

Algoritmos e Programação: Fundamentos

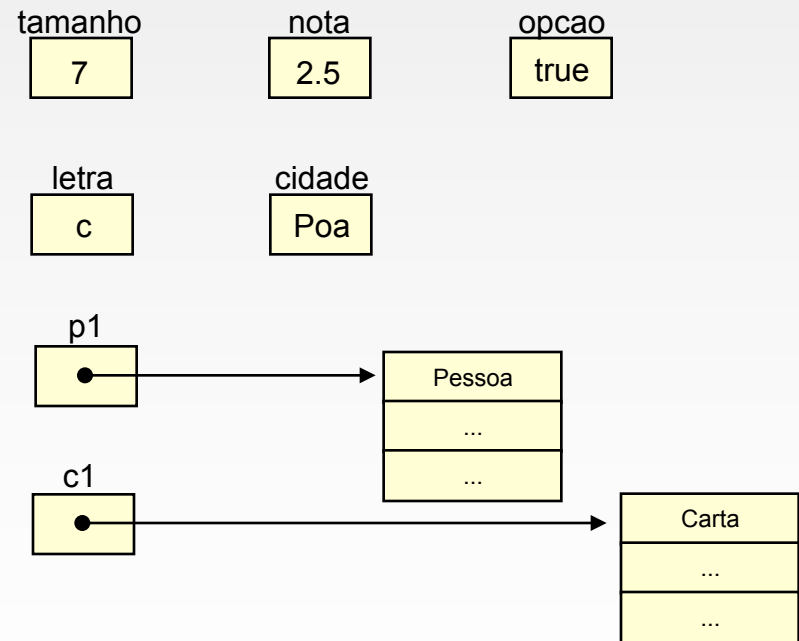
Prof. Mateus Raeder



Arrays

- Variável: armazena apenas um valor

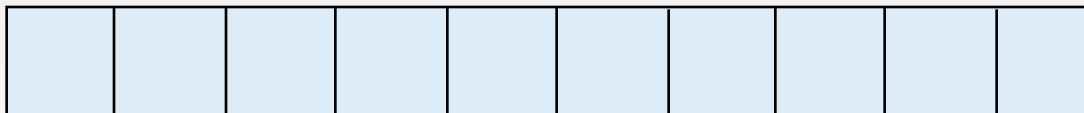
```
int tamanho;  
double nota;  
boolean opcao;  
char letra;  
String cidade;  
Pessoa p1;  
Carta c1;
```



- Como criar em uma única variável, espaço para armazenar diversos valores diferentes?

O que são arrays

- Arrays são objetos que permitem ao programador armazenar diversas variáveis do **mesmo tipo**



- Estas variáveis podem ser tanto tipos primitivos (int, char, ...) como outros objetos

Array de “int” →

0	1	2	3	4	5	6	7	8	9
---	---	---	---	---	---	---	---	---	---

Array de “char” →

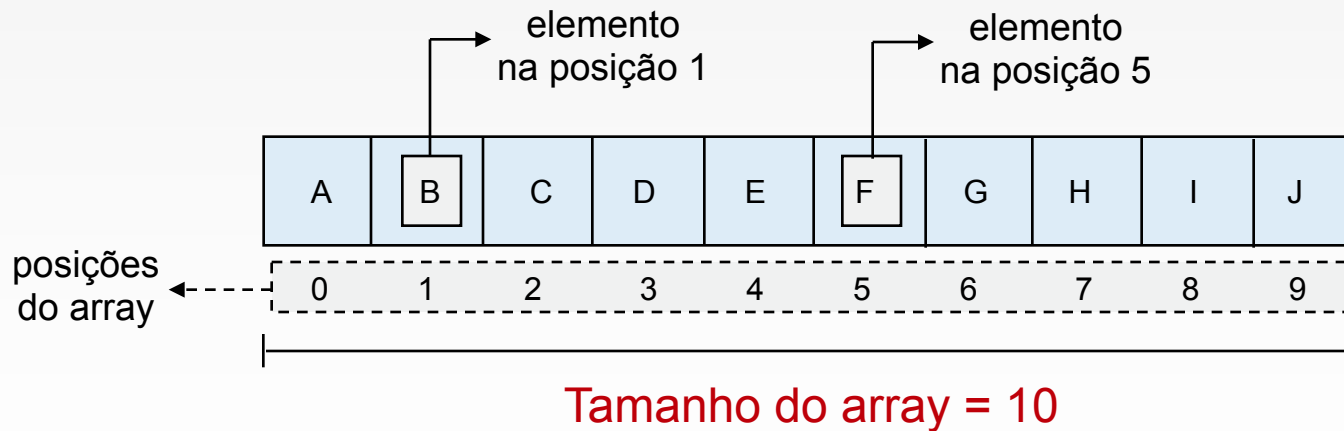
A	B	C	D	E	F	G	H	I	J
---	---	---	---	---	---	---	---	---	---

