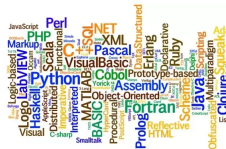
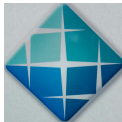


# Paradigmas de Linguagens de Programação em Python

## Aula 02

Evandro J.R. Silva<sup>1</sup>

<sup>1</sup>Bacharelado em Ciência da Computação  
Estácio Teresina



# Sumário

## 1 Introdução

## 2 Variáveis

- Formato de nomes
- Palavras Especiais
- Vinculação

## 3 Exercícios

## 4 FIM

# Introdução

# Introdução

- Tema da ementa:
  - 2. Nomes, vinculações e escopo.
- Tópico
  - 2.1 Variáveis
- Objetivo
  - Caracterizar a natureza dos nomes e palavras especiais nas linguagens de programação, baseando-se na linguagem Python.

# Variáveis

# Variáveis

- “Coleção de propriedades, ou atributos, das quais a mais importante é o tipo, um conceito fundamental em linguagens de programação”.

Tipo	Tamanho (bytes)	Faixa de valores
char	1	-128 a 127 ou 0 a 255
signed char	1	-128 a 127
unsigned char	1	0 a 255
short	2	-32.768 a 32.767
unsigned short	2	0 a 65.535
int	2	-32.768 a 32.767
	4	-2.147.483.648 a 2.147.483.647
unsigned int	2	0 a 65.535 ou
	4	0 a 4.294.967.295
long	8	-9.223.372.036.854.775.808
	(4 em SOs de 32 bits)	a 9.223.372.036.854.775.807
unsigned long	8	0 a 18.446.744.073.709.551.615

Tabela: Tipos de dados em C

# Variáveis

<b>Tipo</b>	<b>Tamanho (bytes)</b>	<b>Faixa de valores</b>	<b>Precisão</b>
float	4	1.2E-38 a 3.4E+38	6 casas decimais
double	8	2.3E-308 a 1.7E+308	15 casas decimais
long double	10	3.4E-4932 a 1.1E+4932	19 casas decimais

**Tabela:** Tipos de dados em C (continuação)

# Variáveis

## ■ Alguns tipos do Python

Tipo	Comentário
bool	Representa os valores <b>True</b> e <b>False</b> . É um subtipo de inteiros, onde 0 é falso e 1 é verdadeiro.
int	Implementado utilizando o tipo <b>long</b> de C.
long	Faixa de valores ilimitada, porém sujeita à quantidade disponível de memória.
double	Implementado utilizando o tipo <b>double</b> de C. A precisão vai depender da máquina em que o Python estiver executando.

Tabela: Alguns tipos do Python



# Variáveis

- Uma variável pode ser caracterizada por seis atributos: nome, endereço, valor, tipo, tempo de vida e escopo
- Enquanto programamos, uma variável terá um nome associado, o que facilita nossa leitura do código
  - Ex.:  
**int** x = 10;  
**String** nome = "Fulano da Silva";
- Essa variável estará associada a um tipo e a um endereço de memória.
- Durante a compilação/interpretação é onde acontece de fato o endereçamento das variáveis na memória.
- Durante a execução, o computador não vê os nomes das variáveis (não existem mais), mas sim os seus endereços.

# Variáveis

- Durante a compilação, o compilador realiza diversas tarefas, entre elas, a associação das variáveis a um endereço.
- O compilador utiliza uma tabela de símbolos para armazenar (provisoriamente) as informações de identificadores, os quais podem ser constantes, funções, variáveis e tipos de dados.

Identificador	Tipo	Endereço	Valor
x	int32 (4 bytes)	0x00001	10
...	float32	0x00005	2.35
nome	String	0x00009	Fulano da Silva
...	int16	0x00024	5
y	int32	0x00001	10

**Tabela:** Exemplo simplificado de uma tabela de símbolos

# Variáveis

## Formato de nomes

# Formato de nomes

- “Um nome é uma cadeia de caracteres usada para identificar alguma entidade em um programa.”
- Java, C++, C#, dentre outras, não impõem limites ao tamanho do nome.
- Outras linguagens, principalmente mais antigas, podem limitar o tamanho.
  - Ex.: em C (C99), os nomes podem ter qualquer tamanho, porém só são considerados os 31 primeiros caracteres, se o nome for definido fora das funções (nome externo), ou os 63 primeiros, se o nome for definido dentro das funções (nome interno).
- As linguagens também podem ser *case sensitive* ou não, ou seja, podem diferenciar letras maiúsculas de minúsculas, ou não.
  - As linguagens que têm inspiração em C são *case sensitive*. Portanto, os nomes `rosa`, `Rosa` e `ROSA` são diferentes.
  - Dependendo da situação, isso pode ser visto como um potencial problema de legibilidade.

