

Ricardo Sérgio

Introdução à Programação e Algoritmia



API B – 12º Ano

1
2
3
4
5
6
7
8
9
10
11
12
13
14

08 {

#Estruturas de
repetição

}

Estruturas de Repetição{

Utilizar estruturas de repetição é muito útil em programação

Permite que o nosso código fique mais pequeno

Evita linhas de código repetidas

Em Python contamos com 2 estruturas de repetição:

While


For

}

```
1 While{
2
3     🐍 While significa Enquanto
4
5     🐍 A lógica implícita desta instrução é a seguinte:
6
7         Enquanto (condição verdadeira):
8             faz esta parte do código
9             [incrementa]
10
11
12
13 }
14
```

```
1 While{
2
3     🐍 Sintaxe:
4
5         while condição:
6             instruções a realizar até que a
7             condição deixe de ser verdadeira
8             [incremento]
9
10
11     🐍 Atenção se utilizarem o incremento esta variável tem de ter valor
12     atribuído antes do ciclo while
13
14 }
```

While{

 Exemplo:

```
5while.py - C:/Users/RS/AppData/Local/Programs/Python/Python311/5while.py (3.11.5)
File Edit Format Run Options Window Help
contador = 1

while contador < 11:
    print(contador)
    contador += 1
```

```
IDLE Shell 3.11.5
File Edit Shell Debug Options Window Help
4 X 10 = 40
>>>
==== RESTART: C:/Users/RS/AppData/Local/Programs/Python/Py
thon311/5while.py ====
1
2
3
4
5
6
7
8
9
10
```


O que aconteceria se o contador fosse inicializado a 11

O que aconteceria se o contador fosse inicializado a 4


E se o contador tivesse a condição de paragem em 14 (início a 1)

E se quisesse que o 14 ainda aparecesse no print

While{

1
2
3  O **While** é muito **útil** para realizar **ciclos infinitos**

4
5  Este “**infinitos**” têm um **fim** 😊 **definido** pelo **programador**

6
7  Por exemplo quando queremos **voltar a repetir** o **programa** ou **parte dele** em caso de:

8
9 **Valores inválidos** (ex: não podia introduzir nºs negativos)

10
11 **Novos pedidos** (ex: voltar a fazer a tabuada, mas de outro nº)

12
13 **Não necessitar** de **voltar a correr** o **programa** para o poder
14 **continuar a utilizar** }

While{

Exemplo:


```
5.8pedenomeinfinito.py - F:/Escola18ano/12anoAPIb/Python/5.8pedenomeinfinito.py (3...  
File Edit Format Run Options Window Help  
while True:  
    nome=input("Como te chamas: ")  
    print("Olá", nome+".")  
    continuar=input("Deseja introduzir novo nome?s/n: ")  
    if continuar != "s":  
        print("Até à próxima!")  
        break
```

```
*IDLE Shell 3.11.5*  
File Edit Shell Debug Options Window Help  
Como te chamas: Ricardo  
Olá Ricardo.  
Deseja introduzir novo nome?s/n: s  
Como te chamas: |
```


Ln: 478 Col: 16



```
1 For {
2
3     🐍 For significa De.
4
5     🐍 A lógica implícita desta instrução é a seguinte:
6
7         De valor até que:
8             faz esta parte do código X vezes
9
10
11
12     🐍 O For não necessita de incrementador à parte nem tão pouco que
13     seja definido um valor inicial
14 }
```

For{

 Sintaxe:

```
for contador in range([valor a iniciar contador], valor  
final do contador, [valor do incremento do contador]):  
    Código a fazer X vezes aqui
```

 Se o **valor inicial** for **omitido** começa em **0** por **defeito**

 Se **incremento** for **omitido** incrementa contador **1 por ciclo**

 O **valor final** será sempre **não pertence**

}

For {

Exemplos:

```
5.1For.py - F:/Escola18ano/12anoAPIb/Python/5.1For.py (3.11.5)
File Edit Format Run Options Window Help
for x in range(6):
    print(x)
```

0 que resulta?

```
IDLE Shell 3.11.5
File Edit Shell Debug Options Window Help
>>>
=== RESTART: F:/Escola18ano/12anoAPIb/Python/5.1For.py ===
0
1
2
3
4
5
>>>
```

1
2
3
4
5
6
7
8
9
10
11
12
13
14

}

Ln: 119 Col: 0

For{

Exemplo:

```
*5For.py - F:/Escola18ano/12anoAPIb/Python/5For.py (3.11.5)*
File Edit Format Run Options Window Help
for contador in range(1, 10):
    print(contador)
```


Qual o resultado?

Como imprimir o 10?


```
IDLE Shell 3.11.5
File Edit Shell Debug Options Window Help
==== RESTART: F:/Escola18ano/12anoAPIb/Python/5For.py ====
1
2
3
4
5
6
7
8
9
>>>
```

Ln: 103 Col: 0

For{

 Exemplos:

```
*5.2For.py - F:/Escola18ano/12anoAPIb/Python/5.2For.py (3.11.5)*
File Edit Format Run Options Window Help
for contador in range(2, 10, 3):
    print(contador)
```

 O que resulta?

```
IDLE Shell 3.11.5
File Edit Shell Debug Options Window Help
>>>
=== RESTART: F:/Escola18ano/12anoAPIb/Python/5.2For.py ===
2
5
8
>>>
```

Ln: 153 Col: 0

1
2
3
4
5
6
7
8
9
10
11
12
13
14

Para Hoje{

}

Realiza os guiões práticos existentes
no Website da tua disciplina

CREDITS: This presentation template was created by **Slidesgo**,
including icons by **Flaticon**, and infographics & images by **Freepik** and
illustrations by **Stories**