Array de “Pessoa” →

Pedro	João	Maria	Ana	Rick	André	Rê	Sofia	José	Milton
-------	------	-------	-----	------	-------	----	-------	------	--------

O que são arrays

- Cada item do array é chamado de **elemento**
- Os elementos de um array são associados a uma **posição numérica**



Declarando arrays

- Arrays armazenam variáveis de um determinado tipo, e sua declaração ocorre da mesma forma de uma variável “comum”, porém utilizando “[]” (colchetes) para a sua identificação.

Diagram illustrating the two ways to declare an array in Java:

Top declaration: `int[] valorTotal;`

- `int[]` is labeled as `tipo do array` (array type).
- `valorTotal` is labeled as `nome do array` (array name).

Bottom declaration: `int valorTotal[];`

- `valorTotal` is labeled as `nome do array` (array name).

The word `ou` (or) is placed between the two declarations, indicating they are equivalent.

- Desta forma, arrays de qualquer tipo podem ser declarados:

```
char[] charArray;  
String[] stringArray;  
double[] doubleArray;  
float[] floatArray;
```

```
boolean[] booleanArray;  
long[] longArray;  
byte[] byteArray;  
short[] shortArray;
```

Declarando arrays

Exemplos:

```
int[] tamanho; //declara um array de inteiro
```

```
double[] nota; //declara um array de double
```

```
char[] letra; //declara um array de char
```

```
Pessoa[] p1; //declara um array de Pessoa
```

```
Carta[] c1; //declara um array de Carta
```

Declarando e inicializando arrays

- Porém, declarar um array não significa criá-lo efetivamente para utilização.
- A **declaração** indica ao compilador que existe um array de determinado tipo e com determinado nome.
- Após a declaração, como fazer com que um array possa ser utilizado?
 - **criar e indicar a quantidade de elementos do array**

`valorTotal = new int[10];`

↓
cria o array

↘
número de posições

quando um array é criado, JAVA cria uma variável chamada **length**, que possui o tamanho do array, e é acessada da seguinte maneira:
nome_do_array.length

Inicializando arrays

Exemplos:

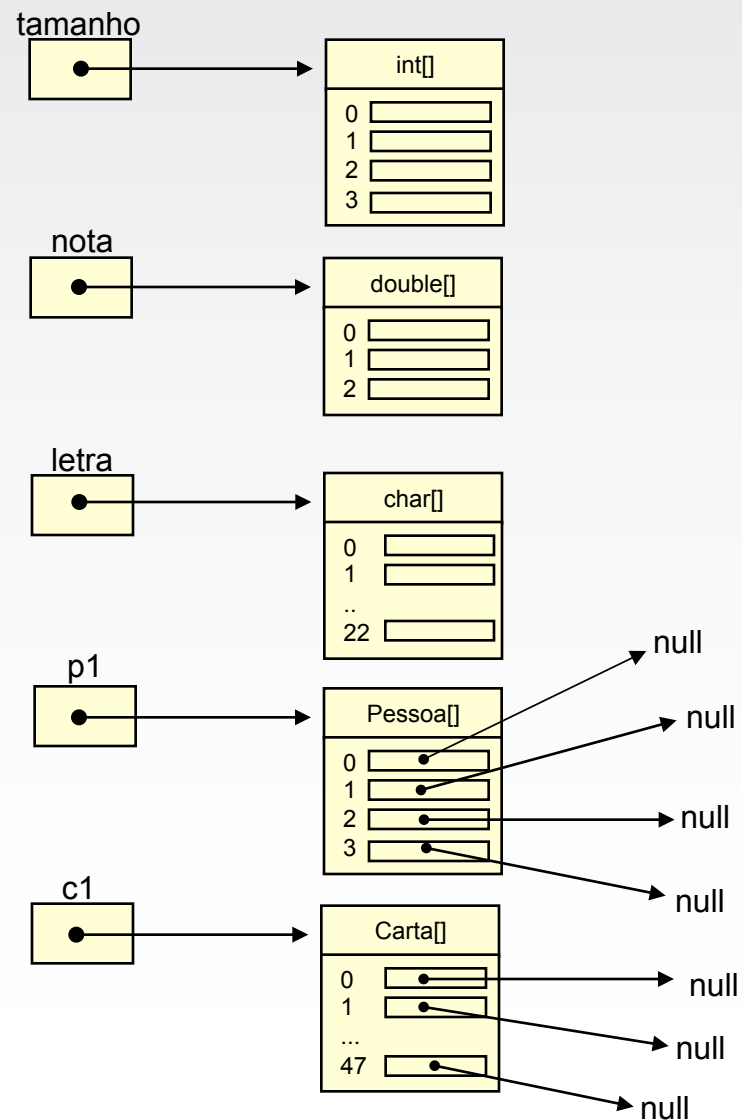
```
int[] tamanho = new int[4];
```

```
double[] nota = new double[3];
```

```
char[] letra = new char[23];
```

```
Pessoa[] p1 = new Pessoa[4];
```

```
Carta[] c1 = new Carta[48];
```



