

Basic Assignment Questions

Java Collection Framework– 1

1. Iterator Interface Assignments

Q1: Using Iterator to Traverse a List

- Create an ArrayList of integers.
- Use an Iterator to traverse and print each element.

Q2: Removing Elements While Iterating

- Create a list of strings.
- Use an Iterator to remove all strings starting with the letter 'A'.
- Print the modified list.

Q3: Using ListIterator for Bi-Directional Traversal

- Create an ArrayList of five student names.
- Use a ListIterator to traverse the list forward and then backward.

2. Collection Interface Assignments

Q4: Converting an Array to a Collection

- Create an array of integers.
- Convert it into a List using Arrays.asList().
- Print the list.

Q5: Performing Bulk Operations on Collections

- Create two ArrayLists of employee names.
- Use retainAll() to find common names.
- Use removeAll() to remove duplicates.

3. ArrayList Assignments

Q6: Dynamic ArrayList Growth

- Create an ArrayList with an initial capacity of 5.
- Add more than 5 elements and observe the behavior.
- Print its size before and after adding elements.

Q7: Sorting an ArrayList

- Create an ArrayList of names.
- Use Collections.sort() to sort them in ascending and descending order.

Q8: Removing Duplicates from an ArrayList

- Create an ArrayList with duplicate elements.
- Remove duplicates using a HashSet or LinkedHashSet.

4. Comparable Assignments

Q9: Sorting Custom Objects Using Comparable

- Create a Student class with attributes: id, name, and marks.
- Implement Comparable<Student> to sort students based on marks.
- Sort a list of students and print the results.

5. Comparator Assignments

Q10: Sorting Using Comparator

- Modify the Student class to use Comparator<Student>.
- Implement multiple comparators to sort students by name and ID.

Q11: Sorting by Multiple Fields

- Create an Employee class with fields name, age, and salary.
- Implement Comparator to sort employees first by salary, then by name.

Q12: Using Lambda for Sorting

- Use Java 8's `Comparator.comparing()` and lambda expressions to sort a list of products by price.

6. Real-World Scenario Assignments

Q13: Filtering a Collection Using Iterator

- Create a list of books with price and title.
- Remove all books priced below ₹500 using an Iterator.

Q14: Implementing a Custom Collection Sorting Utility

- Write a generic utility method that accepts a list and a Comparator.
- Use it to sort different lists (e.g., students, employees, products).