



Arrays em Java

Introdução

Um *array* ou arranjo é uma estrutura de dados que armazena uma coleção de elementos de tal forma que cada um dos elementos possa ser identificado por, pelo menos, um índice ou uma chave.

Criação de Arrays

Arrays de Tipos Primitivos

Para criar *arrays* a partir de tipos primitivos, como `int`, `float`, `double` etc, devemos seguir o exemplo abaixo:

```
int[] array = new int[tamanhoDoArray];
```

Obs.: O tamanho de um *array* deve ser sempre um valor ou uma variável do tipo inteiro!

Arrays de Objetos

Já para criarmos *arrays* a partir de uma classe para podermos armazenar os dados de vários objetos, devemos fazer o seguinte:

```
MinhaClasse[] array = new MinhaClasse[tamanhoDoArray];
```

Obs.: Os *arrays* apenas armazenam referências aos valores, ou sejam, não criam objetos, mas agem como uma espécie de ponteiro.

Matrizes

Também podemos criar matrizes bidimensionais da seguinte forma:

```
ClasseOuTipo[][] matrix = new ClasseOuTipo[numLinhas][numColunas];
```

Para criarmos matrizes de outros tipos, como cúbicas (tridimensionais) ou matrizes em 4D, por exemplo, só precisaríamos continuar acrescentando “tamanhos”. Segue um exemplo de matriz 4D, para fins de demonstração, apenas, uma vez que suas aplicações são um tanto complexas:

```
int[][][][] matriz4D = new int[3][3][3][3];
```

Manipulação de Arrays

Podemos manipular *arrays*, tanto para inserir quanto para alterar ou excluir dados armazenados em um índice ou posição específica. Podemos fazer isso da seguinte forma:

```
// Alterando o valor de um índice  
array[indice] = novoValor;  
  
// Acessando o valor de um índice  
System.out.println(array[indice]);
```

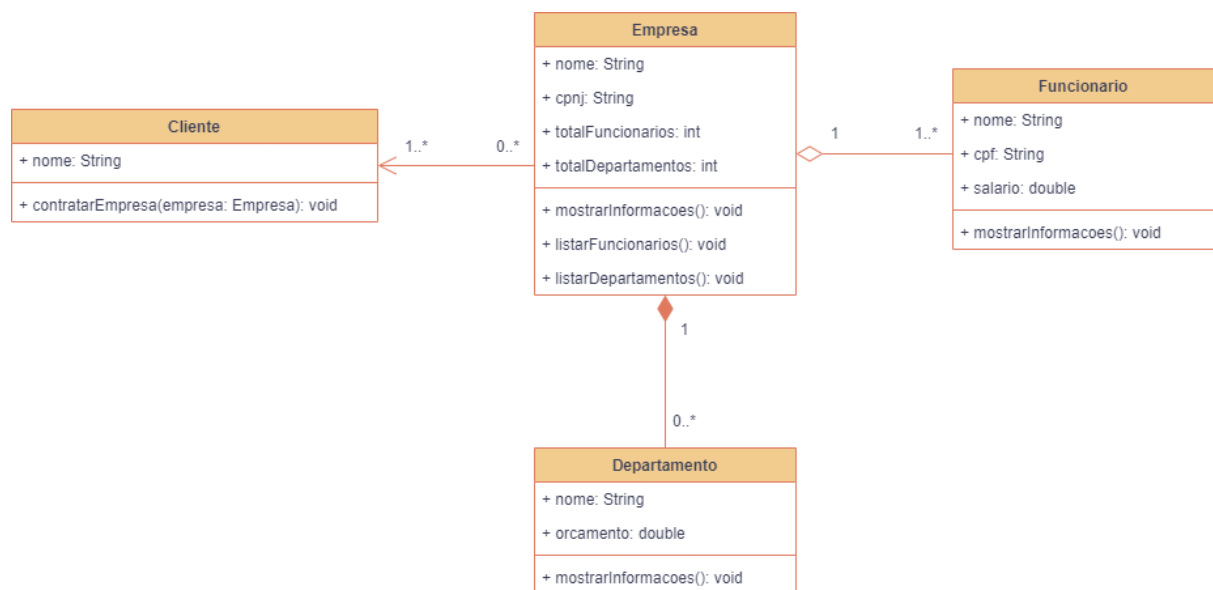
Multiplicidade

Multiplicidade define os limites inferior e superior da quantidade de objetos aos quais outro objeto está associado. Esses limites podem receber os valores: apenas um; zero ou muitos; um ou muitos; zero ou um; ou até um intervalo específico. Segue a tabela abaixo com a notação de cada limite no diagrama.

0..1	Zero ou um.
1..1	Um e somente um.
0..*	Zero ou muitos.
*	Muitos.
1..*	No mínimo um ou muitos.
3..5	Mínimo de três e máximo de cinco.

Exercício

Utilizando nossos conhecimentos sobre multiplicidade, relacionamentos e *arrays*, vamos implementar o diagrama abaixo:



Após a aula, a resolução depois será disponibilizada no GitHub da disciplina:

<https://github.com/bio353/c206>