

Arrays no Java

Capítulo IV

Declarando um array

Array de primitivos

```
byte[] arrayBytes;  
short[] arrayShorts;  
long[] arrayLongs;  
float[] arrayFloats;  
double[] arrayDoubles;  
boolean[] arrayBooleans;  
char[] arrayChars;  
int[] arrayInts;
```

Array de referências (objetos)

```
String[] arrayNomes;  
Cliente[] arrayClientes;
```

Array multidimensionais

```
int[][] matriz;  
String[][] nomes;
```

Sábias palavras...

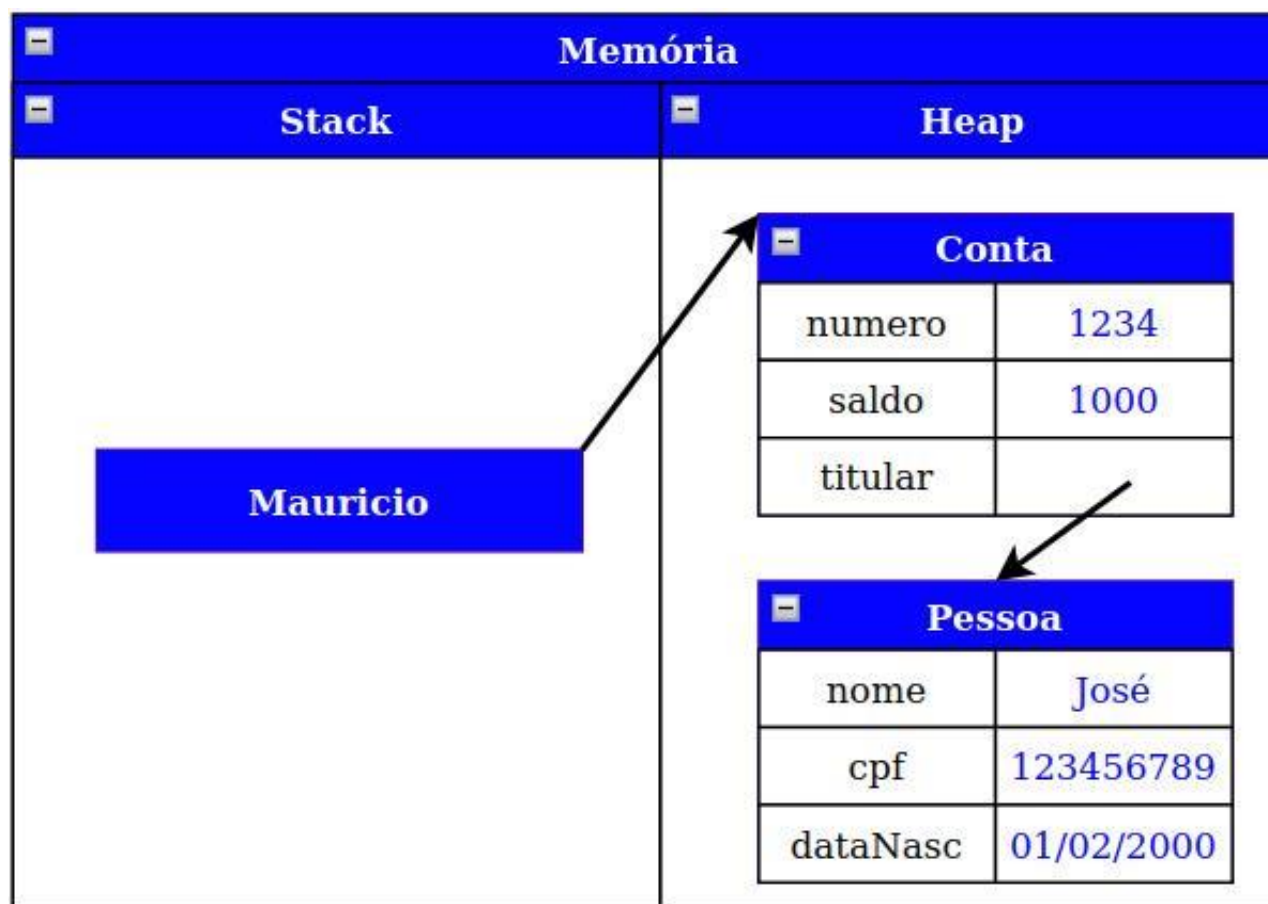
*"Um Array de tipos primitivos guarda valores,
um array de objetos guarda referências."*

