



# Função print e variáveis

🕒 Created	@April 27, 2022 7:14 PM
▼ Class	Python3
▼ Type	Alura

## Print

A função é a **print()**. Basta passar a mensagem como parâmetro para a função, que a mensagem será impressa.

```
print("ola mundo")
```

## Resultado

```
ola mundo
```

No console do Python temos uma função de ajuda, a função **help()**:

```
help()
```

Podemos reparar que um novo console aparece:

```
help>
```

Agora, como queremos saber mais sobre a função **print**, vamos digitá-la:

```
help(print)
```

Help on built-in function print in module builtins:

```
print(...)
  print(value, ..., sep=' ', end='\n', file=sys.stdout, flush=False)

  Prints the values to a stream, or to sys.stdout by default.
  Optional keyword arguments:
  file: a file-like object (stream); defaults to the current sys.stdout.
  sep:   string inserted between values, default a space.
  end:   string appended after the last value, default a newline.
  flush: whether to forcibly flush the stream.

(END)
```

Inicialmente, o que nos importa são os três primeiros valores que a função `print` pode receber:

- `value` é o valor que queremos imprimir, as reticências indicam que a função pode receber mais de um valor, basta separá-los por vírgula.
- `sep` é o separador entre os valores, por padrão o separador é um espaço em branco.
- `end` é o que acontecerá ao final da função, por padrão há uma quebra de linha, uma nova linha ( `\n` ).

Podemos apertar a tecla **Q** para sair da documentação da função e **CTRL + C** ou **CTRL + D** para sair do console de ajuda do Python. De volta ao console do próprio Python, podemos testar a função `print` com os valores que vimos:

```
print("Brasil", "ganhou", 5, "titulos mundiais", sep="-")
```

## Resultado

```
Brasil-ganhou-5-titulos mundiais
```

Como modificamos o separador, agora os valores são separados por um hífen. Vamos testar o `end` agora, não passando nada para ele:

```
print("Brasil", "ganhou", 5, "titulos mundiais", end="")
```

## Resultado

```
Brasil ganhou 5 titulos mundiais>>>
```

Uma nova linha não é criada, ou seja, o que colocarmos dentro do `end` será impresso ao final da função.

## Variáveis

Agora queremos flexibilizar a função, queremos poder imprimir outros países, como Itália, Alemanha, Argentina... Para isso teremos que mudar o nome do país e o número de títulos conquistados. Então vamos **definir o valor com o nome do país fora da função** `print`. Vamos definir uma **variável** para o nome do país, já que o seu valor pode variar:

```
pais = "Italia"
```

Definimos uma variável e atribuímos a ela um valor. Assim como fizemos com o nome do país, vamos fazer também com a quantidade de títulos:

```
pais = "Italia"  
quantidade = 4
```

Com as variáveis definidas, podemos refazer a função `print`, só que dessa vez passando as variáveis no lugar dos antigos valores:

```
pais = "Italia"  
quantidade = 4  
print(pais, "ganhou", quantidade, "titulos mundiais")
```

## Resultado

```
Italia ganhou 4 titulos mundiais
```

Agora a mensagem é impressa no mesmo molde da anterior, só que dessa vez com variáveis! Mas qual é o tipo dessas variáveis? O tipo da variável depende do valor que passamos para ela. Podemos "perguntar" para a variável qual é o seu tipo, passando-a para a função `type`:

```
type(pais)
type(quantidade)
```

## Resultado

```
<class 'str'>
<class 'int'>
```

O valor `str` significa que a variável é do tipo **string**, já que o seu valor está entre aspas duplas. E `int` significa que a variável é do tipo **inteiro**, já que passamos um valor inteiro para a variável.

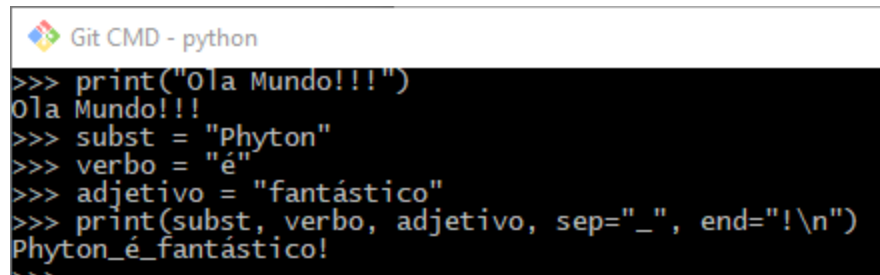
## Exercícios

```
subst = "Python"
verbo = "é"
adjetivo = "fantástico"
print(subst, verbo, adjetivo, sep="_", end="!\n")
```

Qual será a saída do comando `print`?

```
subst = "Phyton"
verbo = "é"
adjetivo = "fantástico"
```

```
print(subst, verbo, adjetivo, sep="_", end="!\n")  
Phyton_é_fantástico!
```



A screenshot of a terminal window titled "Git CMD - python". The terminal shows the following Python code being executed in a REPL environment:

```
>>> print("Ola Mundo!!!")  
Ola Mundo!!!  
>>> subst = "Phyton"  
>>> verbo = "é"  
>>> adjetivo = "fantástico"  
>>> print(subst, verbo, adjetivo, sep="_", end="!\n")  
Phyton_é_fantástico!  
>>>
```