06

Para saber mais: coleções

As coleções são estruturas de dados fundamentais no Java, que nos permitem armazenar e manipular conjuntos de elementos de forma eficiente. Elas são implementadas pela API de coleções do Java, que faz parte do pacote `java.util`. As coleções fornecem uma variedade de classes e interfaces para armazenar e organizar dados de diferentes maneiras, atendendo a diferentes necessidades e cenários.

A API de coleções do Java inclui interfaces, classes abstratas e classes concretas para representar diferentes tipos de coleções. Algumas das principais interfaces de coleções são:

- List: Uma coleção ordenada que permite elementos duplicados. Os

elementos são acessados por índices.

- Set: Uma coleção que não permite elementos duplicados e normalmente não possui ordem definida.
- Queue: Uma coleção que representa uma fila, onde os elementos são adicionados no final e removidos do início.
- Map: Uma coleção de pares chave-valor, onde cada chave é única e mapeada para um valor correspondente.

Além dessas interfaces, a API de coleções também inclui várias classes concretas que implementam essas interfaces, como `ArrayList`, `LinkedList`, `HashSet`, `TreeSet`, `HashMap`, `TreeMap`, entre outras.

Umas das interfaces mais utilizadas é o List. Conforme citado anteriormente, o List é uma interface que define uma sequência ordenada de elementos, onde cada elemento possui uma posição

específica. Ao declarar uma variável desse tipo, estamos indicando que iremos trabalhar com uma coleção que mantém a ordem dos elementos.

Uma das principais razões para usarmos o List é a flexibilidade que ele oferece. Podemos adicionar, remover e acessar elementos de forma fácil e intuitiva. Além disso, ele também permite a duplicação de elementos, ou seja, podemos ter elementos repetidos na coleção. Outra vantagem é a possibilidade de percorrer os elementos de forma sequencial, utilizando laços de repetição, como o for-each, isso facilita a manipulação dos elementos e a realização de operações em massa.

Veja um exemplo prático do uso do List em um código Java:

```
import java.util.List;
import java.util.ArrayLi

public class ExemploList
```

```
    public static void m
        // Criando um ob
        List<Integer> nu

        // Adicionando e
        numeros.add(10);
        numeros.add(20);
        numeros.add(30);

        // Acessando ele
        System.out.print
        System.out.print
        System.out.print

        // Percorrendo o
        for (Integer num
            System.out.p
        }

        // Removendo um
        numeros.remove(1

        // Verificando o
        System.out.print
    }
}
```

COPIAR CÓDIGO

Além disso, o Java também possui outras coleções, como o Set e o Map. O Set é uma

coleção que não permite elementos duplicados, enquanto o Map é uma coleção que associa chaves a valores, permitindo a recuperação rápida de um elemento através de sua chave.

As coleções do Java são extremamente úteis em diversas situações, como armazenar dados em memória, realizar operações de busca, ordenação e filtragem. Elas nos ajudam a organizar e manipular grandes quantidades de dados de forma eficiente e elegante.

Se você deseja se aprofundar no assunto, confira o curso Java:

- Java Collections: Dominando Listas, Sets e Mapas (https://cursos.alura.com.br/course/java-collections)