

For

Prof. Luís Guilherme de S. Lopes



Laços de Repetição: O for Loop

Conceito: Permite executar um bloco de código repetidamente de forma organizada e eficiente.

Vantagens: Automatização de tarefas, redução de código redundante e clareza na implementação.

Estrutura do Laço for em Java



```
for (inicialização; condição; incremento) {  
    // bloco de código  
}
```

Inicialização: Declara e inicializa a variável de controle.

Condição: Expressão booleana que, enquanto verdadeira, mantém o loop.

Incremento: Atualização da variável a cada iteração.

Estruturas Condicionais

Exemplo 1: Contador de 1 a 10

```
public class Contador1a10 {  
    public static void main(String[] args) {  
        for (int C = 1; C <= 10; C++) {  
            System.out.println(C);  
        }  
    }  
}
```

Explicação: Imprime os números de 1 até 10.

Exemplo 2: Contador Decrescente com 3

```
public class ContadorDecrescente {  
    public static void main(String[] args) {  
        for (int C = 10; C >= 1; C -= 3) {  
            System.out.println(C);  
        }  
    }  
}
```

Explicação: Inicia em 10 e decrementa 3 a cada iteração, até atingir ou ultrapassar 1.



Exemplo 3: Soma de Valores com Entrada do Usuário

Lê 5 valores digitados pelo usuário e exibe a soma total.

```
import java.util.Scanner;

public class Somador {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int soma = 0;
        for (int C = 1; C <= 5; C++) {
            System.out.print("Digite um valor: ");
            int N = sc.nextInt();
            soma += N;
        }
        System.out.println("A soma dos valores é " + soma);
        sc.close();
    }
}
```



Estruturas Condicionais

Quando não podemos utilizar a estrutura FOR (Para)

A estrutura de repetição FOR não é utilizada quando não sabemos quantas vezes queremos repetir. Nesse caso devemos utilizar a estrutura Enquanto.

Mesmo que existam alguns métodos para se fazer esta ação não é aconselhado que se utilize a estrutura Para nestes casos.

Exemplo 4 - Contagem Simples

```
public class SomaNumeros {  
    public static void main(String[] args) {  
        int soma = 0;  
        for (int i = 1; i <= 5; i++) {  
            soma += i;  
        }  
        System.out.println("A soma dos números é: " +  
soma);  
    }  
}
```




O laço irá executar 5 vezes.
A cada iteração, o valor de i será exibido na tela.

Exemplo 5 - Soma de Números

```
import java.util.Scanner;

public class Somador {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int soma = 0;
        for (int C = 1; C <= 5; C++) {
            System.out.print("Digite um valor: ");
            int N = sc.nextInt();
            soma += N;
        }
        System.out.println("A soma dos valores é " + soma);
        sc.close();
    }
}
```

Lê 5 valores digitados pelo usuário e exibe a soma total.



Exemplo 6 - Tabuada

```
import java.util.Scanner;

public class Tabuada {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Digite um número: ");
        int numero = sc.nextInt();
        for (int i = 1; i <= 10; i++) {
            System.out.println(numero + " x " + i + " = " +
(numero * i));
        }
        sc.close();
    }
}
```

O programa solicita ao usuário um número.
O laço executa 10 vezes, calculando e exibindo a tabuada do número informado.



Estruturas Condicionais

Tente fazer um algoritmo para mostrar quantos valores digitados estão entre 0 e 10, após 6 digitações.

Estruturas Condicionais

Tente fazer um algoritmo para mostrar quantos valores digitados estão entre 0 e 10, após 6 digitações.

```
import java.util.Scanner;

public class QuantosEntre0e10 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int count = 0;
        for (int C = 1; C <= 6; C++) {
            System.out.print("Digite um valor: ");
            int V = sc.nextInt();
            if (V >= 0 && V <= 10) {
                count++;
            }
        }
        System.out.println("Ao todo foram " + count + " valores entre 0 e 10");
        sc.close();
    }
}
```



Estruturas Condicionais

Tente fazer um algoritmo para mostrar quantos valores digitados estão entre 0 e 10, após 6 digitações. E também mostre a soma de todos ímpares entre 0 e 10 digitados.

Estruturas Condicionais

Tente fazer um algoritmo para mostrar quantos valores digitados estão entre 0 e 10, após 6 digitações. E também mostre a soma de todos ímpares entre 0 e 10 digitados.

```
import java.util.Scanner;

public class QuantosEntre0e10ComImpares {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int count = 0;
        int somalmpares = 0;
        for (int C = 1; C <= 6; C++) {
            System.out.print("Digite um valor: ");
            int V = sc.nextInt();
            if (V >= 0 && V <= 10) {
                count++;
                if (V % 2 == 1) {
                    somalmpares += V;
                }
            }
        }
        System.out.println("Ao todo foram " + count + " valores entre 0 e 10");
        System.out.println("Nesse intervalo, a soma dos ímpares foi " + somalmpares);
        sc.close();
    }
}
```



Conclusão

Pontos importantes sobre o laço para:

- É uma estrutura poderosa para repetir tarefas.
- Torna o código mais eficiente e organizado.
- Permite automatizar tarefas repetitivas.