

# PROGRAMAÇÃO EM PYTHON

Tipos de Dados





# CONTEÚDO PROGRAMÁTICO

## 1. INTRODUÇÃO AO PYTHON

- Histórico geral
- Ambientação
- Instalação dos aplicativos

## 2. Conceitos de Programação em Python

- Variáveis, operadores e operações de E/S
- Listas, estruturas de decisão e laços
- Funções, tuplas e dicionários
- Módulos e pacotes
- Exceções e strings
- Geradores e iteradores
- Gerenciamento de arquivos
- Módulos diversificados.

## 3. Introdução à programação orientada a objetos em Python

# CRITÉRIOS DE AVALIAÇÃO

## Presença

Presença superior  
a 75%

Máximo de 20 horas

## Atividades

Executar os  
exercícios de  
fixação

Toda aula terá exercícios  
de fixação

## Projeto Final

Um projeto que será  
implementado nas  
últimas semanas do  
curso



# TIPOS DEDADOS

Dados são as informações a serem processadas por um computador

Há três principais tipos de dados primitivos

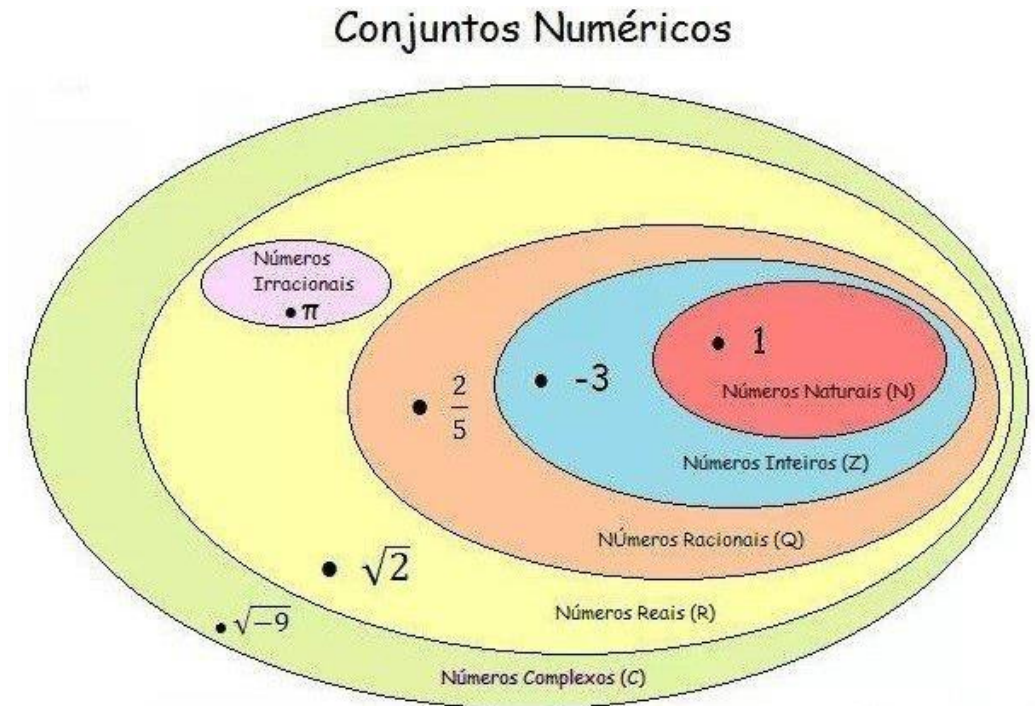
- Numéricos (Decimal ou Inteiro)
- Caracteres
- Lógicos

# TIPO NUMÉRICO

Tornando ao aspecto computacional, os dados numéricos representáveis em um computador são divididos em apenas duas classes: os INTEIROS e os REAIS.

