



# Módulo 8





A decorative graphic on the left side of the slide consists of a large central cyan hexagon with a white number '1' inside. Surrounding this central hexagon are several smaller hexagons of varying shades of blue and cyan. Some of these smaller hexagons contain white icons: a lightbulb, a thumbs-up, a smartphone, a magnifying glass, and a gear. There is also a network-like icon with a central node and connecting lines.

1

# Variáveis, tipos e operadores

Variáveis básicas



# Variáveis básicas

## Tipo primitivos

Tipo	Descrição
int	Números inteiros de 32 bits.
float	Números em notação de ponto flutuante.
boolean	Pode assumir dois valores: true e false
String	Armazena caractere em notação de 16 bits, aceita dados alfanuméricos.
byte	Aloca número inteiro de 8 bits.
short	Armazena números inteiros de 16 bits.
long	Aloca números inteiros de 64 bits.
double	Números em notação de ponto flutuante.



A decorative pattern of hexagons in various shades of blue and cyan. Some hexagons contain white icons: a lightbulb, a thumbs-up, a smartphone, a magnifying glass, and a gear. A network of dots is also visible on the left side.

# 2

## Variáveis de classes e métodos

Variáveis básicas



# Variáveis de classes e métodos/local

Uma variável de classe pode ser acessada no escopo de classe e de método, e a variável local só pode ser acessada no escopo de método.



A decorative graphic on the left side of the slide. It features a large central hexagon with a blue-to-cyan gradient, containing the number '3'. Surrounding this central hexagon are several smaller hexagons of varying shades of blue and cyan. Some of these smaller hexagons contain white icons: a lightbulb, a thumbs-up, a smartphone, a magnifying glass, and a gear. There is also a network-like icon with a central node and five connecting lines, and a speech bubble icon. The entire graphic is set against a dark blue background.

3

Operadores



# Operadores aritméticos

Os operadores são sinais que representam atribuições, cálculos e ordem dos dados. As operações seguem uma ordem de prioridades, ou seja, alguns cálculos são processados antes de outros.








# Operadores aritméticos

Ordem	Operadores	Operação
1	+ -	Soma e subtração
2	/ *	Divisão e multiplicação

Exemplo  $(10 + 5) * 10$





# Operadores aritméticos

`+, -, *, /`

```
int num1 = 10;
```

```
int num2 = 20;
```

```
int num3 = num1 + num2;
```

```
int num4 = num1 - num2;
```

```
int num5 = num1 * num2;
```

```
int num6 = num1 / num2;
```





# Operadores de atribuição

`=, +=, -=, *=, /=`

Exemplos:

Integer num1 = 2;


Integer num2 = 2;

num1 += num2; é igual á num1 = num1 + num2;

num1 -= num2; é igual á num1 = num1 – num2;

num1 \*= num2; é igual á num1 = num1 \* num2;

num1 /= num2; é igual á num1 = num1 / num2;





# Operadores de incremento e decremento

São os operadores que nos permitem incrementar as variáveis em uma unidade. Podem ser usados diante ou atrás da variável.

`++` ou `--`

```
int num1 = 10;
```

```
int num2 = 10;
```

`num1++`; é igual á `num1 + 1`;

`num2--`; é igual á `num2 - 1`;





# Operadores de incremento e decremento


```
int num1 = 10;
```

```
int num2 = 10;
```

`++num1`; é igual á `num1 + 1`;

`--num2`; é igual á `num2 - 1`;

Diferente do anterior neste exemplo ele incrementa ou decrementa a variável antes de fazer qualquer coisa.





# Operadores relacionais

<, <=, >, >=, ==, !=





# Operadores lógicos

&&(and) , || (or), ! (not)





# Referências

