

Objetivos: Utilizar estruturas de repetição para reaproveitar códigos;

Recursos: Aula em laboratório com uso do computador; Quadro e pincel; Roteiro de Aula;

ESTRUTURAS DE REPETIÇÃO

Ao criar programas, existem operações que se repetem diversas vezes e para evitar escrevermos código redundante, podemos utilizar estruturas de repetição. Assim como em C, Java utiliza 3 estruturas básicas `while` (enquanto) `do..while` (faça..enquanto) e `for` (para).

Observe os três exemplos abaixo que realizam a mesma função:

Arquivo: Repete1.java

```
1. int numero = 1, soma = 0;
2. while (numero < 10){
3.     soma += numero;
4.     numero++;
5. }
6. JOptionPane.showMessageDialog(null, "Total = " + soma);
```

A estrutura **while** realiza um teste lógico – que está descrito entre os parênteses – antes de qualquer comando ser executado. Se este teste for verdadeiro, os comandos dentro das chaves serão executados. Se o teste for falso, todo o bloco é ignorado, e neste caso, executa o comando da linha 6. Neste exemplo os números de 1 a 9 serão somados.

Arquivo: Repete2.java

```
1. int numero = 10, soma = 0;
2. do{
3.     soma += numero;
4.     numero++;
5. } while (numero < 10);
6. JOptionPane.showMessageDialog(null, "Total = " + soma);
```

A estrutura **do..while** executa os comandos da estrutura uma primeira vez antes de realizar o teste. Se este teste for verdadeiro, os comandos dentro das chaves serão repetidos. Se o teste for falso, o programa segue adiante, neste caso, indo para a linha 6. Observe que neste exemplo a variável `numero` começa com valor 10 e aumenta para 11 na linha 4, ou seja, na hora do teste ele será falso e o bloco não se repetirá.

Arquivo: Repete3.java

```
1. int soma = 0;
2. for (int numero = 5; numero < 10; numero++){
3.     soma += numero;
4. }
5. JOptionPane.showMessageDialog(null, "Total = " + soma);
```

A estrutura `for` é composta de três partes: a inicialização das variáveis, onde declaramos as variáveis que serão usadas no laço; depois o teste lógico, semelhante às demais estruturas é quem garante que o comando será repetido; e finalmente o incremento.

Observe que nas três estruturas aparece o trecho **numero++**, que aumenta em uma unidade o valor desta variável. Este incremento é indispensável para que o programa possa sair do laço, ou estaremos no que chamamos de loop infinito, onde a condição de parada nunca é atingida.

Outra forma de entrar e sair um laço é usando uma variável *boolean*. Veja o exemplo:

Arquivo: Repete4.java

```
boolean executa = true;
int cont = 0;
while (executa){
    JOptionPane.showMessageDialog(null, "Contador = " + cont);
    cont ++;

    int opcao =
    JOptionPane.showConfirmDialog(null, "Continuar?", "Escolha",
        JOptionPane.YES_NO_OPTION);

    if (opcao == 1) {
        executa = false;
    }
}
```

Neste código, o que define se o programa repetirá não é o valor do contador e sim o valor da variável *executa*. Observe que em cada interação, o sistema pergunta se deseja continuar, caso escolha NÃO, o valor da variável *executa* é modificado e o laço não repetirá mais.

É possível ainda utilizar um laço de repetição dentro de outro, em casos onde é necessário que uma variável só seja atualizada quando outra atingir um determinado valor. Por exemplo ao trabalhar com vetores ou se você precisa escrever a tabuada de multiplicação (1 a 10).

```
String tabuada = "";

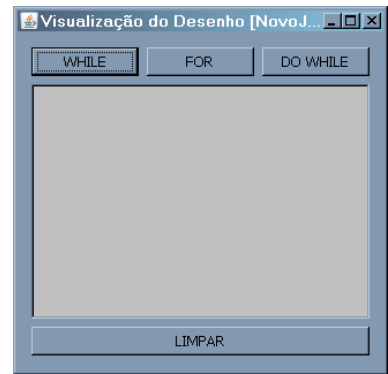
for (int x = 1; x<=10; x++){
    for (int y = 1; y <=10; y++){
        tabuada += " " + x + " * " + y + " = " + (x*y) + "\n";
    }
    Tabuada += "\n";
}
```

O código acima armazena a tabuada de 1 (1x1 = 1, 1x2 = 2, etc), antes de começar a tabuada de 2 (2 x 1 = 2, 2x2 = 4, etc). Compile o código acima exibindo o resultado num *TextArea* e veja como fica.

EXERCÍCIOS

1. Crie um programa que escreva numa caixa de texto os números de 1 a 200 separados por vírgula. OBS.: Após o número 200 não deve haver vírgula. Utilize três botões, em cada um deles use uma estrutura diferente.

Ex.: 1, 2, 3, 4...198, 199, 200



2. Faça um programa que crie as tabuadas de 1 a 10 usando as estruturas while e do while.