributos):`

# Argumentos arbitrários (2)

```
print tag('br')
print tag('img',src='foto.jpg',width=3,height=4)
print tag('a','Wikipédia',
        href='http://wikipedia.org')
print tag('p','Eu não devia te dizer',
        'mas essa lua','mas esse conhaque',
        'botam a gente comovido como o diabo.',
        id='poesia')
```

```
<br />

<a href="http://wikipedia.org">Wikipédia</a>
<p id="poesia">
    Eu não devia te dizer
    mas essa lua
    mas esse conhaque
    botam a gente comovido como o diabo.
</p>
```

# Argumentos arbitrários (3)

```
def tag(nome, *linhas, **atributos):  
    saida = ['<' + nome]  
    for par in atributos.items():  
        saida.append(' %s="%s"' % par)  
    if linhas:  
        saida.append('>')  
        if len(linhas) == 1:  
            saida.append(linhas[0])  
        else:  
            saida.append('\n')  
            for linha in linhas:  
                saida.append('\t%s\n' % linha)  
        saida.append('</%s>' % nome)  
    else:  
        saida.append(' />')  
    return ''.join(saida)
```