Tio Renzo



Não são variáveis

Relembrando objetos em Java



Arrays

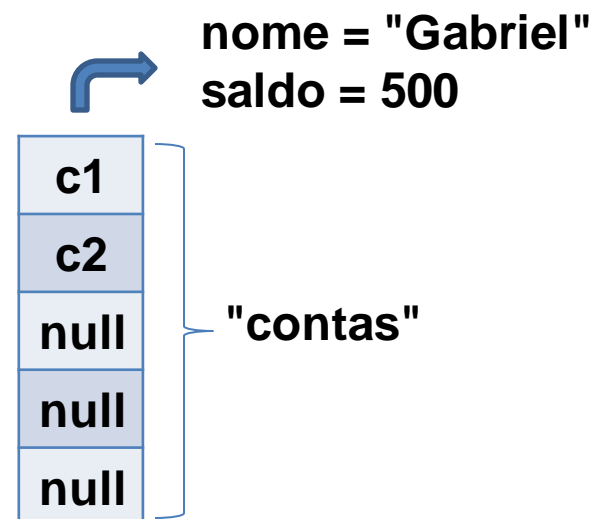
Não basta apenas instanciar o objeto array. O array serve apenas de suporte, é necessário instanciar o elemento que irá ocupar cada posição. Assim como uma estante de livros.

Primeira forma

```
public static void main(String[] args) {  
  
    //CRIANDO UM ARRAY DE CONTAS  
    Conta[] contas = new Conta[5];  
  
    //INSTANCIANDO DUAS CONTAS  
    Conta c1 = new Conta("Gabriel", 500);  
    Conta c2 = new Conta("Timóteo", 200);  
  
    //ATRIBUINDO AS CONTAS CRIADAS ÀS POSIÇÕES DO ARRAY  
    contas[0] = c1;  
    contas[1] = c2;
```

Segunda forma

```
//ALTERNATIVAMENTE, É POSSÍVEL INSTANCIAR O OBJETO NO ARRAY  
contas[2] = new Conta("Fernanda", 1000);  
contas[3] = new Conta("Tadeu", 30);
```



Percorrendo um array do tipo Referência

Acessando elementos específicos

```
//Acessando as informações
//Primeira posição
System.out.println("Nome: " + contas[0].nome);
System.out.println("Saldo: " + contas[0].saldo);

//Segunda posição
System.out.println("Nome: " + contas[1].nome);
System.out.println("Saldo: " + contas[1].saldo);

run:
Nome: Gabriel
Saldo: 500.0
Nome: Timóteo
Saldo: 200.0
BUILD SUCCESSFUL (total time: 0 seconds)
```

Percorrendo um array do tipo Referência

Percorrendo através de um loop

```
//Acessando as informações
//For convencional
for (int i = 0; i < contas.length; i++) {
    System.out.println("Nome: " + contas[i].nome);
    System.out.println("Saldo: " + contas[i].saldo);
}

//Acessando as informações
//For each
for (Conta conta : contas) {
    System.out.println("Nome: " + conta.nome);
    System.out.println("Saldo: " + conta.saldo);
}
```

```
Nome: Gabriel
Saldo: 500.0
Nome: Timóteo
Saldo: 200.0
Nome: Fernanda
Saldo: 1000.0
Nome: Tadeu
Saldo: 30.0
Exception in thread "main" java.lang.NullPointerException
|       at aula5.Aula5.main(Aula5.java:35)
```

A saída dos dois é a mesma. Porém perceba que o Java aponta um erro! Como resolver?

Erro do tipo "NullPointerException"

Primeiro vamos realizar um teste:

```
for (int i = 0; i < contas.length; i++)  
    System.out.println(contas[i]);
```

```
aula5.Conta@5d22bbb7  
aula5.Conta@123772c4  
aula5.Conta@2d363fb3  
aula5.Conta@7d6f77cc  
null  
BUILD SUCCESSFUL (total time: 0 seconds)
```

**As quatro contas que criamos
e seu endereço na memória**

**Posições do Array que estão
"livres", ou seja, não possui
nenhum objeto naquela
posição**

O erro do slide anterior acontece pois estamos tentando acessar um atributo, em uma posição que não existe nenhum objeto do tipo "Conta" instanciado ("null").