Inteiro números inteiros, positivos e negativos

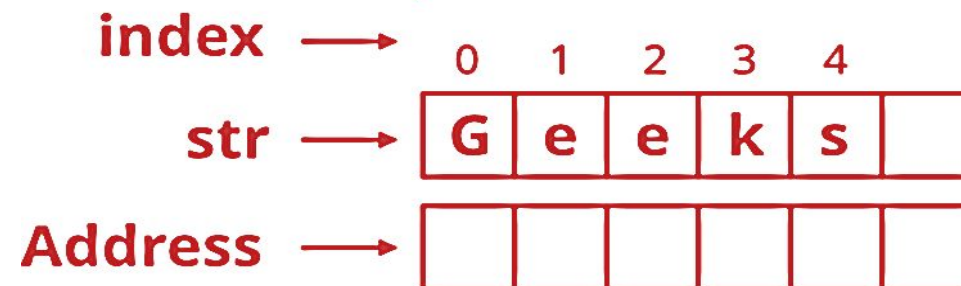
Real números fracionários, positivos e negativos



# TIPO CHARACTER

Também conhecido como tipo literal

- Se refere a uma sequência que contenham letras, números e símbolos
- Os caracteres devem ser sempre representados com aspas no código
- Pode se falar também em cadeia, **string**, alfanumérico ou char



# TIPO LÓGICO (Booleano)

São dados cujos os valores somente podem assumir

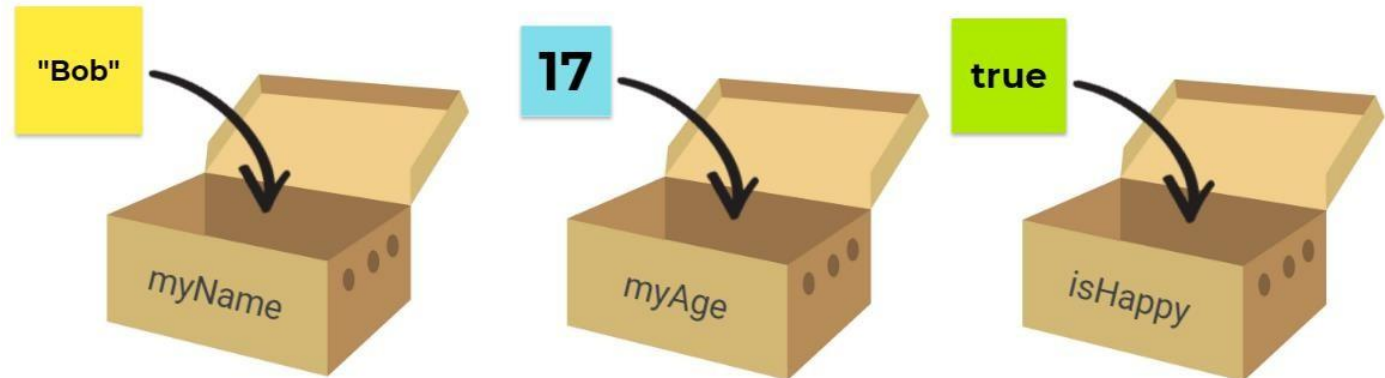
- Verdadeiro ( True)
- Falso (False)

Também conhecido como tipo **Booleano**



# VARIÁVEIS

- Uma variável é uma localização na memória RAM do computador que é utilizada para armazenar temporariamente os dados que são utilizados pelo programa
- As variáveis possuem características como:
  - Identificação
  - Endereço
  - Tipo
  - Tamanho
  - Valor





# CONSTANTES

É uma posição na memória cujo o valor não muda durante a execução do programa;

Por exemplo, o valor Pi é uma constante, pois possui sempre o mesmo valor (3,1415...)

## CONSTANTES FÍSICAS

Constante física	Símbolo	Valor
Aceleração da gravidade na superfície terrestre	$g$	$9,80 \text{ m/s}^2$
Constante eletrostática do vácuo	$k_0$	$9 \cdot 10^9 \text{ N.m}^2/\text{C}^2$
Carga elementar	$e$	$1,6 \cdot 10^{-19} \text{ C}$
Elétron-volt	$\text{eV}$	$1,6 \cdot 10^{-19} \text{ J}$
Massa do elétron	$m_e$	$9,1 \cdot 10^{-31} \text{ kg}$
Massa do próton	$m_p$	$1,67 \cdot 10^{-27} \text{ kg}$
Permeabilidade magnética do vácuo	$\mu_0$	$4\pi \cdot 10^{-7} \text{ T. m/A}$
Permitividade elétrica do vácuo	$\epsilon_0$	$8,8 \cdot 10^{-12} \text{ F/m}$
Velocidade da luz no vácuo	$c$	$3 \cdot 10^8 \text{ m/s}$
Constante de Plank	$h$	$6,63 \cdot 10^{-34} \text{ J.s}$ ou $4,14 \cdot 10^{-15} \text{ eV.s}$

# NOMENCLATURA DE VARIÁVEIS

Um nome de variável é uma sequência de letras ( $a \rightarrow z$ ,  $A \rightarrow Z$ ) e números ( $0 \rightarrow 9$ ), que devem sempre começar com uma letra. Apenas letras comuns são permitidas.

