**Simples e multidimensionais**

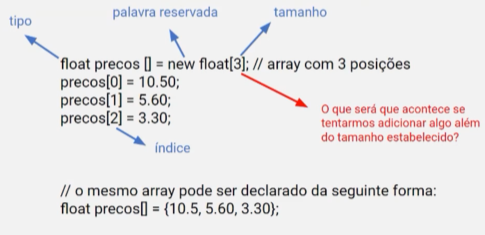
(Diego Martins de Pinho)

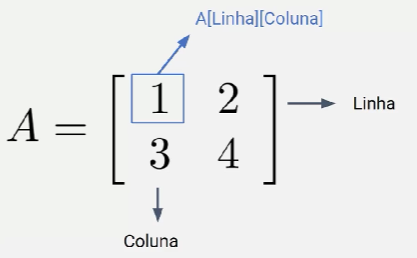
**- Tipos agregados homogêneos simples**

\* Em boa parte dos programas, existem situações onde será **necessário agrupar uma sequencia de dados** de um mesmo tipo.

\* Se cada valor for tratado com uma respectiva variável, existe o risco da pessoa programadora ter de lidar com uma quantidade muito grande de variáveis no código no caso da manipulação de muitos dados simultaneamente. Para evitar este tipo de situação e, principalmente, organizar a forma como os agrupamentos podem ser feitos, o Java possibilita a formação do que é chamado de agregados.

Agregado homogêneo:



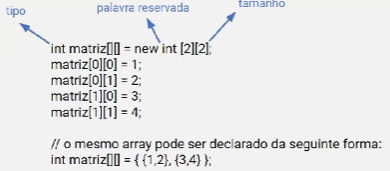


**- Tipos agregados homogêneos bidimensionais**

\* Os agregados também podem ter mais de uma dimensão.

\* O mais utilizado é o agregado bidimensional, também conhecido como matriz.

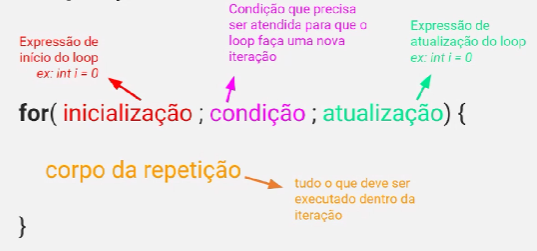
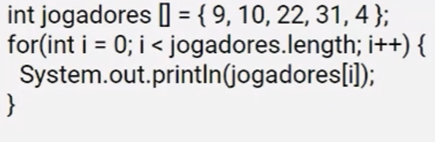
Agregado bidimensional:



**- Varredura de tipos agregados simples**

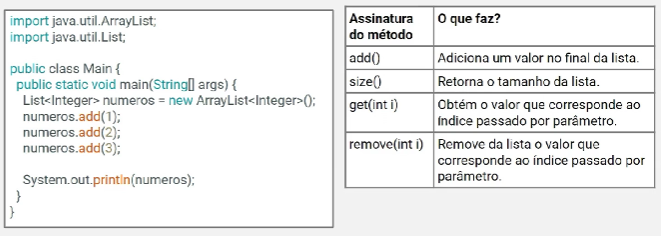
\* Varredura: a forma mais prática de varrer, ou seja, passar por todas os itens de um array, é utilizando um laço de repetição **for**.

Laço de repetição for:



**- Tipos agregados homogêneos multidimensionais**

\* Listas



Listas:

\* No Java, existem um conjunto de classes que abstraem as operações com arrays e facilitam a vida do programador. Essas estrturas são chamadas de **Lists.**

\* As três estruturas mais comuns para lidar com listas são a ArrayList, LinkedList e Vector.

\* A não ser que a questão seja otimização, a estrutura ArrayList geralmente é a mais recomendada.

\* Não aceitam tipos primitivos.

**- Varredura de tipos agregados multidimensionais**

\* Varredura:

Para arrays multidimensionais a estratégia de varredura é praticamente a mesma, a diferença é que é necessário fazer uma varredura dentro de uma varredura, ou seja, **um laço for dentro de outro laço for**.