Solução

Primeira solução: Verificar se a posição do Array é diferente de nulo ("null")

```
//Acessando as informações
for (int i = 0; i < contas.length; i++) {
    if(contas[i] != null){
        System.out.println("Nome: " + contas[i].nome);
        System.out.println("Saldo: " + contas[i].saldo);
    }
}
```

```
run:
Nome: Gabriel
Saldo: 500.0
Nome: Timóteo
Saldo: 200.0
Nome: Fernanda
Saldo: 1000.0
Nome: Tadeu
Saldo: 30.0
BUILD SUCCESSFUL (total time: 0 seconds)
```

Segunda solução: Instanciar todas as posições do Array com um objeto do tipo "Conta"

```
//Instanciando todas as posições com um objeto do tipo "Conta"
for (int i = 0; i < contas.length; i++) {
    contas[i] = new Conta();
}

for (int i = 0; i < contas.length; i++) {
    System.out.println("Contas: " + contas[i]);
}
```

```
Contas: aula05.Conta@3419866c
Contas: aula05.Conta@59690aa4
Contas: aula05.Conta@6842775d
Contas: aula05.Conta@574caa3f
Contas: aula05.Conta@64cee07
BUILD SUCCESSFUL (total time: 1 second)
```

Agora todas as posições possuem um objeto do tipo "Conta" e nenhuma posição nula ("null")

Percorrendo um array do tipo primitivo

Acessando elementos específicos

```
//Criando um array do tipo "int" e já preenchendo com os valores  
int[] codigos = {100, 200, 300, 400, 500, 600, 700, 800, 900};
```

```
//Mostrando uma informação em uma posição específica  
System.out.println("Código: " + codigos[2]);
```

Código: 300

BUILD SUCCESSFUL (total time: 1 second)

Percorrendo um array do tipo primitivo

Percorrendo através de um loop

```
//Criando um array do tipo "int" e já preenchendo com os valores
int[] codigos = {100, 200, 300, 400, 500, 600, 700, 800, 900};
```

```
//For convencional
for (int i = 0; i < codigos.length; i++) {
    System.out.println(codigos[i]);
}

System.out.println("-----");
```

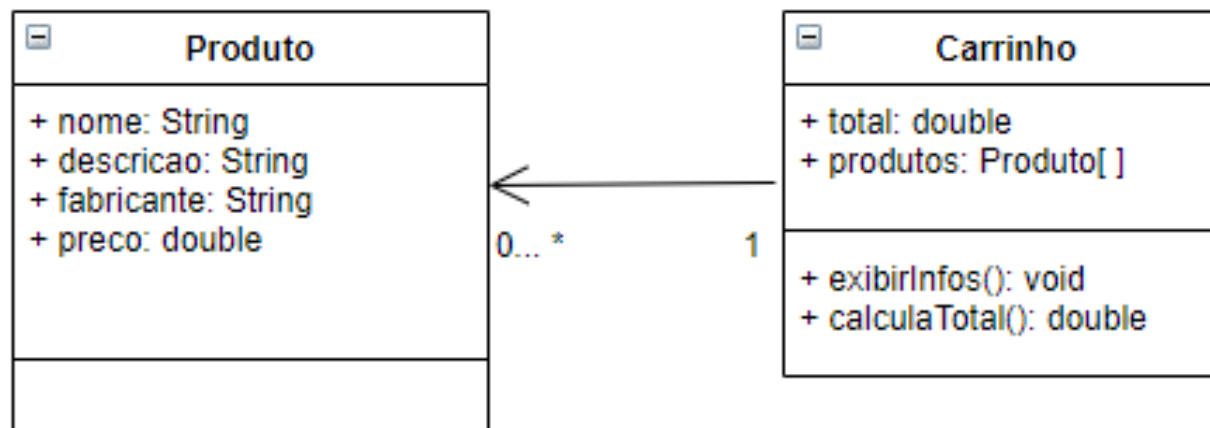
```
//For each
for (int codigo : codigos) {
    System.out.println(codigo);
}
```

```
run:
100,200,300,400,500,600,700,800,900,
BUILD SUCCESSFUL (total time: 1 second)
```