Letras acentuadas, cedilhas, espaços, caracteres especiais como \$, #, @, etc. são proibidos, exceto para o caractere \_ (sublinhado/underline).

Além dessa regra é importante também estar atento às palavras reservadas da linguagem

False	class	finally	is	return
None	continue	for	lambda	try
True	def	from	nonlocal	while
and	del	global	not	with
as	elif	if	or	yield
assert	else	import	pass	
break	except	in	raise	

# MANEIRAS DE DECLARAR VARIÁVEIS

A escolha do estilo de escrita para variáveis, funções e outros elementos do código é mais uma questão de convenção e estilo do que uma questão de funcionalidade direta. No entanto, o estilo de escrita desempenha um papel importante na legibilidade, manutenção e colaboração em projetos de programação. Aqui estão algumas formas de declarar as variáveis.

**Camel Case** – No Camel Case a primeira palavra toda fica no formato minúsculo e as primeiras letras subsequentes de palavras adicionais em maiúscula.

camelCase

**Snake Case** – No Snake Case a todas as letras minúsculas e as palavras são separadas por underline.

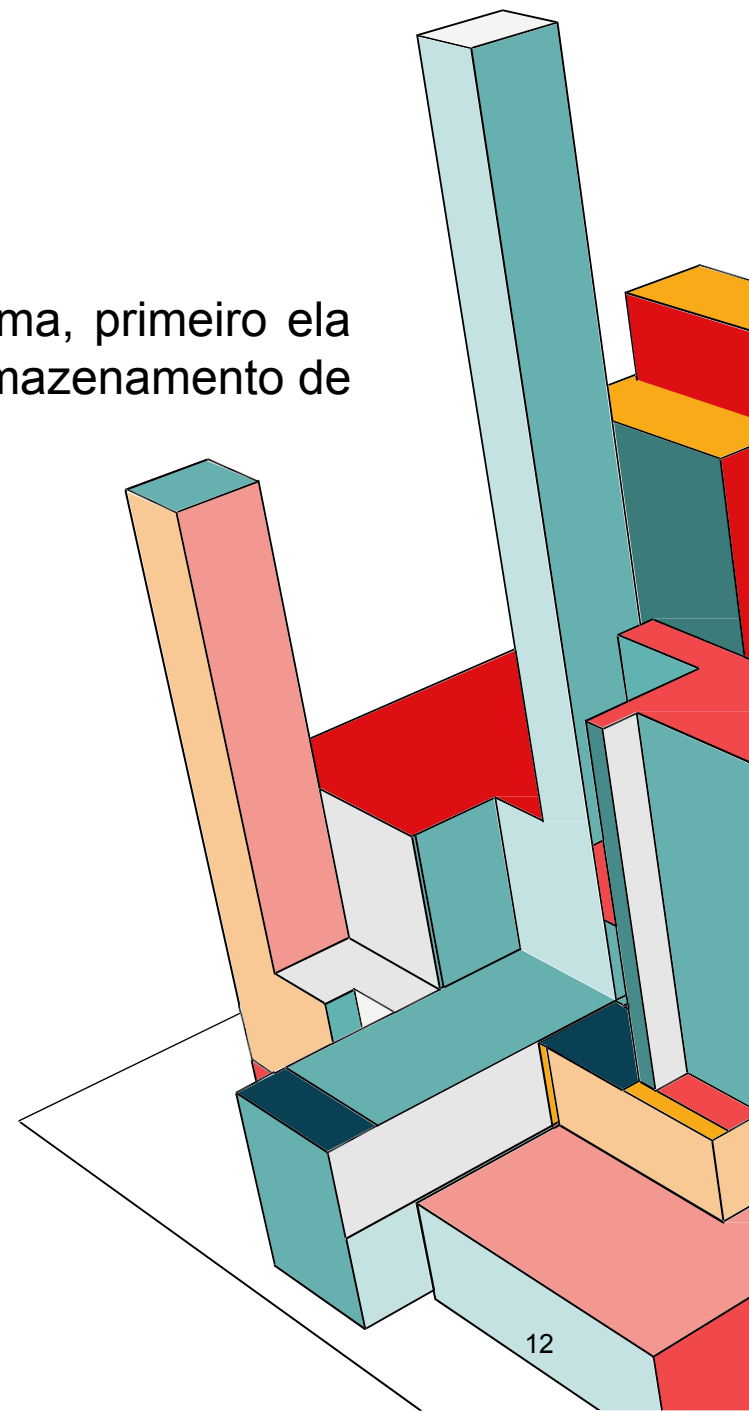
snake\_case

# DECLARAÇÃO DE VARIÁVEIS

Para que uma variável ou uma constante possa ser utilizada pelo programa, primeiro ela deve ser declarada, para que seja reservado o espaço na memória para armazenamento de seus dados

```
texto="2" #declaração de variável string
texto1= str(2); #declaração de variável do tipo string
numero=3 #declaração de variável numerica, podendo ser Inteira ou Real
numero1= int(3) #declaração de variável numerica, do tipo inteiro
numero2= float(3) #declaração de variável numerica, do tipo Real
print(texto)
print(texto1)
print(numero)
print(numero1)
print(numero2)
```

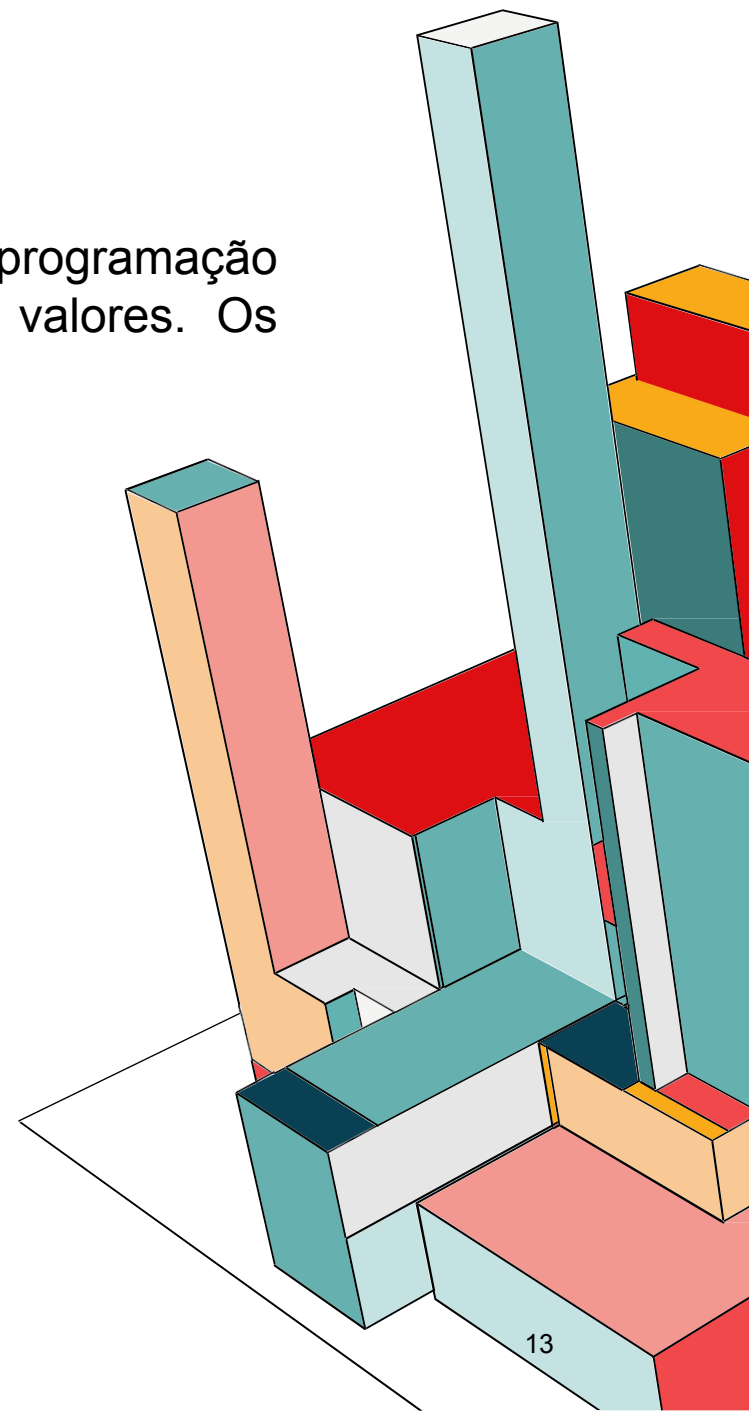
```
2
2
3
3
3.0
```



# OPERADORES ARITMÉTICOS

Os operadores aritméticos são elementos fundamentais em linguagens de programação que permitem a realização de operações matemáticas sobre variáveis e valores. Os operadores aritméticos mais comuns incluem:

