

# Curso Java COMPLETO

**Capítulo: Estruturas repetitivas**

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

## Como utilizar o Debug no Eclipse (execução passo a passo)

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

## Como executar o debug do Eclipse

- Para marcar uma linha de breakpoint:
  - Run -> Toggle Breakpoint
- Para iniciar o debug:
  - Botão direito na classe -> Debug as -> Java Application
- Para executar uma linha:
  - F6
- Para interromper o debug:



```
import java.util.Locale;
import java.util.Scanner;

public class Main {

    public static void main(String[] args) {

        Locale.setDefault(Locale.US);
        Scanner sc = new Scanner(System.in);

        double largura = sc.nextDouble();
        double comprimento = sc.nextDouble();
        double metroQuadrado = sc.nextDouble();

        double area = largura * comprimento;
        double preco = area * metroQuadrado;

        System.out.printf("AREA = %.2f%n", area);
        System.out.printf("PRECO = %.2f%n", preco);

        sc.close();
    }
}
```

# Estrutura repetitiva "enquanto" (while)

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

## Estrutura repetitiva "enquanto"

É uma **estrutura de controle** que **repete** um bloco de comandos **enquanto** uma **condição** for verdadeira.

**Quando usar:** quando **não** se sabe previamente a quantidade de repetições que será realizada.

### Problema exemplo:

Fazer um programa que lê números inteiros até que um zero seja lido. Ao final mostra a soma dos números lidos.

Entrada	Saída
5 2 4 0	11

## Sintaxe / regra

```
while ( condição ) {  
    comando 1  
    comando 2  
}
```

**Regra:**

**V: executa e volta**

**F: pula fora**

## Resumo da aula

- Estrutura repetitiva "enquanto"
- Recomendada quando não se sabe previamente a quantidade de repetições
- Regra:
  - V: executa e volta
  - F: pula fora

# Exercício de testes de mesa com while

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

```
x = 5;  
y = 0;  
while (x > 2) {  
    System.out.print(x);  
    y = y + x;  
    x = x - 1;  
}
```

--	--	--

**x**      **y**      **i**

Tela:

--

```
x = 2;  
y = 0;  
while (x <60) {  
    System.out.println(x);  
    x = x * 2;  
    y = y + 10;  
}
```

--	--	--

**x**      **y**      **i**

Tela:

--

```
x = 100;  
y = 100;  
while (x != y) {  
    System.out.print("olha");  
    x = Math.sqrt(y);  
}
```

--	--	--

**x**      **y**      **i**

Tela:

--

```

x = 0;
while (x < 5) {
    y = x * 3;
    System.out.print(y);
    x = x + 1;
}
System.out.println("Fim");

```

<b>x</b>	<b>y</b>	<b>i</b>

Tela:

```

x = 2;
y = 10;
System.out.println("Olá");
while (x < y) {
    System.out.println(x + "-" + y);
    x = x * 2;
    y = y + 1;
}

```

<b>x</b>	<b>y</b>	<b>i</b>

Tela:

```

x = 4;
y = 0;
i = 0;
while (i < x) {
    i = i + 1;
    y = y + i;
    System.out.print(i);
    System.out.println(y);
}

```

<b>x</b>	<b>y</b>	<b>i</b>

Tela:

```

x = 5;
y = 0;
while (x > 2) {
    System.out.print(x);
    y = y + x;
    x = x - 1;
}

```

<del>5 4 3</del> 2	<del>0 5 9</del> 12	
<b>x</b>	<b>y</b>	<b>i</b>

Tela:

5 4 3

```

x = 2;
y = 0;
while (x < 60) {
    System.out.println(x);
    x = x * 2;
    y = y + 10;
}

```

<del>2 4 8</del> <del>16 32 64</del>	<del>0 10 20</del> <del>30 40 50</del>	
<b>x</b>	<b>y</b>	<b>i</b>

Tela:

2  
4  
8  
16  
32

```

x = 100;
y = 100;
while (x != y) {
    System.out.print("olha");
    x = Math.sqrt(y);
}

```

100	100	
<b>x</b>	<b>y</b>	<b>i</b>

Tela:

<pre>x = 0; while (x &lt; 5) {     y = x * 3;     System.out.print(y);     x = x + 1; } System.out.println("Fim");</pre>	<pre>x = 2; y = 10; System.out.println("Olá"); while (x &lt; y) {     System.out.println(x + "-" + y);     x = x * 2;     y = y + 1; }</pre>	<pre>x = 4; y = 0; i = 0; while (i &lt; x) {     i = i + 1;     y = y + i;     System.out.print(i);     System.out.println(y); }</pre>
<div> <div><del>0 1 2</del> 3 4 5</div> <div><del>0 3 6</del> 8 12</div> <div></div> </div> <div>x      y      i</div>	<div> <div><del>2 4 8</del> 16</div> <div><del>10 11 12</del> 13</div> <div></div> </div> <div>x      y      i</div>	<div> <div>4</div> <div><del>0 1 3</del> 8 10</div> <div><del>0 1 2</del> 3 4</div> </div> <div>x      y      i</div>
<p>Tela:</p> <div>0 3 6 9 12 Fim</div>	<p>Tela:</p> <div>Olá 2-10 4-11 8-12</div>	<p>Tela:</p> <div>1 1 2 3 3 6 4 10</div>

## Estrutura repetitiva "para" (for)

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

## Estrutura repetitiva "para"

É uma **estrutura de controle** que **repete** um bloco de comandos **para** um certo **intervalo de valores**.

Quando usar: quando se sabe previamente a quantidade de repetições, ou o intervalo de valores.

**Por exemplo:**

Fazer um programa que lê um valor inteiro N e depois N números inteiros. Ao final, mostra a soma dos N números lidos

Entrada	Saída
3 5 2 4	11

## Sintaxe / regra

Executa somente na primeira vez

V: executa e volta  
F: pula fora

Executa toda vez depois de voltar

```
for ( início ; condição ; incremento) {  
    comando 1  
    comando 2  
}
```



## Importante

Perceba que a estrutura "para" é ótima para se fazer uma repetição baseada em uma CONTAGEM:

```
for (int i=0; i<5; i++) {  
    System.out.println("Valor de i: " + i);  
}
```

Resultado na tela:

```
Valor de i: 0  
Valor de i: 1  
Valor de i: 2  
Valor de i: 3  
Valor de i: 4
```

## Contagem regressiva

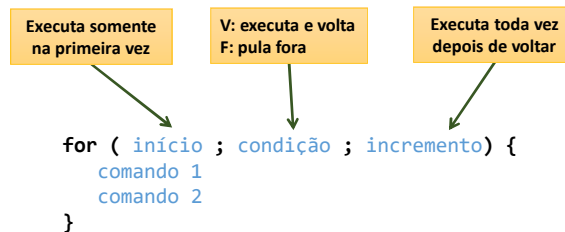
```
for (int i=4; i>=0; i--) {  
    System.out.println("Valor de i: " + i);  
}
```

Resultado na tela:

```
Valor de i: 4  
Valor de i: 3  
Valor de i: 2  
Valor de i: 1  
Valor de i: 0
```

## Resumo da aula

- Estrutura repetitiva "para"
- Usar quando se sabe previamente a quantidade de repetições
- Ótimo para fazer contagens (progressiva ou regressiva)
- Regra:



## Exercício de testes de mesa com for

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

```

x = 4;
y = x + 2;
for (i=0; i<x; i++) {
    System.out.print(x+" "+y);
    y = y + i;
}

```

<b>x</b>	<b>y</b>	<b>i</b>

Tela:

```

for (i=1; i<5; i++){
    y = i - 1;
    x = i * 10;
    System.out.print(i);
}

```

<b>x</b>	<b>y</b>	<b>i</b>

Tela:

```

y = 10;
for (i=0; i<4; i++){
    System.out.print(i);
    y = y + i;
    System.out.println(y);
}

```

<b>x</b>	<b>y</b>	<b>i</b>

Tela:

```

x = 4;
y = 0;
for (i=0; i<x; i++) {
    System.out.print(i);
    System.out.println(x);
    y = y + 10;
}

```

<b>x</b>	<b>y</b>	<b>i</b>

Tela:

```

x = 4;
y = 0;
for (i=0; i<x; i++) {
    y = y + i;
}
System.out.println(y);

```

<b>x</b>	<b>y</b>	<b>i</b>

Tela:

```

x = 8;
y = 3;
for (i=0; y<x; i++){
    x = x - 2;
    y = y + 1;
    System.out.println(i);
}

