

Prof. Dr. Fernando Almeida proffernando.almeida@fiap.com.br

DDD (Domain Driven Design) Estruturas de Repetição

O QUE VAMOS APRENDER HOJE?

Estruturas de repetição

- 1 Estrutura for
- 2 Estrutura while
- 3 Estrutura do/while

4 Instrução break

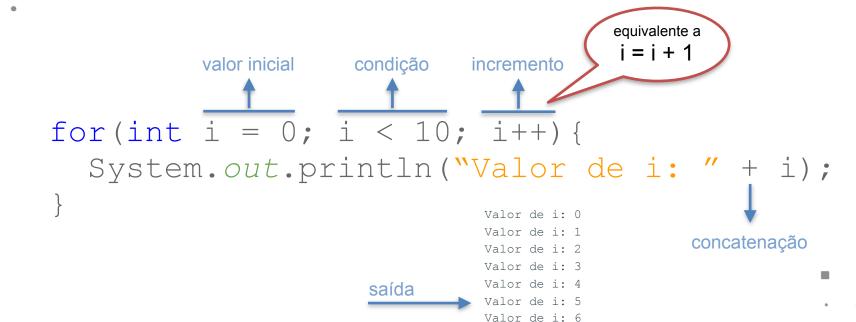
5 Instrução continue

Estruturas de Repetição Instrução for

 Normalmente utilizado quando conhecemos o número de repetições de uma instrução ou bloco de instruções

Estruturas de Repetição em JAVA Exemplo for

i += 10; i = i + 10;



Valor de i: 7 Valor de i: 8 Valor de i: 9

. .

Exemplo prático

Vamos criar um trecho de código para imprimir o valor da variável "i" dentro da estrutura de repetição.

Instrução for

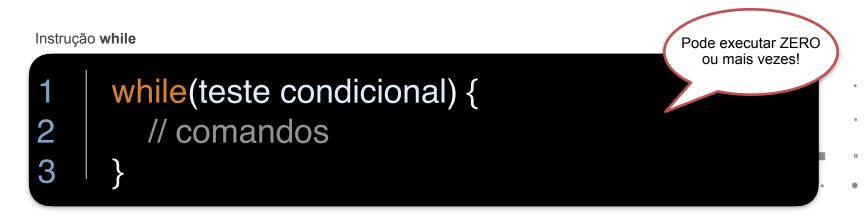
```
1 package Aula8_Repeticao;
3 public class ExemploPraticoRepeticao3 {
      public static void main(String[] args) {
          for(int i=1; i<10; i++) {
              System.out.println("0 valor é i: " + i);
10 }
```

Exercícios - for

- 1) Crie um programa que imprima uma contagem regressiva entre 100 e 0 (inclusive), decrementando em 5.
- 2) Crie um programa que leia um valor X e em seguida, mostre os valores ímpares entre 1 e X.
- 3) Escreva um programa para validar uma senha de um usuário. O programa deve assumir uma senha padrão e realizar o teste. Caso a senha seja válida, o programa deve imprimir a mensagem: "Senha válida", caso contrário, o programa deve imprimir a mensagem: "Senha inválida" e solicitar novamente a entrada de uma nova senha. O usuário possui três chances.

Estruturas de Repetição Instrução while

- Normalmente utilizado quando não sabemos quantas vezes um determinado bloco de instruções precisa ser repetido
- Condição pré-testada
- As instruções se repetem enquanto uma condição for verdadeira



Exemplo while

```
controlador do while
int i = 0;
while (i < 10) {
   System.out.println("Algoritmos");
   i++;//i = i + 1;
                                                  ALGORITMOS
                                                  ALGORITMOS
                                                  ALGORITMOS
                                                  ALGORITMOS
                                        saída
   incremento explícito
                                                  ALGORITMOS
                                                  ALGORITMOS
```

ALGORITMOS
ALGORITMOS
ALGORITMOS

Exemplo prático

Vamos criar um **trecho** de código para calcular o valor de um salário. Considerando o salário inicial de R\$ 1000,00, o salário será reajustado em 10% enquanto o salário for menor do que R\$ 5000,00.