Exercícios

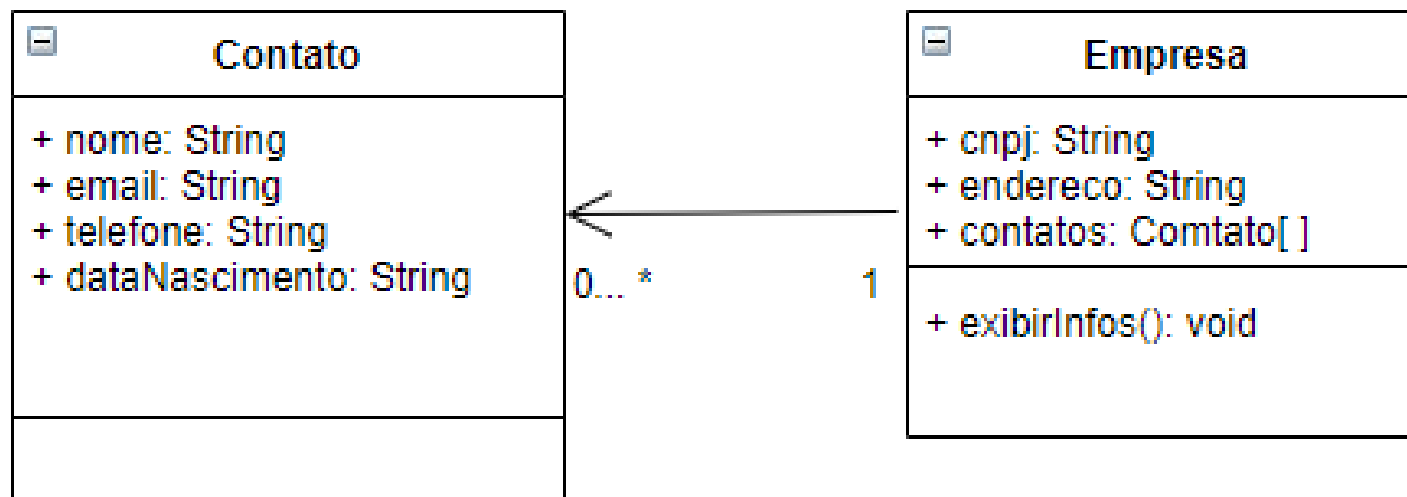
Você está participando de um projeto que irá implantar um e-commerce de um grande varejista global que está iniciando suas operações no Brasil. Atualmente, sua responsabilidade é exibir o carrinho de compras, ou seja, implementar o método que receba os produtos que o consumidor irá comprar e exiba na tela as informações de cada produto: nome, descrição, fabricante e preço. Ao final da tela deverá ser exibido o valor total da compra.

Como um bom desenvolvedor você irá realizar os testes no software antes de assumir a tarefa como concluída (ou seja, implemente uma classe Main para teste).



Exercícios

Implemente um cadastro centralizado dos fornecedores da sua empresa, contendo as seguintes informações: CNPJ, Endereço e contato. Como cada empresa pode ter mais de um contato, armazene de forma separada os dados da pessoa: nome, email, telefone e data de nascimento.

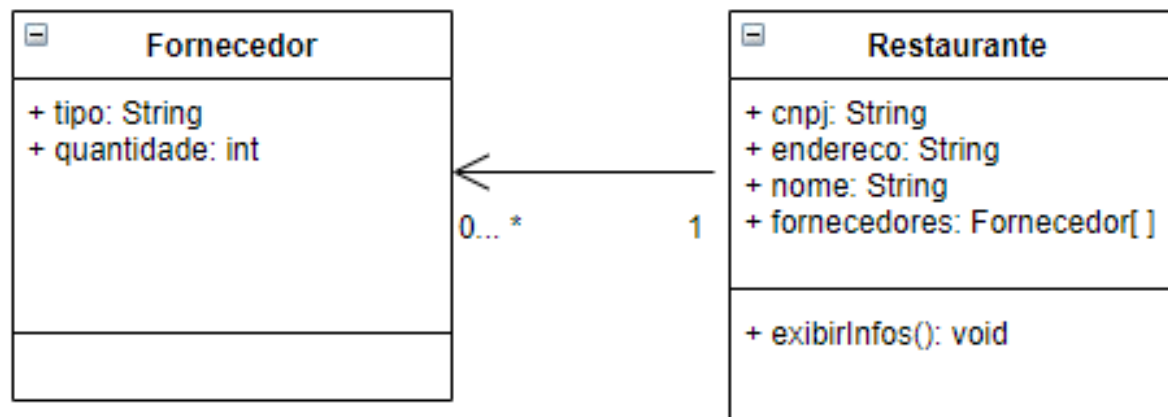


Exercícios

Faça o cadastro dos fornecedores de um determinado restaurante. Sabendo que um restaurante pode ter vários fornecedores e que cada fornecedor possui: tipo e quantidade. Já restaurante possui as seguintes informações: nome, endereço e cnpj. Modele o sistema abaixo e por fim mostre as informações do restaurante e de seus respectivos fornecedores.

Dica:

- 1) Preencha todas as posições do Array com um fornecedor para evitar o erro "NullPointerException".
- 2) Compare as posições do Array com algum atributo específico.



Desafios

Desafio 1: Ordene um array de Strings. (em ordem crescente)

Desafio 2: Os jogos da Mega-Sena são realizados 2 vezes por semana, salvo em datas especiais (como a da Virada), faça um programa que simule, armazene e depois exiba os 105 sorteios anuais (em média). As informações registradas de cada sorteio pela Caixa são: número do concurso, data do sorteio, local, valor do sorteio, números sorteados e ganhadores *.

Nota: para simplificar somente gere se houve ganhador ou não

Dica: use a classe Random

Obrigado!