



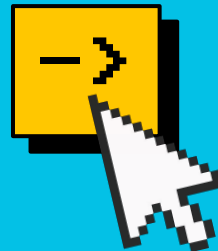
**COMPETEC**





Aula nº07

# Métodos Funções



## Assuntos da aula

- >O que são funções
- >Estrutura de métodos e funções
- >Como utilizar

Ok

Cancel

Options<<

Funções e/ou métodos são um conjunto de códigos que desempenham uma determinada tarefa.

Os métodos têm uma estrutura padrão que irá determinar o comportamento deles.

Basicamente, toda função é um método, mas nem todo método é uma função. Isso porque uma função sempre deve retornar algum valor (assim como na matemática), enquanto o método pode ou não retornar valores.

Use métodos para reduzir a repetição de códigos. Todas as ações que deverão ser feitas em mais de uma parte do código podem se tornar um método.

Métodos também são muito úteis para facilitar mudanças e refatoramentos em seus códigos, se alguma ação específica precisa ser alterada basta modificá-la uma vez.

A estrutura do cabeçalho do método é composta por algumas informações sobre ele. É importante se atentar a ordem em que essas informações devem ser colocadas.

	Informações		
Modificador de acesso	public	private	
Complemento	static		
Tipo de retorno	void	int	Objeto
Nome	main	media	novoUsuario
Lista de parâmetros	(String[ ] args)	(float n1, float n2)	()

Para declarar, ou seja, criar um novo método, é importante sempre lembrar de colocar o **cabeçalho** do mesmo, seguido de chaves. Entre as chaves está o que chamamos de *corpo* do método, coloque nele todo o código necessário com os procedimentos do método.

```
public static void imprimeIMC(float altura, float peso){  
    float imc = peso / (altura * altura);  
    System.out.println("Seu imc eh: " + imc);  
}
```

O que esse código faz? Como você o descreveria? Qual nome você daria?

```
public static boolean _____(float n1, float n2){  
    float m = (n1 + n2) / 2;  
    return (m >= 5);  
}
```



Basicamente, se o método for void, ele pode ser chamado colocando em uma linha o seu nome e os atributos necessários da função.

```
imprimeIMC(altura, peso);
```

Se o método não for void, seu valor de retorno pode ser atribuído a uma variável.

```
int idade = getIdade(id);
```

Seu valor também pode ser utilizado diretamente em uma operação ou comparação.

```
if (aprovado(nota1, nota2)) {  
    System.out.println("Aluno aprovado!");  
}
```

# COMPETEC



Obrigado pela  
atenção! :)

Don't show this message again

Ok

