

Variáveis

Prof. Demétrius de Castro

83 9 8773-0383

@demdecastro

demdecastro@gmail.com

www.demetriusdecastro.com.br

Definição Simples

As variáveis são um dos conceitos fundamentais em programação. Elas são utilizadas para armazenar e manipular dados na memória do computador durante a execução de um programa. Embora o conceito de variável seja comum a todas as linguagens de programação, sua implementação e uso variam dependendo da linguagem.

```
    mirror_mod = modifier_obj
    # Set mirror object to mirror
    mirror_mod.mirror_object = selected_object
    if operation == "MIRROR_X":
        mirror_mod.use_x = True
        mirror_mod.use_y = False
        mirror_mod.use_z = False
    elif operation == "MIRROR_Y":
        mirror_mod.use_x = False
        mirror_mod.use_y = True
        mirror_mod.use_z = False
    elif operation == "MIRROR_Z":
        mirror_mod.use_x = False
        mirror_mod.use_y = False
        mirror_mod.use_z = True
```

```
    # Selection at the end - add
    mirror_ob.select = 1
    selected_object.select = 1
    context.scene.objects.active = selected_object
    print("Selected" + str(modifier))
    mirror_ob.select = 0
    bpy.context.selected_objects.append(selected_object)
    data.objects[one.name].select = 1
    print("please select exactly one object")
```

```
-- OPERATOR CLASSES --
```

```
types.Operator):
    # X mirror to the selected
    # object.mirror_mirrored_x"
    # mirror X"
```

```
context):
    # context.active_object is not
```

Definição Técnica

Em computação, uma variável é uma referência nomeada a um endereço de memória que armazena um valor de um determinado tipo de dado. Dependendo da linguagem de programação, as variáveis podem ter diferentes características relacionadas à tipagem, mutabilidade, escopo e tempo de vida.

Tecnicamente, uma variável é um identificador simbólico associado a um espaço na memória do computador, cujo conteúdo pode ser modificado em tempo de execução (se a variável for mutável). O compilador ou interpretador gerencia essa associação conforme as regras de escopo e tempo de vida definidas pela linguagem.

Definição Técnica

Em linguagens de programação de tipagem estática (como Java, C e Rust), o tipo da variável precisa ser definido no momento da sua declaração e não pode ser alterado. Já em linguagens de tipagem dinâmica (como Python e JavaScript), o tipo é inferido automaticamente com base no valor atribuído.



Para Que Servem as Variáveis?

As variáveis desempenham um papel fundamental na programação, permitindo

Armazenamento Temporário de Dados: Dados podem ser armazenados e recuperados durante a execução do programa.

Manipulação de Informações: O valor pode ser modificado dinamicamente para processamento.

Facilidade de Reuso e Manutenção: Usar variáveis melhora a legibilidade e modularidade do código.

Interação com o Usuário e Entrada de Dados: Permitem que um programa armazene valores informados pelo usuário.

Para Que Servem as Variáveis?

Exemplo de Uso em Algoritmos num cálculo matemático

Java

```
int soma = 10 + 5;  
System.out.println(soma); // Exibe 15
```

Python

```
Soma = 10 + 5  
Print(soma) // Exibe 15
```



O que são variáveis?

Uma variável pode ser entendida como um "espaço nomeado na memória" que armazena um valor que pode ser alterado durante a execução do programa. Cada variável tem:

- Um nome (identificador)
- Um tipo (definido explicitamente ou implicitamente)
- Um valor (o dado armazenado)
- Um escopo (onde pode ser acessada)
- Um tempo de vida (duração na memória)



Declaração e Tipagem de Variáveis

Em Java

Java é uma linguagem fortemente tipada e estática, ou seja, o tipo de uma variável deve ser definido no momento da sua declaração e não pode ser alterado posteriormente.

Declaração de Variáveis em Java

A sintaxe básica para declarar uma variável em Java é:
`tipo nomeDaVariavel = valor;`

Declaração e Tipagem de Variáveis

Exemplo

```
int idade = 25; // Variável inteira
double altura = 1.75; // Variável de ponto flutuante
char inicial = 'J'; // Variável do tipo caractere
boolean ativo = true; // Variável booleana
String nome = "João"; // Variável do tipo string
```



Declaração e Tipagem de Variáveis

Em Python

Python é uma linguagem dinamicamente tipada, ou seja, não exige que o programador defina explicitamente o tipo da variável. O interpretador determina o tipo com base no valor atribuído.

Declaração de Variáveis em Python

A sintaxe é mais simples:

`nome_da_variavel = valor`

Declaração e Tipagem de Variáveis

Exemplos

```
idade = 25 # Inteiro
```

```
altura = 1.75 # Ponto flutuante
```

```
inicial = 'J' # Caractere (string de tamanho 1)
```

```
ativo = True # Booleano
```

```
nome = "João" # String
```

Não há necessidade de especificar o tipo, pois Python faz inferência de tipo automaticamente.