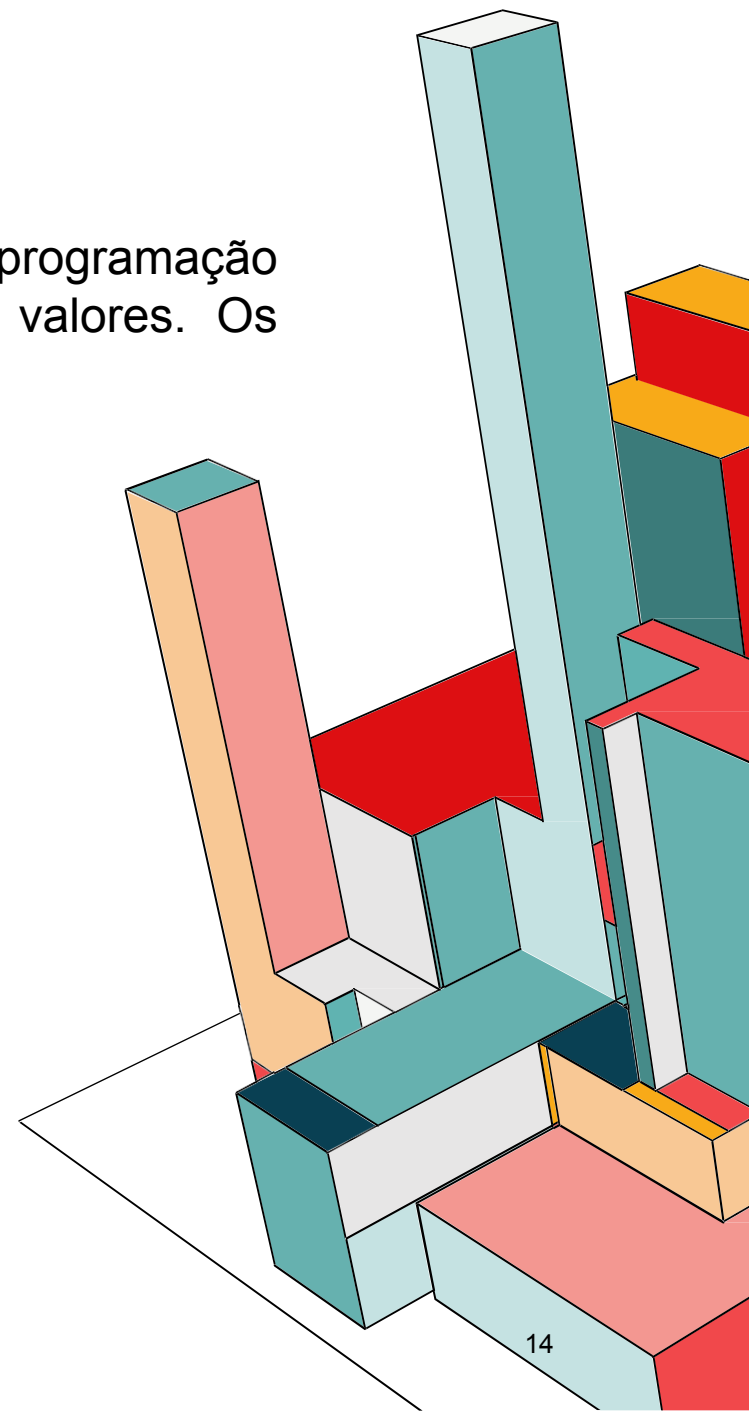
```
8  #Operador de soma
9  soma = n1 + n2
10
11 #Operador de subtração
12 subtracao = n1 - n2
13
14 #Operador de multiplicação
15 multiplicacao = n1 * n2
16
17 #Operador de divisão
18 divisao = n1 / n2
```