Instrução while

Exemplo prático

Vamos criar um **trecho** de código para calcular o valor de um salário. Considerando o salário inicial de R\$ 1000,00, o programa será reajustado em 10% enquanto o salário for menor do que R\$ 5000,00.

```
Adicione
Instrução while
                                                                   um comando para
  1 package Aula8_Repeticao;
                                                                   contar o número de
                                                                      repetições
    public class ExemploPraticoRepeticao1 {
        public static void main(String[] args) {
  4∙
             double salario = 1000;
  5
  6
             while(salario < 5000) {</pre>
  8
                 salario = salario + (salario * 0.1);
                 System.out.println("Salário R$ " + salario);
10
```

Exercícios - while

- 1)Crie um programa que mostre os números pares entre 1 e 100 (incluindo ambos).
- 2) Crie um programa para ler 6 valores do teclado e ao final, imprimir as seguintes saídas:
 - a) Soma de todos os valores positivos digitados;
 - b) Quantidade de valores positivos digitados;
 - c) Média de todos os valores positivos digitados

Estrutura de Repetição Instrução do/while

- Muito parecido com a instrução while
- Condição pós-testada
- Os comandos são executados ao menos uma vez



Exemplo do/while

```
int i = 0, j = 10;
do
    System.out.println(i + " é menor que " + j);
     <u>i++;</u>
                                                        0 é menor que 10
                                                        1 é menor que 10
while (i < j);
                                                        2 é menor que 10
                                                        3 é menor que 10
                                                        4 é menor que 10
                                              saída
                                                        5 é menor que 10
                                                        6 é menor que 10
          condição
```

7 é menor que 10 8 é menor que 10 9 é menor que 10

Exemplo prático

Instrução do/while

```
1 package Aula8 Repeticao;
 3 import java.util.Scanner;
  public class ExemploPraticoRepeticao4 {
       public static void main(String[] args) {
 6•
           Scanner input = new Scanner(System.in);
           int num;
           do {
               System.out.println("Digite um número entre 1 e 5: ");
               num = input.nextInt();
               System.out.println("Número digitado: " + num);
           }while(num < 1 || num > 5);
13
```

Exemplo prático

Instrução do/while

```
1 package Aula8_Repeticao;
                                                                   Acrescente
                                                                um comando para
                                                                contar o número de
 3 import java.util.Scanner;
                                                                   repetições
   public class ExemploPraticoRepeticao4 {
       public static void main(String[] args) {
 6•
           Scanner input = new Scanner(System.in);
           int num;
           do {
                System.out.println("Digite um número entre 1 e 5: ");
                num = input.nextInt();
                System.out.println("Número digitado: " + num);
           }while(num < 1 || num > 5);
13
```

Exercícios - do/while

1) Crie um programa para ler um valor inteiro N do teclado, onde o N é maior ou igual a 2 e menor ou igual a 1000. Após realizar a validação de N, crie a tabuada de N no seguinte formato:

 $1 \times N = N$ $2 \times N = 2N$... $10 \times N = 10N$

Considerando uma entrada 140 (N = 140), o programa deverá imprimir:

```
1 x 140 = 140

2 x 140 = 280

3 x 140 = 420

4 x 140 = 560

5 x 140 = 700

6 x 140 = 840

7 x 140 = 980

8 x 140 = 1120

9 x 140 = 1260

10 x 140 = 1400
```

Estrutura de Repetição Instrução break

Comando break

Quando usado em **estruturas de repetição** (while, do/while e for) e com o **comando de seleção** switch/case, causa a interrupção imediata do mesmo.

Quebrar, parar, frear, interromper...