Python permite alterar o tipo de uma variável em tempo de execução, algo impossível em Java.

Tipos de Dados

Tipos Primitivos em Java

Java tem tipos primitivos, que são mais eficientes em termos de desempenho, e tipos referência, que são objetos.

Tipo	Tamanho	Faixa de valores	Exemplo
byte	8 bits	-128 a 127	byte b = 100;
short	16 bits	-32.768 a 32.767	short s = 1000;
int	32 bits	-2.147.483.648 a 2.147.483.647	int i = 100000;
long	64 bits	Muito maior	long l = 1000000000L;
float	32 bits	Precisa de 'f' no final	float f = 10.5f;
double	64 bits	Padrão para decimais	double d = 10.5;
char	16 bits	Caracteres Unicode	char c = 'A';
boolean	1 bit	true ou false	boolean ativo = true;

Tipos de Dados

Tipos em Python

Python não tem tipos primitivos no sentido tradicional, pois tudo é objeto. No entanto, os tipos básicos são:

Tipo	Exemplo
int	idade = 25
float	altura = 1.75
str	nome = "João"
bool	ativo = True
list	numeros = [1, 2, 3, 4]
tuple	coordenadas = (10, 20)
dict	aluno = {"nome": "Ana", "idade": 22}
set	conjunto = {1, 2, 3, 3, 2}

Escopo de Variáveis

Em Java

Temos:

- Variáveis locais: Declaradas dentro de um método, visíveis apenas nesse método.
- Variáveis de instância: Declaradas na classe, mas fora dos métodos, pertencem ao objeto.
- Variáveis de classe (static): Compartilhadas entre todas as instâncias da classe.

Escopo de Variáveis

Em Java

```
public class Exemplo {  
    static int variavelClasse = 10; // Escopo global na classe  
    int variavelInstancia;      // Pertence ao objeto  
  
    public void metodo() {  
        int variavelLocal = 5; // Apenas dentro deste método  
        System.out.println(variavelLocal);  
    }  
}
```

Escopo de Variáveis

Em Python

- Variáveis globais: Definidas fora de funções e acessíveis globalmente.
- Variáveis locais: Criadas dentro de funções, visíveis apenas nelas.
- Variáveis não locais (nonlocal): Dentro de funções aninhadas.

Escopo de Variáveis

Em Python

```
x = 10 # Variável global

def func():
    y = 5 # Variável local
    global x #nonlocal
    x = 20 # Modifica a variável global

func()
print(x) # 20
```

Nomeação de variáveis

Identificadores: O nome de uma variável é conhecido como identificador. Ele deve ser único dentro do escopo em que é utilizado.

Regras de Nomeação: As regras para nomear variáveis variam conforme a linguagem de programação, mas geralmente incluem:

- Devem começar com uma letra ou underline (_), nunca com números.
- Podem conter letras, números e underline.
- Não podem ser palavras reservadas da linguagem (como if, for, class).
- Devem ser significativos, de preferência, descrevendo o propósito da variável (idade, nome_cliente, etc.).

Boas práticas sobre variáveis

Escolha Nomes Significativos

Use nomes que descrevam o conteúdo ou o propósito da variável. Por exemplo, salario, total_vendas, media_nota.

Evite Nomes Muito Curtos ou Crípticos

Nomes como x, y, z devem ser evitados, a menos que em contextos matemáticos onde o significado é claro.



Boas práticas sobre variáveis

Considere a Imutabilidade Quando Apropriado

Em algumas situações, usar variáveis imutáveis pode prevenir erros (como reatribuições acidentais).

Minimize o Uso de Variáveis Globais

Elas podem levar a dependências ocultas e o código difícil de manter.



Variáveis são um pilar central da programação, permitindo que você armazene, manipule e utilize dados de maneira eficiente. Compreender como declarar, inicializar, e manipular variáveis é essencial para escrever código funcional e eficaz em qualquer linguagem de programação. As boas práticas associadas ao uso de variáveis ajudam a criar programas mais claros, mais fáceis de entender e manter, e menos propensos a erros.



Exemplos

Escreva um código Python que faça a soma de 2 valores inteiros



Exemplos

Escreva um código Python com uma variável com seu nome e outra com sua idade e imprima para o usuário o seu ano de nascimento.



Exemplos

Escreva um código Python com uma variável chamada valor e que receba um valor inteiro de 0 a 9.

Exiba a tabuada de SOMA desse valor.



Exemplos

Escreva um código Python com duas variáveis e:
Exiba o valor da Soma dos 2 valores;
Exiba o valor da Subtração dos 2 valores;
Exiba o valor da Multiplicação dos 2 valores;
Exiba o valor da Divisão dos 2 valores;
Exiba o valor da Potência do valor 1 pelo valor 2.



Exemplos

Escreva um código Python com uma variável nome que receba seu nome e imprima (*Olá, nome!*)



Exemplos

Escreva um código Python com duas variáveis nome e idade, que receba seu nome e sua idade, e imprima (*Olá nome, você tem idade anos!*)