# OPERADORES ARITMÉTICOS

Os operadores aritméticos são elementos fundamentais em linguagens de programação que permitem a realização de operações matemáticas sobre variáveis e valores. Os operadores aritméticos mais comuns incluem:

```
20  #Operador de Divisão Inteira
21  divisaoInteira = n1 // n2
22
23  #Operador de potência
24  potencia = n1 ** n2
25
26  #Operador Módulo
27  restoDaDivisão = n1 % n2
28
```

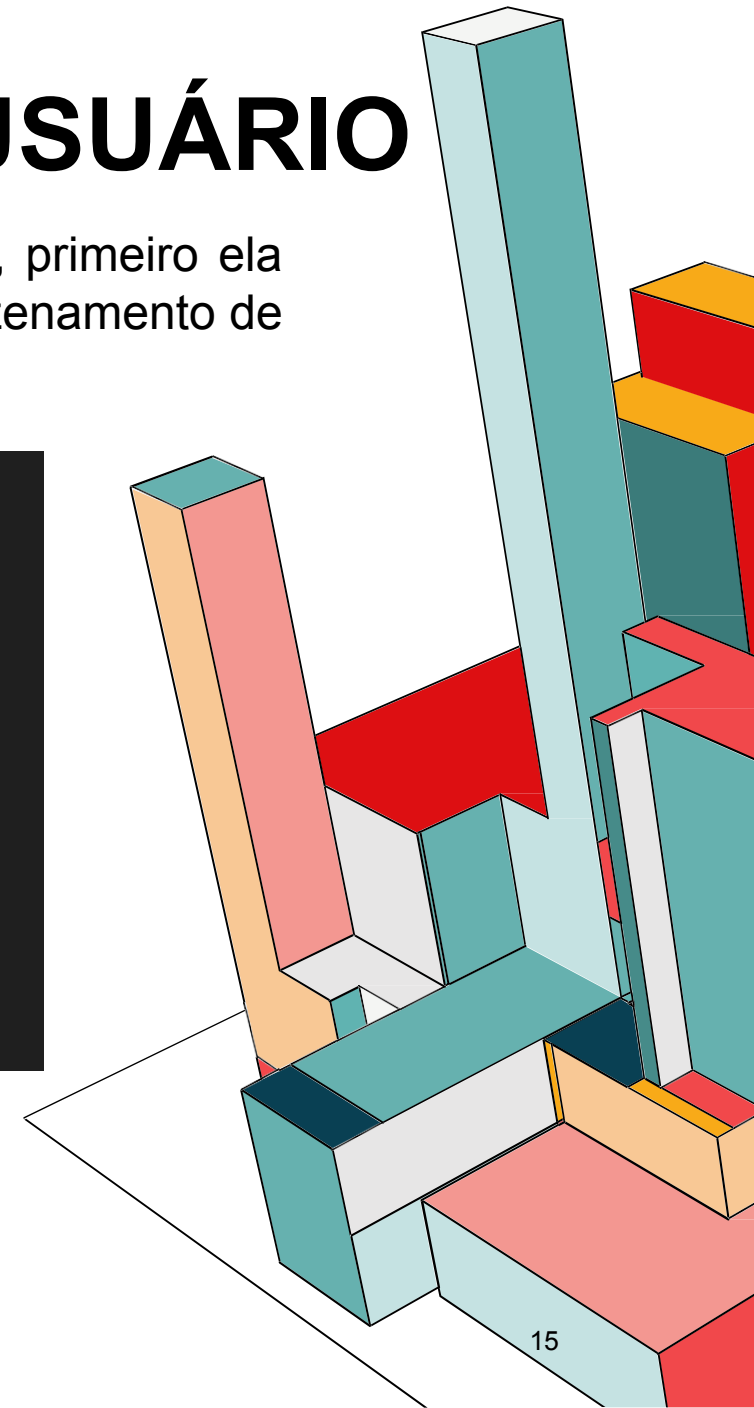




# LER VALORES INSERIDOS PELO USUÁRIO

Para que uma variável ou uma constante possa ser utilizada pelo programa, primeiro ela deve ser declarada, para que seja reservado o espaço na memória para armazenamento de seus dados

