

Variáveis e tipos de variáveis.

# Nesta aula

- Ler módulo 3,4
- PAPR
- Exercícios

# Variáveis e tipos de variáveis.

- Para calcular um valor para uma expressão única com certas entradas, e fazer isso apenas uma vez, não é necessário escrever um programa de software completo.
- No entanto, se o processo dos cálculos precisar ser realizado frequentemente e com diferentes entradas a cada vez, faz sentido automatizar esse processo.

# O formato do operador de atribuição

- `<Nome da variável>:=<expressão>;`

# Variáveis

- Você pode usar variáveis no Delphi para diversas tarefas e motivos. As variáveis precisam ser definidas antes de poderem ser usadas.
- A palavra-chave `var` é usada para iniciar uma seção de definições da variável, que está localizado após a definição do procedimento e antes da palavra-chave `begin`.
- `var`
- `<Nome da variável>:=<tipo de dados>;`

# Variáveis

- O nome de uma variável é uma sequência de caracteres com a seguinte sintaxe:
  - Pode conter letras, números (0 a 9) ou sublinhado;
  - Deve começar com uma letra
- O nome da variável pode ter qualquer comprimento (o número de caracteres não é limitado) e não diferencia maiúsculas de minúsculas.
- Ao pensar sobre a nomenclatura de uma variável, observe que você deve evitar nomes de um caractere, exceto para variáveis de contador de loop ou temporárias

# Variáveis

- As variáveis de contador de loop são os nomes I e J. Outras instâncias de nomes de variáveis de um caractere são S (string) ou R (raio).
- Os nomes de variáveis de um caractere sempre devem estar em letra maiúscula, mas é sempre preferível usar nomes mais significativos

# Inteiros

Tipo de dados	Intervalo	Representação
Integer	-2.147.483.648 a 2.147.483.647	32 bits, com sinal
Cardinal	0 a 4.294.967.295	32 bits, inteiros positivos
ShortInt	-128 a 127	8 bits, com sinal
SmallInt	-32.768 a 32.767	16 bits, com sinal
LongInt	-2.147.483.648 a 2.147.483.647	32 bits, com sinal
Int64	$-2^{63}$ a $2^{63}-1$	64 bits, com sinal
Byte	0 a 255	8 bits, sem sinal
Word	0 a 65535	16 bits, sem sinal
LongWord	0 a 4.294.967.295	32 bits, sem sinal



# Real

Tipo de dados	Intervalo	Dígitos significativos (precisão)	Bytes
Real48	$\pm 2,9 \times 10^{-39}$ a $\pm 1,7 \times 10^{38}$	11 a 12	6
Real	$\pm 5,0 \times 10^{-324}$ a $\pm 1,7 \times 10^{308}$	15 a 16	8
Single	$\pm 1,5 \times 10^{-45}$ a $\pm 3,4 \times 10^{38}$	7 a 8	4
Double	$\pm 5,0 \times 10^{-324}$ a $\pm 1,7 \times 10^{308}$	15 a 16	8
Extended	$\pm 3,6 \times 10^{-4951}$ a $\pm 1,1 \times 10^{4932}$	19 a 20	10
Comp	$-2^{63}$ a $2^{63} - 1$	19 a 20	8
Currency	-922337203685477,5808 a +922337203685477,5807	19 a 20	8

# Expressões aritméticas

- Vamos analisar os operadores de atribuição novamente. O tipo de dados à esquerda deve corresponder ao tipo de dados do valor da expressão aritmética à direita

**Operadores usados em expressões aritméticas:**

Operador	Nome do operador	Tipos de dados dos operandos	Tipo de dados do resultado
+	Adição	Pelo menos um dos operandos é real	Real
		Integer	Integer
-	Subtração	Pelo menos um dos operandos é real	Real
		Integer	Integer
*	Multiplicação	Pelo menos um dos operandos é real	Real
		Integer	Integer
/	Divisão	Real, Integer	Real
<b>Div</b>	Divisão de integer	Integer	Integer
<b>Mod</b>	Resto da divisão de integer	Integer	Integer

