



# Coleções

## Array e ArrayList

DC – UFRPE

Programação II

Prof. Gustavo Callou

[gustavo.callou@ufrpe.br](mailto:gustavo.callou@ufrpe.br)

# Agenda

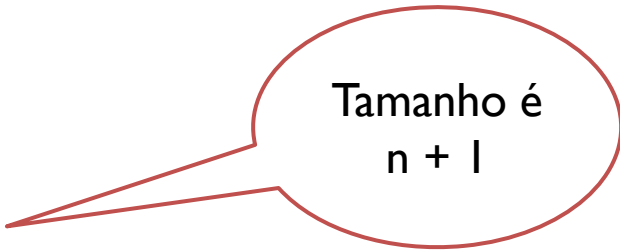


- ▶ Array
- ▶ ArrayList

# Arrays em Java

- ▶ São objetos (tipo de referência)
- ▶ Guardam tipos primitivos e de referência

0	v_1
1	v_2
...	
n+1	v_n



Tamanho é  
 $n + 1$

# Declaração e inicialização de arrays

int numeros[];

Conta contas[];

Declaração  
sem  
inicialização

int a[] = **new** int[50];

String nomes[] = **new** String[300];

Declaração e  
inicialização  
com valor  
padrão

# Atribuindo valor não padrão



- ▶ Declarando primeiro e inicializando depois

```
Mostrador mostradores[] = new Mostrador[2];  
mostradores[0] = new Mostrador(24);  
mostradores[1] = new Mostrador(60);
```

# Atribuindo valor não padrão



- ▶ Declaração e inicializando simultânea

```
float cotacoes[] = {1.73f, 1.64f, 1.81f, 1.85f};
```

```
String tratamento[] = {"Sr.", "Sra.", "Dr.", "Dra."};
```

# Tamanho do array

- ▶ length: atributo que indica o tamanho do array
- ▶ Índices: de 0 até `array.length - 1`
- ▶ Exemplo: inicializando array de quadrados

```
int quadrados[] = new int[tamanho];  
for(int i=0; i < quadrados.length; i++) {  
    quadrados[i] = i*i;  
}
```

# ArrayIndexOutOfBoundsException

- ▶ Exceção que ocorre ao acessar um índice inválido

```
char array[] = new char[10];
```

```
array[0] = 'a';
```

```
array[9] = 'j';
```

```
...
```

```
array[10] = 'k';
```





# for para array

- ▶ Existe um tipo de for que pode ser utilizado para varrer arrays e outros tipos iterativos (ex: listas) do primeiro ao último elemento
- ▶ Exemplo: imprimindo todos os elementos do array

```
float cotacoes[] = {1.73f, 1.64f, 1.81f, 1.85f};
```

```
for (float cot : cotacoes){  
    System.out.println(cot);  
}
```

```
float cot = 0;  
for (int i=0; i < cotacoes.length; i++) {  
    cot = cotacoes[i];  
    System.out.println(cot)  
}
```

# Arrays multidimensionais

```
int[][] a = { {1,0}, {0,1} };
```

2 x 2

a [0][0]

a [0][1]

a [1][0]

a [1][1]

# Arrays multidimensionais

```
int matriz[][] = new int[2][3];
```

```
for (int i = 0; i < matriz.length; i++) {  
    for (int j = 0; j < matriz[0].length; j++) {  
        matriz[i][j] = -1;  
    }  
}
```

Array de  
linhas

Array das  
colunas

# Passando argumentos na linha de comando



```
public class AppAnuncio {  
    public static void main (String[] args) {  
        System.out.println("Anuncio de :" + args[0]);  
        System.out.println("Mensagem : " + args[1]);  
    }  
}
```

```
java AppAnuncio "Denis" "Hoje haverá reunião do projeto"
```

---

# Argumentos de uma aplicação



- ▶ É um array de strings cuja referência é passada para o método principal

**public void static** main (String[] args)

- ▶ Se nenhum argumento é passado array tem tamanho 0

# Listando argumentos

```
public class ListandoParametros {  
    public static void main (String[] args) {  
        if(args.length == 0)  
            System.out.println("0 parametros");  
        else  
            for(int i = 0; i < args.length; i++) {  
                System.out.printf("Parametro %d = %s \n", i, args[i]);  
            }  
    }  
}
```