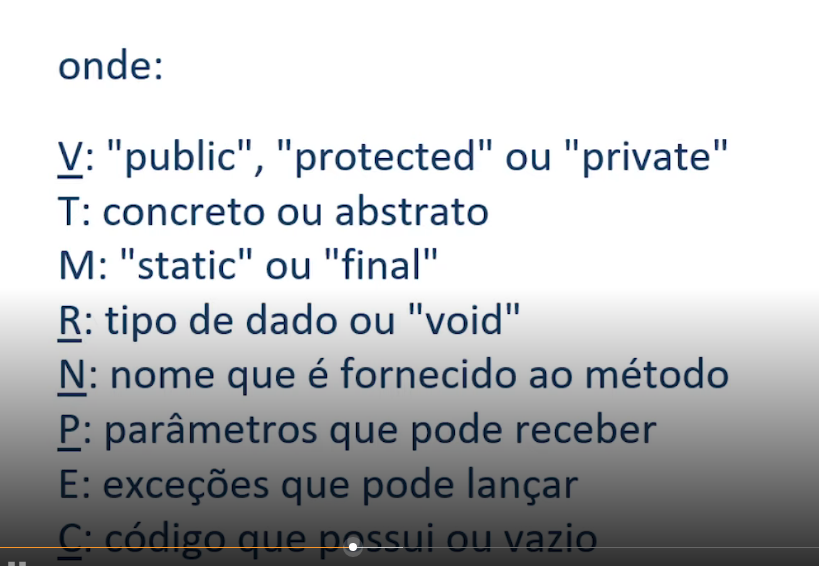
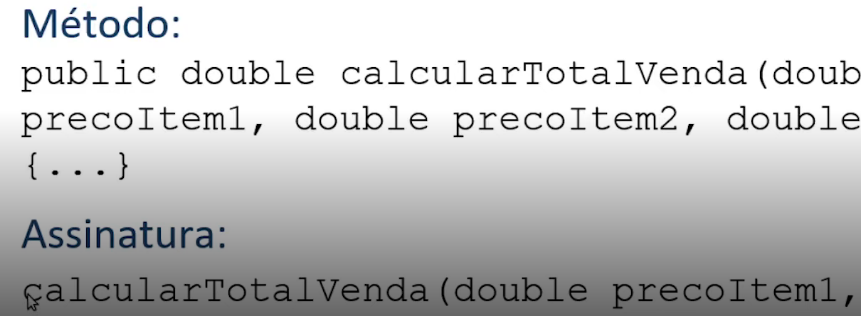
**Java**

**Método** – Uma porção de código que é responsável por definir e realizar um determinado comportamento.



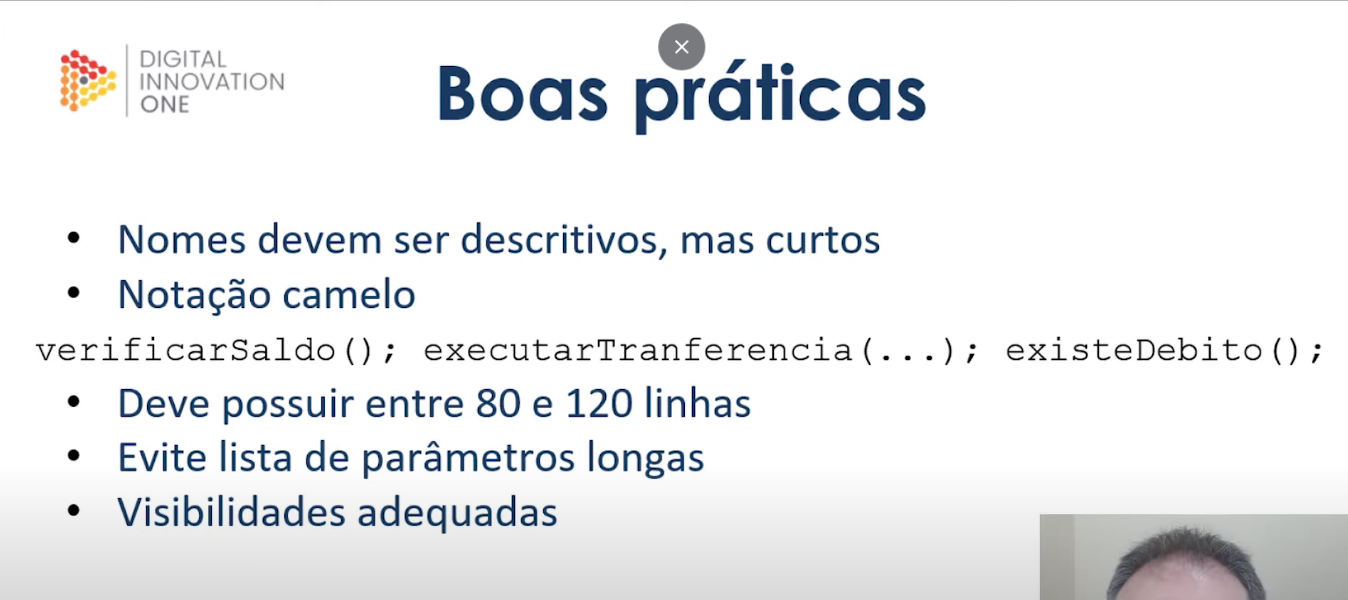
Assinatura – Forma de identificar unicamente o método – nome + parâmetros