Estruturas de Repetição - break

```
int i = 0, soma = 0, x;
Scanner sc = new Scanner(System.in);
while (i < 10) {
  System.out.println("Digite o item " + i);
  x = sc.nextInt();
  if(x < 0)
  System.err.println("Número inválido!");
  break; imprime uma mensagem (de erro) em vermelho
     a instrução break irá interromper imediatamente a estrutura while
  soma = soma + x;
  i = i + 1;
System.out.println("TOTAL: " + soma);
```

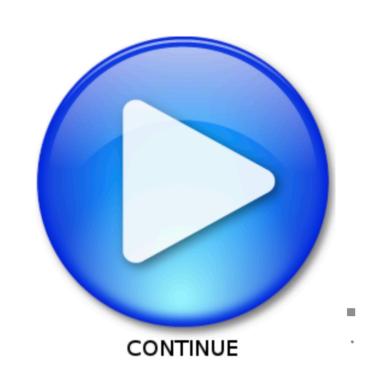
Exemplo prático - comando break

```
10 public class BreakTest
       public static void main(String[] args) {
11⊜
            long i = System.currentTimeMillis();
           boolean imprimir = true;
           for(int count = 1; count <1000000; count++) {</pre>
                if((count % 17 == 0) && (count % 19 == 0)) {
                    if(imprimir) {
                        System.out.println(count);
                        imprimir = false;
23
           System.out.println("Tempo de execução, em milisegundos: " + (System.currentTimeMillis() - i));
           System.out.println("\n\ncom o uso do BREAK!!! \n");
           i = System.currentTimeMillis();
           for(int count = 1; count <1000000; count++) {</pre>
                if((count % 17 == 0) && (count % 19 == 0)) {
                    System.out.println(count);
                    break:
35
36
           System.out.println("Tempo de execução, em milisegundos: " + (System.currentTimeMillis() - i));
```

Comando continue

"Continua" a **estrutura de repetição**. Enquanto o comando **break** interrompe o laço, o **continue** interrompe apenas a
iteração **atual**.

Continuar, seguir em frente...



Exemplo prático - comando continue

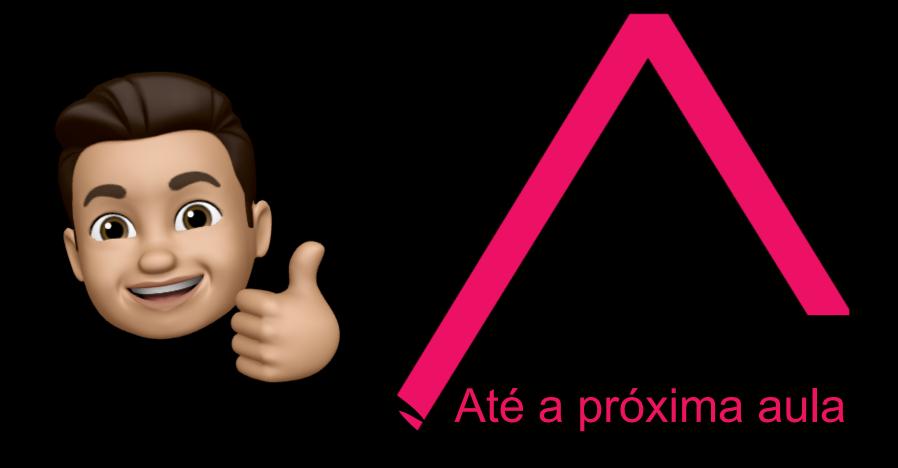
```
package Aula8_Repeticao;
   public class continueTest {
       public static void main(String[] args) {
 40
            long i = System.currentTimeMillis();
           for(int count=1; count <= 1000000; count++) {</pre>
               if(count % 2 == 0) {
                    continue;
11
12
               if((count % 17 == 0) && (count % 19 == 0)) {
                    System.out.println(count);
13
                    break:
14
           System.out.println("Tempo de execução, em milisegundos: " + (System.currentTimeMillis() - i));
17
18 }
```

OBRIGADO



Copyright © 2020 | Prof. Dr. Fernando Luiz de Almeida

Todos os direitos reservados. Reprodução ou divulgação total ou parcial deste documento, é expressamente proibido sem consentimento formal, por escrito, do professor/autor.



Copyright © 2020 | Prof. Dr. Fernando Luiz de Almeida

Todos os direitos reservados. Reprodução ou divulgação total ou parcial deste documento, é expressamente proibido sem consentimento formal, por escrito, do professor/autor.