



Neste guião irás continuar a aprender a utilizar **estruturas de repetição** em **PYTHON**. As **estruturas de repetição** permitem **diminuir o volume do código repetindo** partes do código **X vezes** ou enquanto uma **condição se verificar verdadeira**.

Começa por **realizar os exercícios online da W3Schools [aqui](#)**. Realiza os exercícios de **While Loops e For Loops** (apenas **Exercício 3**).

Abre o **IDLE**, cria um **novo programa** de nome **5numcrescente.py** e **digita o seguinte código**:

```
*5,4numcrescente.py - F:/Escola18ano/12anoAPIb/Python/5.4numcrescente.py (3.11.5)*
File Edit Format Run Options Window Help
contador=0
while contador < 20:
    print(contador)
    contador+=1
```

Altera o **programa** para que **imprima, por linha, nºs de 1 a 20**.

Guarda o **programa**.

Cria um **novo programa** e **guarda-o** com o nome **5repete.py** dentro da **pasta Programas**.

Começa por **digitar** o seguinte **código**:

```
*5,3For.py - F:/Escola18ano/12anoAPIb/Python/5.3For.py (3.11.5)*
File Edit Format Run Options Window Help
for contador in range(15):
    print('*')
```

Verifica o **resultado**. Ficou um **asterisco** por linha? Altera o **código** para que seja **possível imprimir sempre na mesma linha**.

```
print('*', end='')
```

Transforma o **programa** para que **dê exatamente o mesmo resultado (na linha seguinte)**, mas **utilizando** uma estrutura de repetição **WHILE**.... Fica a sintaxe:

```
while contador < ???:
```

```
    O que queres repetir aqui
```

```
    contador += ???
```

**Atenção: lembra-te que deves reiniciar o contador antes deste ciclo.**

Imprime **tudo** na **mesma linha**??? Adiciona este **código** ao local **correto** para que **dê** uma **linha** entre os **dois ciclos**.

```
print(end="\n")
```

Guarda o teu **programa**.

Cria um **novo programa** de nome **5reptnome.py**

O **programa** deve **solicitar** ao **utilizador** que **introduza** um **nome** para ser **repetido 5 vezes**. Podes utilizar a **estrutura de repetição** que **quiseres**. Por **exemplo**:

**Qual a palavra que queres que repita 5 vezes?**

**Desafio:** achas que **consegues solicitar** ao **utilizador** o **nº de vezes** que ele **quer** que **repita** a **palavra**?

Guarda o **programa**.




 Cria um novo **programa** de nome **5repnum.py**


 Utiliza a **estrutura de repetição** que **quiseres** para imprimir números de **35 a 49**.

**Recordo** a **sintaxe** da estrutura de repetição **FOR**...

**FOR contador in range([início], fim, [passo do contador]):**

O que queres repetir aqui


 Adiciona outra **estrutura de repetição** para **imprimir** número de **100 até 70 (decrescente)**.

 Adiciona ainda **forma de solicitar** um **valor** ao **utilizador** e **imprima** todos os **nºs** de **0** até esse **número**.


**Desafio:** **solicita**, em vez de um, **dois valores** ao utilizador e **imprime** de **um valor** até ao **outro**.

**NOOB:** introduz sempre um valor menor antes e maior depois.


**PRO:** compara os valores introduzidos para saber qual o maior e menor para saber como ordenar os valores.

 **Guarda** o teu **programa**.

 Cria um novo **programa** de nome **5infinito.py**

 Neste programa deve ser **pedido** ao **utilizador** um valor **numérico** maior ou **igual** a **zero**. Caso o valor **introduzido** seja **negativo** deve aparecer um alerta a dizer **“Valor inválido”** e **voltar a solicitar** o valor **numérico** ao **utilizador**. Caso o valor **respeite** as **regras** é mostrada a seguinte mensagem **“O número escolhido foi o X”**. Neste caso **nunca mais volta a solicitar** um **número**.

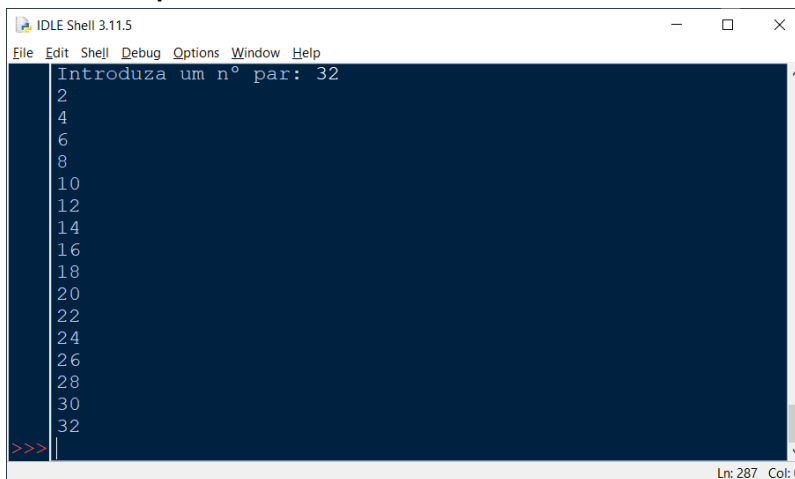
*Ajuda: os ciclos **While True**: podem ser feitos para repetir infinitamente. Só interrompem quando recebem a instrução **break**.*

 **Guarda** o teu **programa**.

 Cria um novo **programa** de nome **5pares.py**

Escreve um **programa** que **solicite** ao **utilizador** um **número inteiro positivo par**.

Deves mostrar todos os **pares anteriores** ao **nº introduzido**.



```
IDLE Shell 3.11.5
File Edit Shell Debug Options Window Help
Introduza um nº par: 32
2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
32
>>>
```

*Dica: lembra-te que o primeiro par inteiro é o 2.*



**NOOB:** utiliza uma estrutura de decisão if que permita verificar se o nº introduzido é par ou não. Caso não seja dá a mensagem no ecrã “O nº introduzido não é par”

**PRO:** para além da verificação de ser par ou não acrescenta forma de questionar se quer introduzir novo nº.



**Guarda o teu programa.**



**Chama o teu professor para avaliar. Mostra também os outros programas que realizaste.**



**Envia todos os programas para a tua pasta Programas na cloud.**