

# Java Collections & Utility Classes

## 1. Collections Overview

The Java Collections Framework (JCF) is a set of interfaces and classes for storing and manipulating groups of data as a single unit. It supports operations like searching, sorting, insertion, deletion, and traversal.

## 2. Collection Interfaces

- Collection (root)
- List: Ordered, allows duplicates
- Set: Unordered, no duplicates
- Queue: FIFO, for holding elements prior to processing
- Map: Key-value pairs (not part of Collection interface)

## 3. Collection Implementation Classes

- ArrayList, LinkedList (List)
- HashSet, LinkedHashSet, TreeSet (Set)
- PriorityQueue, ArrayDeque (Queue)
- HashMap, TreeMap, LinkedHashMap (Map)

## 4. Sorting in Collections

- Collections.sort(list): Sorts a list using natural order
- Collections.sort(list, comparator): Custom sort using Comparator
- Arrays.sort(array): Sorts arrays

## 5. Comparable Interface

Used to define the natural ordering of objects.

Method: `int compareTo(T o)`

Example: `this.age - o.age` (ascending sort by age)

## 6. Comparator Interface

Used for external, custom ordering.

Method: `int compare(T o1, T o2)`

Example: `return s1.age - s2.age;`

## 7. Summary: Comparable vs Comparator

Comparable:

- In `java.lang`
- `compareTo(T o)`
- Natural order

Comparator:

- In `java.util`

## Java Collections & Utility Classes

- compare(T o1, T o2)
- Custom order