Construtor – Cria objetos

Destrutor – Destrui objetos

Passagem de parâmetros: por valor (cópia) ou por referência (endereço)

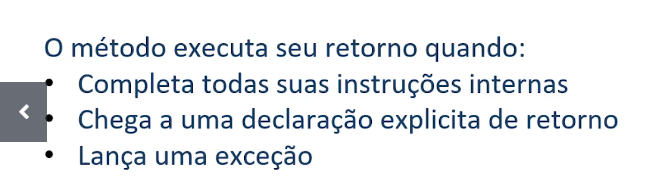


**Sobrecarga do Método**

Capacidade de definir métodos para diferentes contextos, mas preservando seu nome

**Sobrescrita do Método**

Herança



Unchecked – Roda e dps quebra

Checked – Só roda se resolver a exception

Voce pode usar THROWS ou Try....Catch

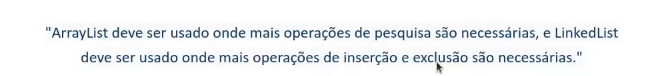
IOException extende de RunTimeException

**Collection**

**List –** Elementos duplicados e garantem ordem de inserção

Thread – Vector

Na dúvida use sempre ArrayList



indexOf(1), 2 – Troca o elemento 1 por 2 na lista

contains(2) – Vê se tem o 2 na list

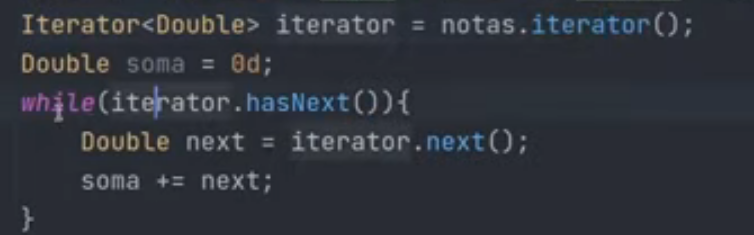
Add – Adiciona

Get(2) – retorna o elemento na posição 3

Collections.min(notas) – Retorna a menor nota da lista de notas

Collections.max(notas) – Retorna a maior nota da lista de notas

Para fazer a soma dos elemento da lista



**SET**

Não permite elementos duplicados

Não possui índice

**HashSet** – Não mantem a ordem – Perfomace melhor – Maximo 1 null

**LinkedHashSet** – Mantem a ordem – máximo 1 null

**TreeSet** – Mantem a ordem natural dos elementos (Ordenado) – não pode null

**Map – Coleção de Pares**

Elementos Unicos (key) para cada valor (value)

**HashMap –** Itera aleatório, permite null

**LinkedHashMap – Ordem de inserção, permite null**

**TreeMap –** Ordem natural, sem null