

**Unidade 1: Iniciação à programação em Python**

**Aplicação: Os diferentes tipos de dados em Python**

Nesta aplicação da Unidade 1, irá construir alguns programas recorrendo aos conhecimentos adquiridos nas lições desta unidade:

- Funções em linguagem Python
- Criação de variáveis numéricas e cadeias de caracteres

**Objetivos:**

- Criar um conversor de sistemas de medida de temperatura.
- Criar um programa que permita desenvolver uma expressão algébrica

**EXEMPLO:**

**Conversor de Temperatura**

Para medir a temperatura em Portugal, utilizamos como unidade de medida o grau Celsius (°C). Nos países anglo-saxónicos, utilizam como unidade de medida o grau Fahrenheit (°F).

O desafio para este exemplo consiste em programar uma função que permita converter a temperatura entre estas duas unidades de medida, nos dois sentidos: °C ↔ °F .

Existirá uma temperatura igual nas duas unidades?

Consideremos a função que permite converter a temperatura de graus Celsius para graus Fahrenheit:

$$t(^{\circ}F) = \frac{9}{5} \times t(^{\circ}C) + 32.$$

Donde, resulta que podemos utilizar em sentido contrário a função:

$$t(^{\circ}F) = t(^{\circ}C) \times 1.8 + 32$$



**IMPLEMENTAÇÃO DO EXEMPLO:**

- Abrir a aplicação TI-Python e começar um novo programa.
- Nomear o programa como **Temperatura** e validar pressionando a tecla `enter`.
- Abrir na mesma página uma nova aplicação (premir tecla `doc`) seguido de **5: Esquema de página**), permitindo obter na primeira aplicação o Editor já aberto e na segunda o Interpretador (Shell).
- Importar o módulo de **Matemática** da TI-Nspire CX (premir tecla `menu`) seguido de **5: Matemática...** e finalmente **1: from math import \***
- Criar uma primeira função de conversão no sentido °C → °F (`menu`) seguido de **4: Planos integrados** e finalmente **1: funções**).
- Utilizar a tecla `tab` para passar de uma aplicação para a outra (Editor para Interpretador e vice-versa), ou o touchpad colocando o cursor na aplicação pretendida.
- Executar o programa (atalho `ctrl R`) e utilizar no interpretador a função criada para converter em grau Fahrenheit uma temperatura de 40°C, `f(40)`.

```
1.1 | *Doc | RAD | X  
*Temperatura.py | 5/6  
from math import *  
def f(c):  
    F=1.8*c+32  
    return F  
Shell Python | 3/3  
>>>f(40)  
104.0  
>>>|
```





### SUGESTÃO:

Ao executar um programa no modo interpretador (Shell) pode premir a tecla `[var]` e selecionar a função, sem argumentos, que pretende utilizar. Para a executar basta inserir os argumentos e pressionar a tecla `[enter]`, assim obterá os resultados da mesma.

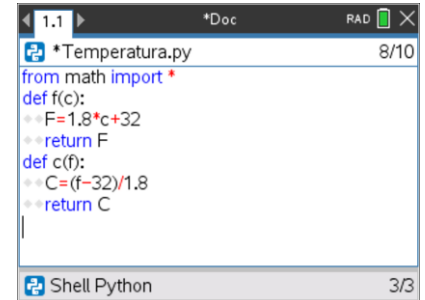
- Concluir a construção do programa **Temperatura**, definindo de forma análoga a função para a conversão contrária:  $^{\circ}F \rightarrow ^{\circ}C$ .

### ATENÇÃO:

Utilizar a opção **2: Retirar indentação** do submenu **3: Editar** do menu do editor para retornar à linha não recuada (caso contrário, uma mensagem de erro será exibida quando o programa for executado). Pode usar o atalho `[shift] + [tab]`.

- Resumindo, a função  $f(c)$  converte a temperatura de  $^{\circ}C$  para  $^{\circ}F$  e a função  $c(f)$  realiza a conversão de  $^{\circ}F$  para  $^{\circ}C$ .
- Guarde o programa (atalho `[ctrl] [B]`), sendo a sintaxe verificada e se for detetado algum erro, este será assinalado.

