**Criar Métodos**

**Objetivo:**

1. Entender o que é.
2. Sabe como e quando usar.
3. Aplicar boas práticas em sua criação e uso.

**Conceito:** É uma porção de código que é disponibilizada por uma classe. Este é executado quando é feita uma requisição a ele. São responsáveis por definir e realizar um determinado comportamento.

**Padrão de definição:**

* Visibilidade: public, protected ou private;
* Tipo: concreto ou abstrato;
* Modificador: static ou final;
* Retorno: tipo de dado ou void;
* Nome: nome fornecido ao método;
* Parâmetro: parâmetro que pode receber;
* Exceções: exceções que pode lançar;
* Corpo: código que possui ou vazio;

**Particularidades**

* **Assinatura:** é a forma de identificar unicamente o método;  
  Ass = nome + parâmetros

**Método:**

Public double calcularTotalVenda (double precoItem1, double precoitem2, double precoItem3) {...};

**Assinatura:**

calcularTotalVenda (double precoItem1, double precoitem2, double precoItem3);

* **Construtor e Destrutor:** são métodos usados na Orientação a Objetos.
* **Mensagem:** é o ato de solicitar o método que o mesmo execute.
* **Passagem de parâmetros:**
  + Por valor (cópia)

EX:  
int i =10;

Public void fazerAlgo (int i) {  
i = i + 10;  
System.out.print(“valor de i dentro: “+ i);

(Aqui ele mostraria o valor de 20 pois seria a soma de i10 + 10)  
}

System.out.print(“valor de i fora: “+ i);

(Aqui ele mostraria apenas 10 que é o valor atribuído a variável i)

* + Por referência (endereço)

**Boas Práticas**

* Nomes devem ser descritivos, mas curtos;
* Camel-case;
* Deve possuir entre 80 e 120 linhas;
* Evite lista de parâmetros longas;
* Visibilidades adequadas;

**Sobrecarga Métodos**

**Objetivo:**

1. Entender o que é.
2. Saber como criar.

**Conceito:** É a capacidade de definir métodos para diferentes contextos, mas preservando seu nome.

Alterar a assinatura do método:

ASS = nome + parâmetros;