# Funções matemáticas padrão

Função	Computação	Parâmetros de entrada	Tipo de resultado	Exemplos
ABS (X)	Retorna o valor absoluto de um número.	X — valor REAL ou INTEGER	Igual ao tipo de parâmetro	ABS (2.0) = 2.0000e+00;
SQR (X)	Retorna o quadrado de um número.	X — valor REAL ou INTEGER	Igual ao tipo de parâmetro	SQR (3) = 9; SQR (-2.0) = 4.0000e+00;
SQRT (X)	Retorna a raiz quadrada de um número.	X — valor REAL ou INTEGER	REAL	SQRT (16) = 4.0000e+00; SQRT (25.0) = 5.0000e+00;
EXP (X)	Retorna a exponenciação de um número.	X — valor REAL ou INTEGER	REAL	EXP (0) = 1.00000e+00; EXP (-1.0) = 3.67879e-01;
LN (X)	Retorna o logaritmo natural de um número.	X — valor REAL ou INTEGER	REAL	LN (1) = 0.00000e+00 LN (7.5) = 2.01490e+00
SIN (X)	Retorna o seno do parâmetro.	X — valor REAL ou INTEGER, em radianos	REAL	SIN (0) = 0.00000e+00; SIN (1.0) = 8.41471e-01
COS (X)	Retorna o cosseno do parâmetro.	X — valor REAL ou INTEGER, em radianos	REAL	COS (0) = 1.00000e+00; COS (1.0) = 8.41471e-01
ARCTAN (X)	Retorna o arco tangente do parâmetro.	X — valor REAL ou INTEGER	REAL	ARCTAN (0) = 0.0000e+00 ARCTAN (-1.0) = -7.8539e-01

Função	Computação	Parâmetros de entrada	Tipo de resultado	Exemplos
ROUND (X)	Arredonda um valor real para o valor inteiro mais próximo (por valor absoluto).	X — REAL	INTEGER	ROUND(3.1) = 3; ROUND(-3.1) = -3; ROUND(3.8) = 4; ROUND(-3.8) = -4;  Observação: números com partes fracionárias iguais a 0,5 são arredondados para o número par mais próximo.  ROUND (3.5) = 4; ROUND (2.5) = 2;
TRUNC (X)	Retorna a parte inteira do parâmetro eliminando sua parte fracionária.	X — REAL	INTEGER	TRUNC (3.1) = 3; TRUNC (-3.1) = -3; TRUNC (3.8) = 3;
INT (X)	Retorna a parte inteira do parâmetro eliminando sua parte fracionária.	X — REAL	REAL	INT (3.1) = 3.00000E+00  INT (-3.1) = -3.00000E+00  INT (3.8) = 3.00000E+00

# Passos

- Crie uma aplicação do tipo Windows VLC application
- Renomeie o formulário para frmPrincipal
- Adicione o nome do caption com o título “Trabalhando com Número”
- Adicione um Tedit e renomeie para edtNum1
- Adicione outro Tedit e renomeie para edtNum2
- Adicione um botão e renomeie para btnSomar
- Adicione um label com lblResultado

# Passos

```
procedure TfrmPrincipal.btnSomarClick(Sender: TObject);  
var  
    resultado, num1,num2: Integer;  
begin  
    num1:=strToInt(edtNum1.Text);  
    num2:=strToInt(edtNum2.Text);  
    resultado:=num1+num2;  
    lblResultado.caption:='RESULTADO: '+  
    intToStr(resultado);  
end
```

# Passos : Quociente da divisão

- Adicione um botão chamado btnDiv
- E adicione no onclick do botão o código abaixo:

```
procedure TfrmPrincipal.btnDivClick(Sender: TObject);  
var  
    resultado, num1,num2: Integer;  
begin  
    num1:=strToInt(edtNum1.Text);  
    num2:=strToInt(edtNum2.Text);  
    resultado:= num1 div num2;  
    lblResultado.caption:='RESULTADO: ' + intToStr(resultado);  
end;
```

# Passos : Quadrado do primeiro número

- Adicione um botão chamado btnQuadrado
- No onclick do botão btnQuadrado adicione o código abaixo:

```
procedure TfrmPrincipal.btnQuadradoClick(Sender: TObject);  
var  
    quadrado1,quadrado2, num1,num2: Real;  
begin  
    num1:=strToFloat(edtNum1.Text);  
    num2:=strToFloat(edtNum2.Text);  
    quadrado1:=sqr(num1);  
    quadrado2:=sqr(num2);  
    lblResultado.caption:='RESULTADO: O quadrado do núm 1 é '+  
        floatToStr(quadrado1)+' o quadrado do núm 2 é '+floatToStr(quadrado1) ;  
end;
```