# Formato de nomes

## ■ Convenções

- Nas linguagens *case sensitive* os nomes podem ser escritos de múltiplas formas.
- Com o tempo surgiram convenções de como escrever os nomes, principalmente por causa dos casos que envolvem múltiplas palavras.
- As convenções têm seus *apelidos*

Nome	Formato
flatcase	duaspalavras
UPPERCASE	DUASPALAVRAS
(lower)camelCase, dromedaryCase	duasPalavras
PascalCase, UpperCamelCase, StudyCase	DuasPalavras
snake_case, snail_case, pothole_case	duas_palavras
ALL_CAPS, SCREAMING SNAKE CASE MACRO_CASE, CONSTANT_CASE	DUAS_PALAVRAS
camel_Snake_Case	duas_Palavras
Pascal_Snake_Case, Title_Case	Duas_Palavras
kebab-case, dash-case lisp-case, spinal-case	duas-palavras
TRAIN-CASE, COBOL-CASE SCREAMING-KEBAB-CASE	DUAS-PALAVRAS
Train-Case, HTTP-Header-Case	Duas-Palavras

Tabela: Convenções de escrita de nomes

# Formato de nomes

## ■ Convenções

### ■ C / C++

- Na Biblioteca Padrão de C o mais comum é o flatcase, com abreviações. Por exemplo: `isalnum`.
- Na Biblioteca Padrão do C++ é mais utilizado o `sake_case` (ex.: `out_of_range`). **Macros e constantes** são escritas com `ALL_CAPS`.

### ■ Java

- Os nomes de classes são escritas com `UpperCamelCase`, métodos e variáveis são escritos com o `camelCase`, e constantes são escritas com `UPPERCASE`.

### ■ Python

- Os nomes de classes devem ser escritas com `UpperCamelCase`, constantes como `ALL_CAPS`, e os demais como `snake_case`.

# Variáveis

## Palavras Especiais

# Palavras Especiais

- “Palavras especiais em linguagens de programação são usadas para tornar os programas mais legíveis ao nomearem as ações a serem realizadas. Elas também são usadas para separar as partes sintáticas das sentenças e programas.”



# Palavras Especiais

- “Palavras especiais em linguagens de programação são usadas para tornar os programas mais legíveis ao nomearem as ações a serem realizadas. Elas também são usadas para separar as partes sintáticas das sentenças e programas.”
- “Na maioria das linguagens, as palavras especiais são classificadas como palavras reservadas, o que significa que não podem ser redefinidas pelos programadores. Mas, em algumas, como em Fortran, elas são apenas palavras-chave, ou seja, podem ser redefinidas.”

# Palavras Especiais

- “Palavras especiais em linguagens de programação são usadas para tornar os programas mais legíveis ao nomearem as ações a serem realizadas. Elas também são usadas para separar as partes sintáticas das sentenças e programas.”
- “Na maioria das linguagens, as palavras especiais são classificadas como palavras reservadas, o que significa que não podem ser redefinidas pelos programadores. Mas, em algumas, como em Fortran, elas são apenas palavras-chave, ou seja, podem ser redefinidas.”
- Também, “nomes que são definidos em outras unidades de programa como os pacotes em Java e as bibliotecas em C e C++, podem se tornar visíveis para um programa.”

# Palavras Especiais

- “Palavras especiais em linguagens de programação são usadas para tornar os programas mais legíveis ao nomearem as ações a serem realizadas. Elas também são usadas para separar as partes sintáticas das sentenças e programas.”
- “Na maioria das linguagens, as palavras especiais são classificadas como palavras reservadas, o que significa que não podem ser redefinidas pelos programadores. Mas, em algumas, como em Fortran, elas são apenas palavras-chave, ou seja, podem ser redefinidas.”
- Também, “nomes que são definidos em outras unidades de programa como os pacotes em Java e as bibliotecas em C e C++, podem se tornar visíveis para um programa.”
- “Esses nomes são predefinidos, mas visíveis apenas se explicitamente importados. Uma vez importados, eles não podem ser redefinidos.”

# Variáveis Vinculação

# Vinculação

- “É a associação entre um atributo e uma entidade, como entre uma variável e seu tipo ou valor, ou entre uma operação e um símbolo”.
- O momento no qual uma vinculação ocorre é chamado de **tempo de vinculação**.

## Exemplo

```
count = count + 5;
```

Algumas das vinculações e seus tempos de vinculação para as partes dessa sentença são:

- O tipo `count` é vinculado em tempo de compilação.
- O conjunto dos valores possíveis de `count` é vinculado em tempo de projeto de compilador.
- O significado do símbolo de operador `+` é vinculado em tempo de compilação, quando os tipos dos operandos tiverem sido determinados.
- A representação do literal `5` é vinculada ao tempo de projeto do compilador.
- O valor de `count` é vinculado em tempo de execução a essa sentença.

# Vinculação

- Uma vinculação de atributos a variáveis pode ser estática ou dinâmica.
- **Vinculação Estática**
  - A vinculação ocorre pela primeira vez antes do tempo de execução e permanece inalterada ao longo da execução do programa.
  - Ex.: Constantes, e Tipagem Estática.
- **Vinculação Dinâmica**
  - A vinculação ocorre pela primeira vez durante o tempo de execução ou pode ser mudada ao longo do curso da execução do programa.
  - Ex.: Variáveis, e Tipagem Dinâmica.

## Exercícios

# Exercícios

## ■ Vamos começar a praticar:

- <https://wiki.python.org.br/EstruturaSequencial>
- <https://wiki.python.org.br/EstruturaDeDecisao>



Terminamos por hoje!

Aula baseada no livro:

SEBESTA, Robert W. **Conceitos de Linguagens de Programação**. 11a Ed., Capítulo 5, Porto Alegre: Bookkman, 2018.