

Básico de Python

Sergio Pedro Rodrigues Oliveira

SUMÁRIO

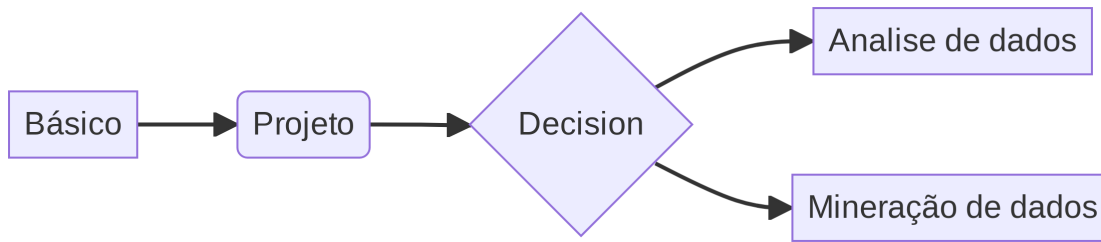
1	Diagrama de estudo	1
2	Variáveis e tipos de dados simples	1
2.1	<code>print()</code>	1
2.2	<code>print()</code> com variáveis	1
2.3	Regras de nomes de variáveis	1
2.4	Concatenando strings	2
2.5	Métodos auxiliares da função <code>print()</code>	2
2.6	Caracteres de escape	3
2.7	Removendo espaços em branco <code>print()</code>	4
2.8	Números	5
2.9	Funções de conversão de tipo	5
2.10	Operações básicas	6
2.11	Evitando erros de tipo com a função <code>str()</code>	7

LISTA DE FIGURAS

LISTA DE TABELAS

1	Caracteres de escape	3
2	Principais tipos de dados	5
3	Funções de conversão de tipo	5
4	Operações básicas	6
5	Algumas operações da biblioteca <code>math</code>	6
6	Operações Lógicas	6

1 Diagrama de estudo



2 Variáveis e tipos de dados simples

2.1 print()

Print é uma função que exibe uma string na tela.

Exemplo: `print("string")`

2.2 print() com variáveis

Podemos usar a função `print()` para imprimir uma variável string.

Exemplo:

```
message = "Hello world!"  
print(message)
```

2.3 Regras de nomes de variáveis

Regras ou diretrizes para usar variáveis em Python.

- Nomes de variáveis deve conter apenas letras, números e underscores. Podemos começar a variável com letra ou underscore, mas nunca com um número.
- Espaços não são permitidos em nomes de variáveis, mas underscores podem ser usados para separar palavras.
- Evite usar palavras reservadas e nome de funções em Python como nome de variáveis.
- Nomes de variáveis devem ser concisos, porém descritivos.
- Tome cuidado ao usar a letra l e a letra maiuscula O, pois podem ser confundidas com os números 1 e 0.

2.4 Concatenando strings

Podemos usar o simbolo de (+) para combinar strings (concatenar).

Exemplo:

```
>>> first_name = "ada"
>>> last_name = "lovelace"
>>> full_name = first_name + " " + last_name
>>> print("Hello, " + full_name.title() + "!")
Hello, Ada Lovelace!
```

Os espaços em branco entre aspas servem para criar espaços na string.

2.5 Métodos auxiliares da função print()

1. .title()

Coloca apenas as primeiras letras em maiúsculas de cada palavra e o resto em minúscula.

Exemplo:

```
>>> print(full_name.title())`
Ada Lovelace
```

2. .upper()

Coloca todas as letras em maiúsculas.

Exemplo:

```
>>> print(full_name.upper())`
ADA LOVELACE
```

3. .lower()

Coloca todas as letras em minusculas. O método `.lower()` é particularmente útil para armazenar dados. Converter os dados em minúscula antes de armazenar.

Exemplo:

```
>>> print(full_name.lower())`
ada lovelace
```

2.6 Caracteres de escape

Podemos inserir alguns caracteres de escape no texto para executar alguma ação, como pular linha, gerar tabulação e etc. Alguns caracteres podem ser vistos na [Table 1](#).

Todos os caracteres de escape começam com barra(\) + complemento.

Table 1: Caracteres de escape

Caracteres de escape	Descrição
\t	Gera tabulação (tab).
\n	Gera quebra de linha.

Exemplo:

```
print("Language:\nPython\nJava\nC\nJavaScript")
```

```
Language:
Python
Java
C
JavaScript
```

2.7 Removendo espaços em branco print()

1. .rstrip()

Remove espaço em branco do lado direito.

Exemplo:

```
favorite_language = 'python '  
favorite_language.rstrip()
```

'python'

2. .lstrip()

Remove espaço em branco do lado esquerdo.

Exemplo:

```
favorite_language = ' python'  
favorite_language.lstrip()
```

'python'

3. .strip()

Remove os espaços em branco dos dois lados ao mesmo tempo.

Exemplo:

```
favorite_language = ' python '  
favorite_language.strip()
```

'python'

- Os metodos usados não removem os espaços em branco em definitivo, para remover em definitivo é necessario armazenar o valor novo na variável.

```
favorite_language = ' python '  
favorite_language = favorite_language.strip()  
favorite_language
```

'python'

2.8 Números

A linguagem Python faz tipagem automática (dinâmica), tipa a variável de acordo com o uso. E o Python contém uma tipagem forte, não faz conversão automática do tipo de uma variável para executar uma ação (operação).

Em resumo, python tem é uma linguagem de tipagem dinâmica e forte.

Os principais tipos de dados no Python são estão presentes na tabela Table 2.

Table 2: Principais tipos de dados

Nome	Abreviação	Descrição
Inteiro	int	Números inteiros
Ponto flutuante	float	Números com ponto decimal

2.9 Funções de conversão de tipo

Table 3: Funções de conversão de tipo

Tipo para converter	Função	Descrição
int	<code>int()</code>	Converte variável para tipo inteiro(int)
float	<code>float()</code>	Converte variável para tipo float

2.10 Operações básicas

Table 4: Operações básicas

Operação	Símbolo	Exemplo
Soma	+	$2+2=4$
Subtração	-	$3-2=1$
Multiplicação	*	$2*3=6$
Divisão	/	$3/2=1$
Resto da divisão	%	$10\%8=2$
Potência	**	$3**2=9$
Raiz	**	$4**0.5=2$

Podemos usar o pacote `math` para ampliar as funções matemáticas do Python.

inserir tabela com as principais funções matemáticas do pacote `math`

Table 5: Algumas operações da biblioteca `math`

Operação	Símbolo	Exemplo
Soma	+	$2+2=4$
Subtração	-	$3-2=1$
Multiplicação	*	$2*3=6$
Divisão	/	$3/2=1$
Resto da divisão	%	$10\%8=2$
Potência	**	$3**2=9$
Raiz	**	$4**0.5=2$

Inserir operações lógicas

Table 6: Operações Lógicas

Operação	Símbolo	Exemplo
Soma	+	$2+2=4$
Subtração	-	$3-2=1$
Multiplicação	*	$2*3=6$
Divisão	/	$3/2=1$
Resto da divisão	%	$10\%8=2$
Potência	**	$3**2=9$
Raiz	**	$4**0.5=2$

2.11 Evitando erros de tipo com a função `str()`