

LABORATÓRIO 03 (20/08/2007)

Objetivos: Capturar valores em campos de texto e realizar conversões de tipos e operações matemáticas.

Recursos: Aula no computador, quadro e pincel, roteiro de aula.

Palavras-chave: Números inteiros, números reais, operadores aritméticos.

OPERADORES MATEMÁTICOS EM JAVA

Podemos realizar diversos cálculos matemáticos em Java, alguns utilizam operadores simples (ver tabela) outros podem se utilizar de classes predefinidas, ou o programador pode escrever sua própria função.

Os operadores matemáticos usados em Java são:

=	ATRIBUIÇÃO	Usado para colocar um valor numa variável, como em <code>int x = 10;</code>
+	SOMA	Soma dois valores. Ex.: <code>x = 5 + 8;</code> resulta em 13.
-	SUBTRAÇÃO	A diferença entre dois valores. Ex.: <code>x = 8 - 5;</code> resulta em 3.
*	MULTIPLICAÇÃO	Multiplifica dois valores. Ex.: <code>x = 5 * 8;</code> resulta em 40.
/	DIVISÃO	Divide o primeiro valor pelo segundo. Ex.: <code>x = 10 / 5;</code> resulta em 2.
%	MÓDULO (RESTO)	Guarda o resto de uma divisão entre dois números inteiros. Ex.: <code>x = 10 / 3;</code> resulta em 1, pois 10 dividido por 3 dá 3 e sobra 1.

Outros operadores:

- `++` Realiza o incremento de 1 em 1. Ex.: `x++;` x será 1, depois 2, depois 3, etc.
- `--` Realiza o decremento de 1 em 1. Ex.: `x--;` x será 3, depois 2, depois 1, etc.
- `+=` Realiza o incremento de acordo com o valor informado. Ex.: `x += 2;` x será 2, depois 4, depois 6, etc.
- `-=` Realiza o decremento de acordo com o valor informado. Ex.: `x -= 2;` x será 10, depois 8, depois 6, etc.
- `*=` Realiza multiplicações sucessivas de acordo com o valor. Ex.: `x *= 2;` x será 2, depois 4, depois 8 depois 16...
- `/=` Realiza divisões sucessivas de acordo com o valor. Ex.: `x /= 2;` x será 24, depois 12, depois 6 depois 3...

CONVERTENDO STRING EM NUMEROS

Ao capturarmos informações contidas em campos `TextField` através do método `getText()`, os dados serão gravados dentro de variáveis `String` e não poderão ser operadas, pois o operador `+`, por exemplo, é usado para concatenação (ver aula 2). Para operar os valores, primeiro precisamos converter para números propriamente ditos. Veja o código abaixo inserido num botão do programa:

```
String valor1 = jTextField1.getText();
```

a linha 1 captura o texto como uma `String`;

```
int numerol = Integer.parseInt(valor1);
```

a linha 2 usa o método `parseInt` para converter a `String` em `Inteiro`

```
int dobro = numerol * 2;
```

```
valor1 = Integer.toString(dobro);
```

a linha 3 realiza o cálculo

```
jTextField1.setText(valor1);
```

a linha 4 faz o inverso da linha 2, convertendo de `inteiro` para `String`

a linha 5 mostra o resultado no campo de texto.

Em resumo, para realizar um cálculo é preciso: capturar o valor, converter de string para inteiro, efetuar o cálculo, converter de inteiro para string e exibir o resultado. Pode-se também usar a classe Double como no exemplo abaixo, que mostra uma forma resumida de realizar o processo:

```
Double x = Double.parseDouble(jTextField2.getText);  
  
Double quadrado = x * x;  
  
jTextField2.setText(Double.toString(quadrado));
```

VARIÁVEIS GLOBAIS

Em alguns programas, precisamos manter o valor dos dados quando o código de um botão é encerrado. Podemos fazer isso declarando as variáveis fora do código dos botões, de preferência na primeira linha após a declaração da classe. Ex.:

```
public class Variaveis{  
  
    int variável_global = 0;  
  
private void botao(java.awt.event.ActionEvent evt){  
  
    int variavel_local = 10;  
  
    variavel_global += 10;  
  
    }  
  
}
```

Nesse código, se pedirmos para exibir os valores da variável local e da variável global a cada clique no botão, veremos que `variavel_local` sempre exibirá o valor 10, porém `variavel_global` irá exibir 10, depois 20, depois 30 e assim por diante, pois o valor dela não é reinicializado dentro do botão.

EXERCÍCIOS

1. Crie um programa que simule uma calculadora com as 4 operações básicas. Sugestão. Use três campos de texto e cinco botões (um pra cada operação e outro para limpar os campos).
2. Modifique o programa anterior, removendo o botão limpar e adicionando os botões M+, que servem para transferir o total para uma variável chamada memória, e M-, que traz o valor na memória para o primeiro campo onde você digita valores para que seja possível usar o total atual em novos cálculos. Neste caso, sempre que você clicar numa das operações (+, -, * ou /) os campos dos valores são apagados, exceto o total.