```

<b>x</b>	<b>y</b>	<b>i</b>

Tela:

```

x = 4;
y = x + 2;
for (i=0; i<x; i++) {
    System.out.print(x+" "+y);
    y = y + i;
}

```

4	<del>0 8 7</del> 8 12	<del>0 1 2</del> 3 4
---	--------------------------	-------------------------

**x**      **y**      **i**

Tela:

4 6 4 6 4 7 4 9

```

for (i=1; i<5; i++){
    y = i - 1;
    x = i * 10;
    System.out.print(i);
}

```

<del>10 20 30</del> 40	<del>0 1 2</del> 3	<del>1 2 3</del> 4 5
---------------------------	-----------------------	-------------------------

**x**      **y**      **i**

Tela:

1 2 3 4

```

y = 10;
for (i=0; i<4; i++){
    System.out.print(i);
    y = y + i;
    System.out.println(y);
}

```

	<del>10 10 11</del> 13 16	<del>0 1 2</del> 3 4
--	------------------------------	-------------------------

**x**      **y**      **i**

Tela:

0 10  
1 11  
2 13  
3 16

```

x = 4;
y = 0;
for (i=0; i<x; i++) {
    System.out.print(i);
    System.out.println(x);
    y = y + 10;
}

```

4	<del>0 10 20</del> 30 40	<del>0 1 2</del> 3 4
---	-----------------------------	-------------------------

**x**      **y**      **i**

Tela:

0 4  
1 4  
2 4  
3 4

```

x = 4;
y = 0;
for (i=0; i<x; i++) {
    y = y + i;
}
System.out.println(y);

```

4	<del>0 0 1</del> 3 6	<del>0 1 2</del> 3 4
---	-------------------------	-------------------------

**x**      **y**      **i**

Tela:

6

```

x = 8;
y = 3;
for (i=0; y<x; i++){
    x = x - 2;
    y = y + 1;
    System.out.println(i);
}

```

<del>8 8 4</del>	<del>3 4 5</del>	<del>0 1 2</del>
------------------	------------------	------------------

**x**      **y**      **i**

Tela:

0  
1

# Estrutura repetitiva "faça-enquanto"

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

## Estrutura repetitiva "faça-enquanto"

Menos utilizada, mas em alguns casos se encaixa melhor ao problema.

O bloco de comandos executa pelo menos uma vez, pois a condição é verificada no final.

## Sintaxe / regra

```
do {  
    comando 1  
    comando 2  
} while ( condição );
```

**Regra:**

**V: volta**

**F: pula fora**

### Problema exemplo:

Fazer um programa para ler uma temperatura em Celsius e mostrar o equivalente em Fahrenheit. Perguntar se o usuário deseja repetir (s/n). Caso o usuário digite "s", repetir o programa.

Fórmula:  $F = \frac{9C}{5} + 32$

Exemplo:

```
Digite a temperatura em Celsius: 30.0  
Equivalente em Fahrenheit: 86.0  
Deseja repetir (s/n)? s  
Digite a temperatura em Celsius: 21.0  
Equivalente em Fahrenheit: 69.8  
Deseja repetir (s/n)? s  
Digite a temperatura em Celsius: -10.5  
Equivalente em Fahrenheit: 13.1  
Deseja repetir (s/n)? n
```

## Resumo da aula

- Estrutura repetitiva "faça-enquanto"
- O bloco de comandos executa pelo menos uma vez, pois a condição é verificada no final.
- Regra:
  - V: volta
  - F: pula fora

```
do {  
    comando 1  
    comando 2  
} while ( condição );
```

```
import java.util.Locale;  
import java.util.Scanner;  
  
public class Main {  
  
    public static void main(String[] args) {  
  
        Locale.setDefault(Locale.US);  
        Scanner sc = new Scanner(System.in);  
  
        char resp;  
        do {  
            System.out.print("Digite a temperatura em Celsius: ");  
            double C = sc.nextDouble();  
            double F = 9.0 * C / 5.0 + 32.0;  
            System.out.printf("Equivalente em Fahrenheit: %.1f%n", F);  
            System.out.print("Deseja repetir (s/n)? ");  
            resp = sc.next().charAt(0);  
        } while (resp != 'n');  
  
        sc.close();  
    }  
}
```