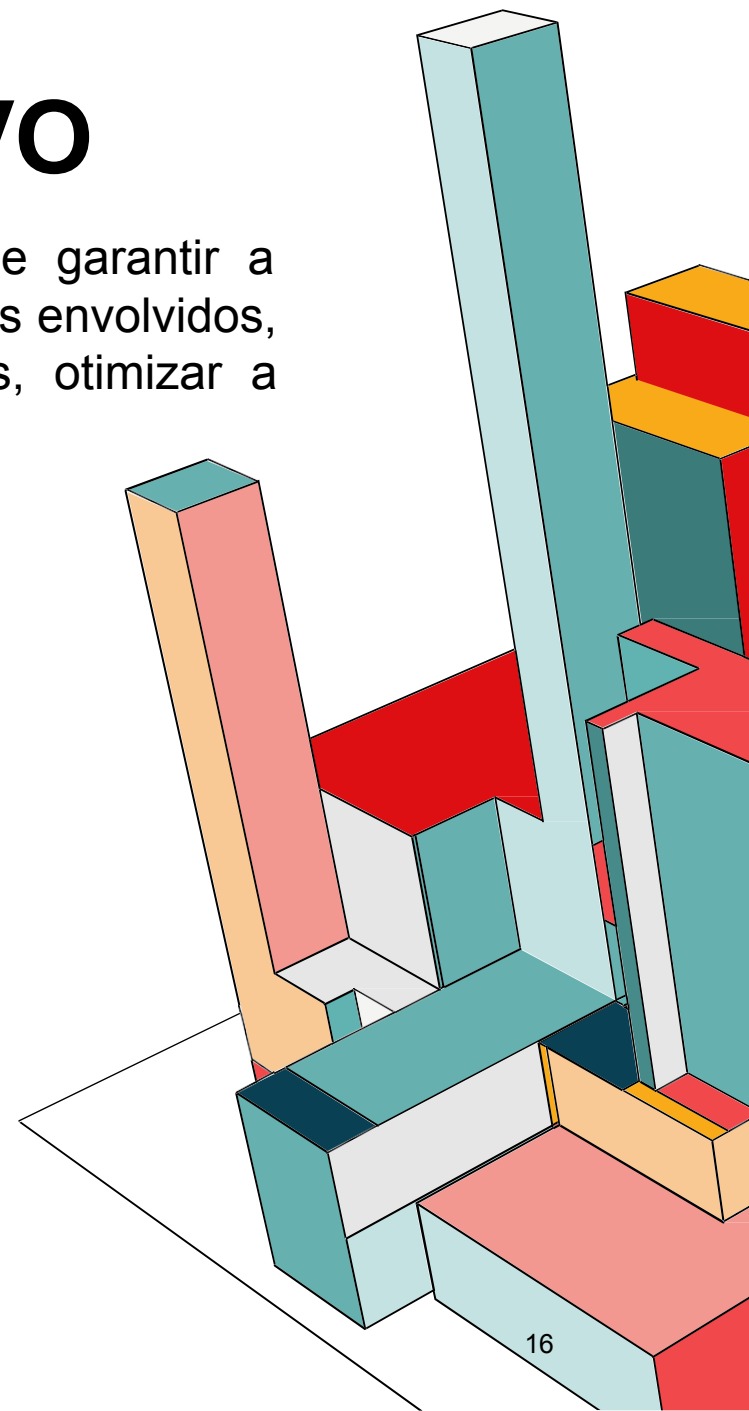
```
7  #Atribui dados digitado pelo usuario e converte em Numero Inteiro
8  numeroInteiro = int(input("Digite um numero Inteiro: "))
9
10 #Atribui dados digitado pelo usuario e converte em Numero Decimal
11 numeroReal = float(input("Digite um numero Decimal: "))
12
13 #Atribui dados digitado pelo usuario e converte em Bool
14 booleano = bool(input("Digite um valor Boleano: "))
15
16 #Atribui dados digitado pelo usuario e converte em Texto
17 string = str(input("Digite um texto: "))
18
```



# IDENTIFICAÇÃO DO TIPO PRIMITIVO

A identificação do tipo primitivo em programação desempenha o papel de garantir a integridade, eficiência e compreensão do código. Ao conhecer os tipos de dados envolvidos, os programadores podem realizar operações apropriadas, validar entradas, otimizar a alocação de memória e escolher estruturas de dados eficientes.

```
19  #Identificar o tipo primitivo
20  print(type(numeroInteiro))
21  print(type(numeroReal))
22  print(type(booleano))
23  print(type(string))
```