Inicializando arrays

- **IMPORTANTE:** quando um array é criado, suas posições são automaticamente inicializadas com valores pré-determinados pelo JAVA (**0** para *int*, **false** para *boolean*, **null** para objetos, etc...)
- Pode-se inicializar os valores de cada elemento do array de acordo com a necessidade/vontade, atribuindo valores para todos os seus elementos:

```
valorTotal[0] = 26;  
valorTotal[1] = 296;  
valorTotal[2] = 11;  
    ⋮  
valorTotal[9] = 1997;
```

ou

```
for(int i=0; i<valorTotal.length; i++)  
    valorTotal[i] = 0;
```

- Pode-se também declarar, criar e inicializar arrays diretamente, da seguinte forma:

```
int[] valorTotal = {26, 296, 11, 34, 58, 98, 114, 754, 236, 1997}
```

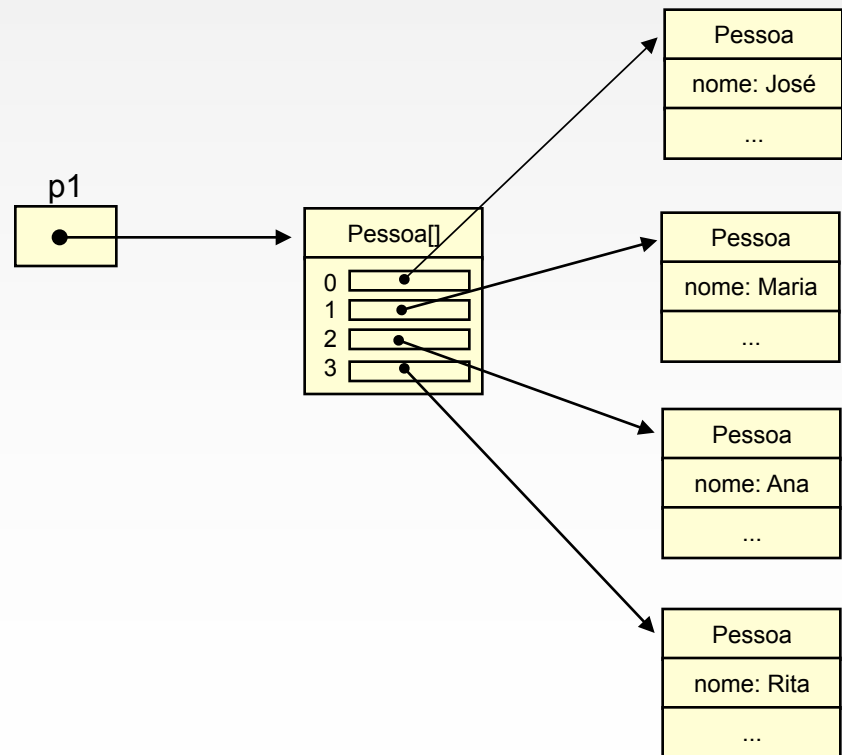
↓
quantidade de elementos = tamanho do array

Inicializando arrays de Objetos

- Ao declarar e inicializar um array de objetos, cada posição comporta-se como uma variável normal, ou seja, **deve ser inicializada**

Exemplo:

```
Pessoa[] p1 = new Pessoa[4];  
p1[0] = new Pessoa("José");  
p1[1] = new Pessoa("Maria");  
p1[2] = new Pessoa("Ana");  
p1[3] = new Pessoa("Rita");
```



Acessando elementos em arrays

- Os elementos do array são, então, acessados através de seus **índices**, que são as posições numéricas do array.

valorTotal

26	296	11	34	58	98	114	754	236	1997
0	1	2	3	4	5	6	7	8	9

```
System.out.println("Valor da primeira posição é = "+valorTotal[0]);  
System.out.println("Valor da segunda posição é = "+valorTotal[1]);  
int aux = valorTotal[7];  
int temp = valorTotal[0] + valorTotal[1] + valorTotal[2];  
for(int i=0; i<valorTotal.length; i++)  
    System.out.println("Posição "+i+" do array é "+valorTotal[i]);
```