

Mapa Mental: Interface Collections e suas Subinterfaces

Collections

- A interface base para coleções de objetos em Java, provendo métodos utilitários para manipulação de coleções.
- Este mapa mental ajuda a visualizar as relações hierárquicas e as implementações de coleções na biblioteca Java.

Subinterfaces:

- **List**
 - Representa uma coleção ordenada de elementos, permitindo acesso a elementos por índice.
 - **Classes que implementam List :**
 - ArrayList
 - LinkedList
 - Vector
 - Stack
- **Set**
 - Representa uma coleção que não permite elementos duplicados.
 - **Classes que implementam Set :**
 - HashSet
 - LinkedHashSet
 - TreeSet
- **Queue**

- Representa uma coleção que segue a política FIFO (First In, First Out).
- **Classes que implementam Queue :**
 - LinkedList (também implementa List)
 - PriorityQueue
 - ArrayDeque
- **Deque** (extensão de Queue)
 - Representa uma coleção que permite adicionar e remover elementos de ambos os lados (FIFO e LIFO).
 - **Classes que implementam Deque :**
 - ArrayDeque
 - LinkedList
- **SortedSet** (extensão de Set)
 - Representa uma coleção que mantém os elementos em ordem.
 - **Classes que implementam SortedSet :**
 - TreeSet
- **NavigableSet** (extensão de SortedSet)
 - Representa uma coleção com métodos adicionais para navegação avançada, como busca de elementos.
 - **Classes que implementam NavigableSet :**
 - TreeSet
- **Map**
 - Representa uma coleção de pares chave-valor.
 - **Classes que implementam Map :**
 - HashMap
 - LinkedHashMap
 - TreeMap
 - Hashtable
 - WeakHashMap
 - ConcurrentHashMap

- **SortedMap** (extensão de `Map`)
 - Representa um mapa que mantém as chaves em ordem.
 - **Classes que implementam `SortedMap` :**
 - `TreeMap`
- **NavigableMap** (extensão de `SortedMap`)
 - Representa um mapa com métodos adicionais para navegação avançada.
 - **Classes que implementam `NavigableMap` :**
 - `TreeMap`

Explicação:

- **List** : Permite a duplicação de elementos e fornece acesso a elementos por índice.
- **Set** : Não permite elementos duplicados e pode ser ordenado ou não.
- **Queue** : Define uma fila de elementos com inserção e remoção em um padrão FIFO.
- **Deque** : Uma fila dupla, permitindo inserção e remoção em ambas as extremidades.
- **SortedSet** e **NavigableSet** : Representam subconjuntos de `Set` que mantêm a ordenação dos elementos, com o `NavigableSet` oferecendo métodos mais avançados de navegação.
- **Map** : Representa uma coleção de pares chave-valor, com diferentes implementações, como `HashMap` , `TreeMap` , etc.
- **SortedMap** e **NavigableMap** : Subinterfaces de `Map` , oferecendo ordenação das chaves e métodos adicionais para navegação.