Java Collections: All Functions with Code Examples

Introduction to Java Collections Functions

- Java Collections Framework provides numerous functions to manipulate data structures.
- Functions vary across different interfaces and classes such as List, Set, Map, etc.

List Interface: Common Functions

- add(E e): Appends the specified element to the end of the list.
- remove(Object o): Removes the first occurrence of the specified element.
- get(int index): Returns the element at the specified position.
- Example:
- List<String> list = new ArrayList<>();
- list.add("Apple");

Set Interface: Common Functions

- add(E e): Adds the specified element if it is not already present.
- remove(Object o): Removes the specified element if it is present.
- contains(Object o): Returns true if the set contains the specified element.
- Example:
- Set<String> set = new HashSet<>();

Map Interface: Common Functions

- put(K key, V value): Associates the specified value with the specified key.
- remove(Object key): Removes the mapping for a key if it is present.
- get(Object key): Returns the value to which the specified key is mapped.
- Example:
- Map<String, Integer> map = new HashMap<>();

ArrayList: Common Functions

- add(E e): Appends the specified element to the end of the list.
- remove(int index): Removes the element at the specified position.
- clear(): Removes all elements from the list.
- Example:
- ArrayList<String> arrayList = new ArrayList<>();
- arrayList.add("Apple");

LinkedList: Common Functions

- addFirst(E e): Inserts the specified element at the beginning of the list.
- addLast(E e): Appends the specified element to the end of the list.
- removeFirst(): Removes and returns the first element.
- Example:
- LinkedList<String> linkedList = new LinkedList<>();
- linkedList.addFirst("Apple");

HashSet: Common Functions

- add(E e): Adds the specified element to the set if it is not already present.
- remove(Object o): Removes the specified element if it is present.
- clear(): Removes all elements from the set.
- Example:
- HashSet<String> hashSet = new HashSet<>();
- hashSet.add("Apple");

TreeSet: Common Functions

- add(E e): Adds the specified element to the set if it is not already present.
- first(): Returns the first (lowest) element currently in the set.
- last(): Returns the last (highest) element currently in the set.
- Example:
- TreeSet<String> treeSet = new TreeSet<>();
- treeSet.add("Apple");
- treeSet.first();

HashMap: Common Functions

- put(K key, V value): Associates the specified value with the specified key.
- remove(Object key): Removes the mapping for the specified key if present.
- clear(): Removes all mappings from the map.
- Example:
- HashMap<String, Integer> hashMap = new HashMap<>();
- hashMap.put("Apple", 1);

TreeMap: Common Functions

- put(K key, V value): Associates the specified value with the specified key.
- firstKey(): Returns the first (lowest) key currently in the map.
- lastKey(): Returns the last (highest) key currently in the map.
- Example:
- TreeMap<String, Integer> treeMap = new TreeMap<>();
- treeMap.put("Apple", 1);
- treeMap.firstKey();