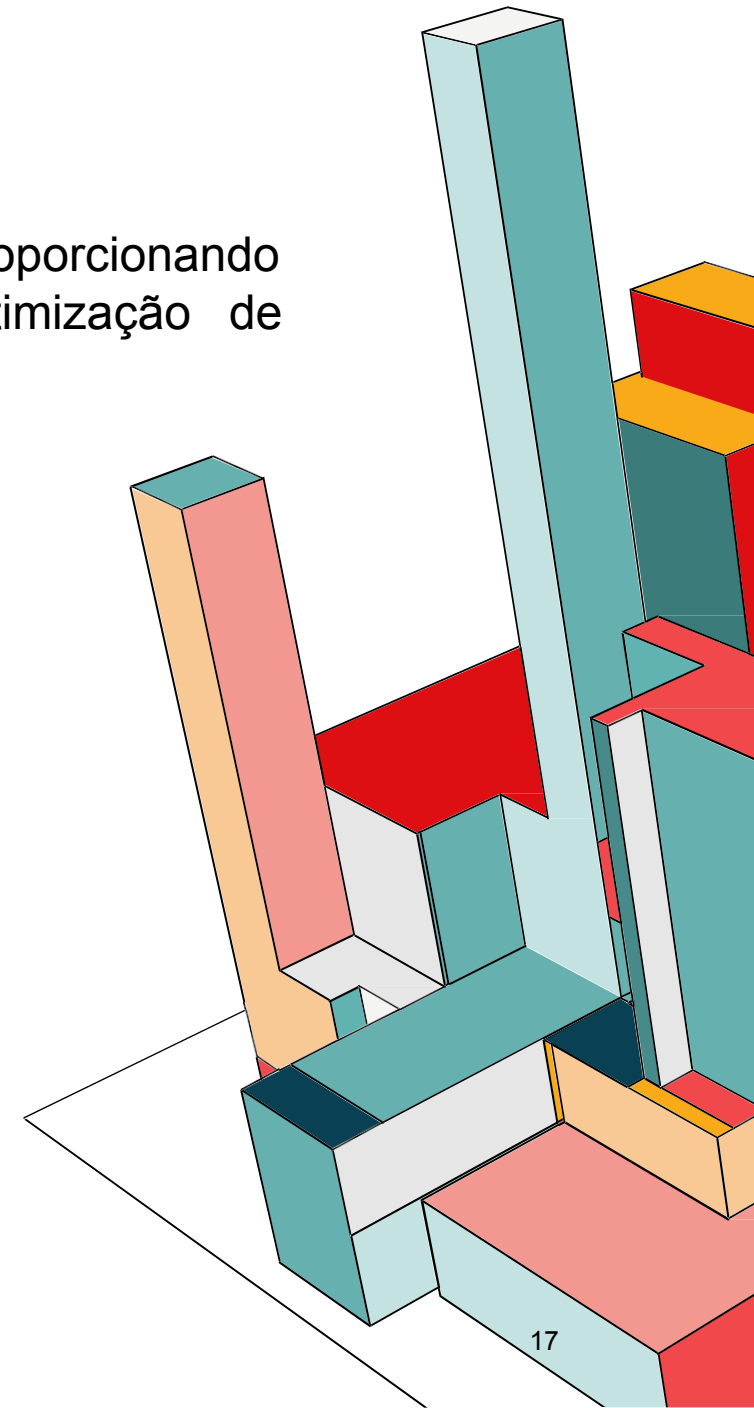




# ANÁLISE DA VARIÁVEL

A análise de variáveis é importante no desenvolvimento de software, proporcionando compreensão do comportamento dos dados, identificação de erros, otimização de desempenho e validação de entrada.

```
3  #Identifica se o valor é do tipo numerico
4  print(n.isnumeric())
5
6  #Identifica se o valor é do tipo string
7  print(n.isalpha())
8
9  #Identifica se o valor é do tipo alfanumerico
10 print(n.isalnum())
11
12 #Identifica se o valor esta com as letras maiusculas
13 print(n.isupper())
14
15 #Identifica se o valor esta com as letras minusculas
16 print(n.islower())
17
18 #Identifica se o valor não tem casas decimais
19 print(n.isdecimal())
```



# VAMOS PRATICAR

