# Java DSA Practice Questions (Collections, Comparator, Java 8+)

## Strings & Arrays

1. Sort words in a sentence based on word length
2. Sort words by length first, then alphabetically
3. Remove duplicate characters from a string (keep order)
4. Check whether two strings are anagrams
5. Find the first non-repeating character in a string
6. Find the most frequent character in a string
7. Print all characters with maximum frequency
8. String compression (aaabb → a3b2)
9. Decompress a compressed string
10. Check if one string is a rotation of another
11. Find the longest word in a sentence
12. Sort characters in a string by frequency
13. Count vowels and consonants using streams
14. Reverse words in a sentence using collections
15. Group characters of a string by frequency

## Collections (List, Set, Map)

1. Remove duplicates from a list while preserving order
2. Find the most frequent element in a list
3. Find all duplicate elements in a list
4. Find intersection of two lists
5. Find union of two lists
6. Find difference between two lists
7. Convert a list into a set and vice-versa
8. Convert a list of objects into a map
9. Count frequency of words in a sentence
10. Group words by their length
11. Group words by their first character

## Comparator & Comparable

1. Sort a list of strings by length
2. Sort strings by number of vowels
3. Sort integers by absolute difference
4. Sort employees by salary
5. Sort employees by salary (descending), then name
6. Find employee with maximum salary
7. Find employee with minimum salary
8. Sort a map by keys
9. Sort a map by values
10. Sort objects based on multiple fields

## Iterator & Traversal

1. Remove even numbers from a list using Iterator
2. Remove elements while iterating over a map
3. Traverse a list using Iterator
4. Traverse a list using ListIterator
5. Demonstrate fail-fast iterator
6. Demonstrate fail-safe iterator
7. Explain difference between Iterator and ListIterator with code

## Java 8+ (Streams & Lambda)

1. Filter even numbers from a list
2. Square all numbers in a list
3. Find sum of all elements using streams
4. Find maximum and minimum using streams
5. Count occurrences of characters using streams
6. Find first non-repeating character using streams
7. Group elements using Collectors.groupingBy()
8. Convert list to map using Collectors.toMap()
9. Sort list using streams
10. Sort map using streams
11. Find top-K elements using streams

## PriorityQueue & Advanced

1. Find Kth largest element in a list
2. Find Kth smallest element in a list
3. Merge two sorted lists
4. Find median of a list
5. Find top 3 frequent elements
6. Sort elements using PriorityQueue

## Mixed Interview-Level

1. Check if a list is a palindrome
2. Find missing number using collections
3. Find common elements in multiple lists
4. Partition list into even and odd numbers
5. Remove null values from a collection
6. Flatten a list of lists using streams
7. Count distinct elements using streams
8. Convert array → list → set → map
9. Sort strings by custom logic (case-insensitive)
10. Implement word frequency counter using Java 8