No fim da gravação será exibida a seguinte mensagem:  **Temperatura.py guardado com sucesso**



```

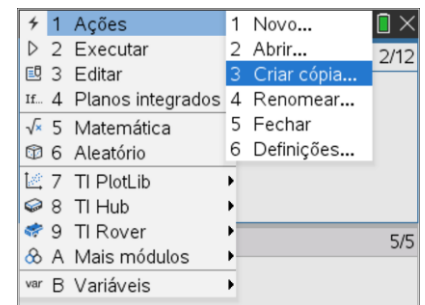
1.1 | *Doc | RAD | 8/10
*Temperatura.py
from math import *
def f(c):
    F=1.8*c+32
    return F
def c(f):
    C=(f-32)/1.8
    return C
Shell Python | 3/3

```

Para encontrar, finalmente, um valor da temperatura que seja igual nas duas unidades de medida é possível fazê-lo através de diversos métodos implementando ciclos e testes, mas que apenas veremos nas unidades 2 e 3.

Assim, prosseguiremos utilizando um ciclo que permita obter medidas com passos de 10 graus (para possivelmente vir a refinar, criando um outro programa).

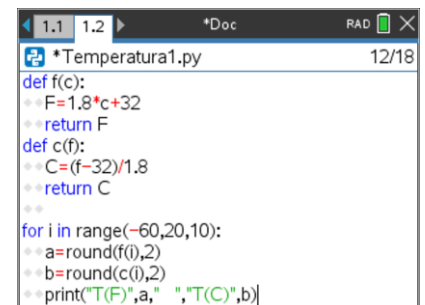
- Pressione a tecla `[menu]`, escolha **1: Ações** e depois **3: Criar cópia...**
- É sugerido nome **Temperatura1** para a cópia. Validar ou modificar.
- Deve notar que o programa foi duplicado na página **2** do problema **1**, pelo que o número da página será **1.2**.



Procuraremos a solução no intervalo  $[-60; 10]$  em passos fixados inicialmente de 10 graus. O passo será um argumento da função, portanto a ser inserido pelo utilizador.

Vamos recorrer a três funções ainda não utilizadas, que são:

- round(a,2)** - para arredondar o valor da variável numérica **a** com 2 casas decimais.
- for i in range(início, fim, passo)** - para gerar um conjunto de temperaturas correspondentes, em ambas as unidades.
- e a instrução **print()** para apresentar os resultados.



```

1.1 | 1.2 | *Doc | RAD | 12/18
*Temperatura1.py
def f(c):
    F=1.8*c+32
    return F
def c(f):
    C=(f-32)/1.8
    return C
for i in range(-60,20,10):
    a=round(f(i),2)
    b=round(c(i),2)
    print("T(F)",a," ", "T(C)",b)

```

Estas funções obtêm-se através do menu, tecla `[menu]`, e da opção **4: Planos integrados** e, respetivamente, das opções:

- 5: Tipo**
- 2: Controlo**
- 6: I/O**

A tecla `[tab]` pode facilitar a edição das instruções, pois permite deslocar mais rapidamente entre campos.





Deverá obter os resultados corretos depois de executar o programa (**ctrl** **R**).

Desta forma, por observação dos resultados do programa, encontrará a resposta para a questão suscitada!

```
1.1 1.2 1.3 *Doc RAD 12/12
Shell Python
>>>from Temperatura1 import *
T(F) -76.000000000000001 T(C) -51.1100000
0000001
T(F) -58.000000000000001 T(C) -45.56
T(F) -40.0 T(C) -40.0
T(F) -22.0 T(C) -34.44
T(F) -4.0 T(C) -28.89
T(F) 14.0 T(C) -23.33
T(F) 32.0 T(C) -17.78
T(F) 50.0 T(C) -12.22
>>>
```

### SUGESTÃO:

Para melhorar o programa, pode ser necessário modificá-lo, solicitando ao aluno que proponha o intervalo de variação da temperatura ao iniciar o programa, bem como o valor do passo (incremento para os valores da temperatura).

Portanto, poderá utilizar-se uma função cujos quatro argumentos sejam os valores de início, fim e passo ou, então, instruções que solicitem ao utilizador, um a um, os valores destes parâmetros (por exemplo, `passo=float(input('Passo='))`).

### NOTA:

Tenha cuidado ao executar um ciclo do tipo `for i in range(início, fim, passo)`, pois o ciclo termina com a variável `i` a tomar o valor de “fim – passo”. O ciclo **FOR**, será abordado durante o estudo da Unidade 2, na lição 2.

