

04

## Sintaxe das funções

Douglas está criando uma calculadora simples, para realizar cálculos entre dois números. Para isso, ele criou quatro funções, uma para cada operação matemática (soma, subtração, multiplicação e divisão):

```
def soma_dois_numeros(primeiro_numero, segundo_numero):
    print(primeiro_numero + segundo_numero)

def subtrai_dois_numeros(primeiro_numero, segundo_numero)
    print(primeiro_numero - segundo_numero)

def multiplica_dois_numeros(primeiro_numero, segundo_numero):
    print(primeiro_numero * segundo_numero)

def divide_dois_numeros(primeiro_numero, segundo_numero):
    print(primeiro_numero / segundo_numero)
```

Mas na hora de testá-las, Douglas reparou algo estranho no funcionamento da sua calculadora. Então, das quatro funções declaradas acima, podemos dizer que:

A A função `multiplica_dois_numeros` possui erro de sintaxe.

B A função `divide_dois_numeros` possui erro de sintaxe.

C A função `soma_dois_numeros` possui erro de sintaxe.

 A função `subtrai_dois_numeros` possui erro de sintaxe.

Correto! A função `subtrai_dois_numeros` possui não só um, mas dois erros de sintaxe. Após o nome da função, deve haver parênteses (e não são chaves!), e está faltando os dois pontos após o nome da função.

Quando declaramos uma função, é importante os parênteses logo após o seu nome, é dentro dele que os parâmetros da função ficam. Inclusive toda função possui um bloco, em que podemos adicionar o código que quisermos que a função execute quando for chamada.

Lembre-se que um bloco é caracterizado por tudo que fica após os dois pontos, indentado por quatro espaços. Tudo que estiver indentado por quatro espaços são instruções da função.

[PRÓXIMA ATIVIDADE](#)