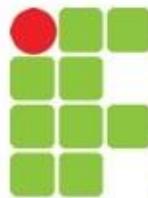


Java: Procedimentos, Funções e Métodos

Professor:

Vilson Heck Junior
vilson.junior@ifsc.edu.br

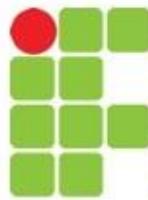




Agenda

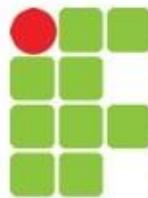
- Introdução;
- Definições e Exemplos:
 - Procedimento;
 - Função;
 - Método;





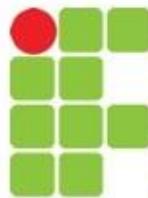
Introdução

- Estudamos Algoritmos na disciplina de Introdução à Programação, basicamente no Visualg;
- Agora, é importante utilizar estes recursos em Java, pois a disciplina de Estrutura de Dados irá necessitar, e muito, do uso destes recursos.



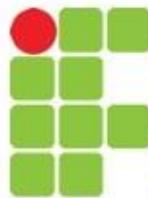
Introdução

- Para complementar o conceito de algoritmos, iremos estudar três elementos básicos da programação, que visam facilitar e tornar nossos algoritmos modular:
 - Procedimentos;
 - Funções;
 - Métodos.



Definição

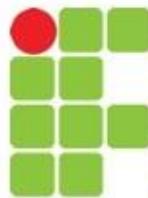
- **Procedimento:** é algo que deve ser feito uma ou várias vezes, sempre que for necessário.
 - Pode, ou não, receber parâmetros.
 - Não retorna um resultado.
 - Exemplos:
 - Gravar dados em arquivo no disco;
 - Enviar documento para a impressora;
 - Ordenar que o computador desligue.



Definição

- **Função:** é algo que deve ser feito uma ou várias vezes, sempre que for necessário para se obter um resultado.
 - Pode, ou não, receber parâmetros;
 - Sempre retorna um resultado;
 - Exemplos:
 - Mostrar um menu e retornar a opção;
 - Solicitar uma entrada ao usuário;
 - Efetuar uma operação matemática;
 - Ler dados de um arquivo;

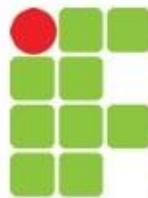
```
1  package exfuncao;
2  ⊖ import java.util.Scanner;
3  public class ExFuncao {
4      ⊖ public static void main(String[] args) {
5          int opcao;
6          do {
7              opcao = MostraMenu();
8          } while (opcao != 2);
9      }
10     ⊖ public static int MostraMenu() {
11         Scanner entrada = new Scanner(System.in);
12         System.out.println("==== MENU ====");
13         System.out.println("1 - Mostrar de novo");
14         System.out.println("2 - Sair");
15         return Integer.parseInt(entrada.nextLine());
16     }
17 }
```



Parâmetros/Argumentos

- Podem ser utilizados tanto em procedimentos quanto em funções;
- Servem para informar valores de dados que serão necessários para executar a tarefa solicitada;
 - `System.out.println("Olá mundo!");`

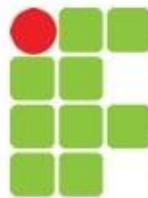
Chamada do método	Parâmetro	Fechamento da chamada
<code>System.out.println(</code>	<code>"Olá mundo!"</code>	<code>);</code>



Parâmetros/Argumentos

- Ex.: Calcular descontos percentual em valores de produtos.
 - Precisa de quais dados?
 - Precisa do valor atual do produto – **double**;
 - Precisa do percentual de desconto – **double**;
 - Resulta em algum novo dado?
 - Sim, é uma função! O valor com desconto – **double**;

```
public static double calcDesc(double va, double pc) {  
    double vd = va * (pc / 100);  
    return va - vd;  
}
```



Procedimentos/Funções

- Outras orientações:
 - São subalgoritmos, ou subprogramas;
 - Por questões de reaproveitamento de código, evite efetuar entrada/saída de dados dentro dos procedimentos/funções:
 - Passe os dados necessários através de parâmetros;
 - Receba resultados pelo retorno;
 - Lembre que ao terminar a execução de um procedimento ou função, o seu programa irá continuar a partir do ponto onde havia efetuado a chamada.



Definição

- **Método:** é o mesmo que um procedimento ou uma função, o detalhe é que um método é o nome dado para um destes, quando estamos programando classes, em orientação a objetos.
 - Exemplos:
 - Disciplina de Orientação à Objetos!

```
1  package exfuncao;
2  [-] import java.util.Scanner;
3  public class ExFuncao {
4      [-]     public static void main(String[] args) {
5          int opcao;
6          do {
7              opcao = MostraMenu("==== MENU ====");
8          } while (opcao != 2);
9      }
10     [-]     public static int MostraMenu(String titulo) {
11         Scanner entrada = new Scanner(System.in);
12         System.out.println(titulo);
13         System.out.println("1 - Mostrar de novo");
14         System.out.println("2 - Sair");
15         return Integer.parseInt(entrada.nextLine());
16     }
17 }
```