

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

04 {

#Outputs

}

# Output Vs Input {

1  
2  
3  OUTPUT implica saída. Ecrã, Impressora, movimento etc

4  
5  INPUT implica entrada. Teclado, rato, etc

6  
7  Em PYTHON Output é informação dada ao utilizador através da  
8 instrução print

9  
10  Em PYTHON Input é informação solicitada ao utilizador, através da  
11 instrução input

12  
13  
14 }  
}

# Output{

1  
2  
3  Já vimos a instrução `print`. Recordemos a `sintaxe`:

```
4  
5         print("Texto aqui")
```

6  O que acontece se utilizarmos `plicas` `'` em vez de `aspas` `"`?

```
7  
8         print('Olá mundo!')
```

9  
10  `Experimenta` para ver se `funciona`!

11  Também `funciona` com `plicas`.

12  
13  
14 }  
}

# Output{

1  
2  
3  Desafio: `Imprime` o teu `nome` na `linha seguinte`

4  
5  E se `quisesses imprimir` a `seguir` ao `ponto de exclamação` do `Olá`  
6 `mundo?` Com `férias?`

7  
8  Claro escrevias `tudo` no `mesmo Print`

9  
10 `print('Olá mundo! Ricardo Sérgio')`

11  
12  
13 }  
14

# Texto Vs Número {

1  
2  
3  O `print` apenas faz `output` de `texto`?

4  
5 `Noppp`. Faz `output` de `tudo`.

6  `Números`?

7  
8 `Sim`. Como?

9 `print(Nº)`

10  `Com` ou `sem` `aspas`?

11 `Se é nº é sem aspas`

12  
13  
14 }  
}



```
*1 output.py - F:/Escola18ano/12anoAPIb/Python/1 output.py (3.11.5)*  
File Edit Format Run Options Window Help  
print(123)
```

# Texto Vs Número{

Com ou sem aspas eis as questão!

Qual o output que produz cada linha de código do programa?

```
*1.0 output.py - F:/Escola18ano/12anoAPIb/Python/1.0 output.py (3.11.5)*
```

```
File Edit Format Run Options Window Help
```

```
print('2+4')  
print(5-3)  
print(5*2)  
print("4/2")
```

```
IDLE Shell 3.11.5
```

```
File Edit Shell Debug Options Window Help
```

```
2+4  
2  
10  
4/2
```

# Operadores {

🐍 Quais os operadores existentes em Python?

Operador	O que faz?
+	Somar
-	Subtrair
*	Multiplicar
/	Dividir
%	Resto da divisão inteira
//	Cociente (parte inteira)
**	expoente

# 1 Texto Vs Número {

2  
3  Conclusão:

4 `Se está dentro de aspas (ou plicas) é considerado texto`

5  É possível `misturar, texto e números no mesmo print?`

6  `Sim, mas para isso tens de concatenar.`

7  
8  `WHAAAAAT????`

9  
10  `Pois... concatenar é o termo que utilizamos quando queremos`  
11 `misturar algo diferente em programação`

12  
13  `É o local onde dizemos ao compilador que a partir daquele ponto`  
14 `algo muda` }

# Texto Vs Número{

🐍 Por exemplo, pode ser **mudar** de **texto** para **número**

```
print("O resultado desta soma é ") 14
```

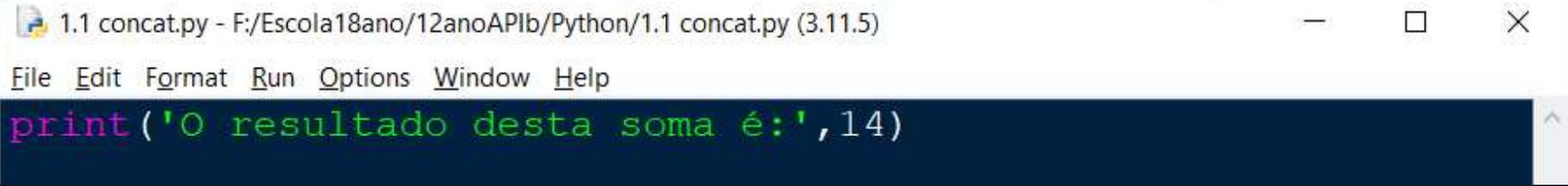
🐍 Alterna entre **texto** e **número**. Isto funciona?

🐍 Noppppp!

🐍 Temos de **separar** a parte do **texto** da parte do **número**, temos de concatenar.

🐍 Em Python, com **print**, utiliza-se a **vírgula** , para **concatenar**

🐍 O código **correto seria**:



```
1.1 concat.py - F:/Escola18ano/12anoAPIb/Python/1.1 concat.py (3.11.5)  
File Edit Format Run Options Window Help  
print('O resultado desta soma é:',14)
```

# Concatenar{

🐍 No próximo capítulo iremos concatenar utilizando o sinal +

🐍 Lembra-te que a vírgula, dentro da instrução print, acrescenta um espaço em branco antes e depois do local onde foi utilizada. Isto nem sempre pode ir de encontro ao que o programador pretende.

🐍 Exemplo:

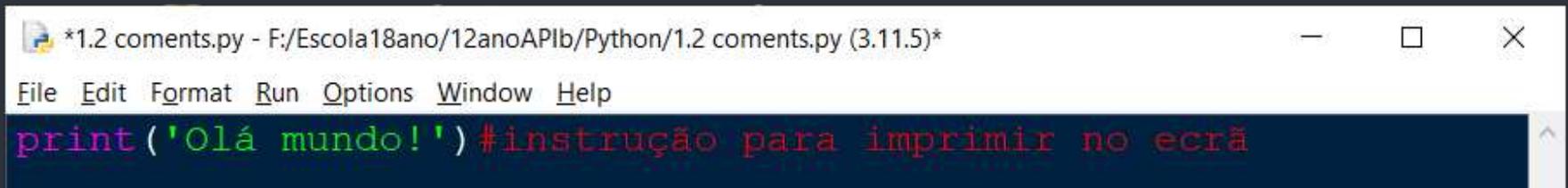
```
*2.2 variaveisguiiao.py - F:/Escola18ano/12anoAPIb/Python/2.2 variaveisguiiao.py (3.11.5)*
File Edit Format Run Options Window Help
print("Eu tenho",18, ".")
```

Eu tenho 18 .

} Iremos abordar a solução para este problema na próxima aula.

# Comentários{

- Comentar em Python faz-se através de um # antes do que pretendemos comentar. Não é lido pelo compilador.
- Úteis para dizer o que faz cada parte do código – orientação em códigos muito longos ou a ser feitos por várias pessoas
- Retirar da compilação parte do código, seja para depuração ou para retirar pontualmente partes do programa



```
*1.2 coments.py - F:/Escola18ano/12anoAPIb/Python/1.2 coments.py (3.11.5)*  
File Edit Format Run Options Window Help  
print('Olá mundo!') #instrução para imprimir no ecrã
```



```
IDLE Shell 3.11.5  
File Edit Shell Debug Options Window Help  
Olá mundo!
```

# Comentários{

Se pretendermos **comentar mais** do que **uma linha** temos de utilizar 3 aspas

Exemplo:

```
""" Comentário  
de várias  
linhas  
aqui """
```

}

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**

