

Cartilha de Métodos de Array em Java

Introdução

Arrays em Java são estruturas que armazenam múltiplos valores do mesmo tipo. Embora arrays em Java não possuam tantos métodos próprios como as classes ArrayList ou List, é possível manipulá-los com auxílio da classe utilitária Arrays, do pacote java.util.

1. Arrays.toString(array)

Converte o array em uma representação em texto.

```
import java.util.Arrays;
int[] numeros = {1, 2, 3, 4};
System.out.println(Arrays.toString(numeros)); // [1, 2, 3, 4]
```

2. Arrays.sort(array)

Ordena os elementos do array em ordem crescente.

```
int[] valores = {4, 2, 7, 1};
Arrays.sort(valores);
System.out.println(Arrays.toString(valores)); // [1, 2, 4, 7]
```

3. Arrays.copyOf(array, novoTamanho)

Copia um array para outro, com o tamanho especificado.

```
int[] original = {1, 2, 3};
int[] copia = Arrays.copyOf(original, 5);
System.out.println(Arrays.toString(copia)); // [1, 2, 3, 0, 0]
```

4. Arrays.equals(array1, array2)

Verifica se dois arrays são iguais (mesmos elementos e mesma ordem).

```
int[] a = {1, 2, 3};
int[] b = {1, 2, 3};
System.out.println(Arrays.equals(a, b)); // true
```

5. Arrays.fill(array, valor)

Preenche todo o array com um valor específico.

```
int[] vetor = new int[4];
Arrays.fill(vetor, 9);
System.out.println(Arrays.toString(vetor)); // [9, 9, 9, 9]
```

Cartilha de Métodos de Array em Java

6. Arrays.binarySearch(array, valor)

Procura o índice de um valor em um array ordenado. Retorna o índice se encontrado, ou um número negativo se não estiver presente.

```
int[] ordenado = {1, 3, 5, 7};  
int indice = Arrays.binarySearch(ordenado, 5);  
System.out.println(indice); // 2
```

Dicas Importantes

- Antes de usar binarySearch, certifique-se de que o array esteja ordenado.
- Para arrays multidimensionais, use Arrays.deepToString(array) para imprimir.
- Arrays são de tamanho fixo. Para coleções mais flexíveis, use ArrayList.

Conclusão

Embora simples, os arrays são poderosos aliados na programação. Com os métodos da classe Arrays, é possível ordená-los, copiá-los, compará-los e muito mais. Dominar essas ferramentas torna seu código mais eficiente e organizado.