8. QB - Basic collection (list and set)

📘 **Java Collections Assignment: List & Set (Enhanced For-loop Only)**

🔹 **Section A: Basics of List (10 Questions)**

1. **Add 10 integers to a `List<Integer>` and print all elements using enhanced for-loop.**
2. **Create a `List<String>` with names and print only those names which start with the letter 'A'.**
3. **Create a `List<Integer>` and print the sum of all even numbers.**
4. **Create a `List<String>` and count how many names have more than 5 characters.**
5. **Create a `List<Double>` of prices. Print only those prices that are greater than 100.**
6. **Create a `List<Integer>` and find the maximum and minimum number using enhanced for-loop.**
7. **Create a `List<String>` of names and print all names in uppercase using enhanced for-loop.**
8. **Create a `List<Integer>` and count how many elements are divisible by 3 and 5 both.**
9. **Create a `List<String>` with some duplicate entries. Print all unique elements (without using Set).**
10. **Create a `List<String>` of names and print the name with the longest length.**

🔹 **Section B: Basics of Set (10 Questions)**

11. **Add 10 integers to a `Set<Integer>` (include some duplicates). Print the set to observe how duplicates are handled.**
12. **Create a `Set<String>` of fruits and print all elements using enhanced for-loop.**
13. **Create a `Set<Integer>` and count how many numbers are even.**
14. **Create a `Set<String>` and print all elements whose length is exactly 4.**
15. **Add some country names in a `Set<String>`, and print those which end with the letter ‘a’.**

16. **Create a `Set<Integer>` and check if a specific number (e.g. 10) exists in the set without using `contains()` (using loop).**
17. **Create a `Set<String>` and print the total number of vowels across all strings in the set.**
18. **Create a `Set<Integer>` and print only prime numbers from it.**
19. **Create a `Set<String>` and check how many strings contain the substring “ing”.**
20. **Create a `Set<String>` and print the string with the maximum number of vowels.**

🔹 Section C: List + Set Comparison and Manipulation (10 Questions)

21. **Create a `List<Integer>` with some duplicate numbers. Convert it to a `Set<Integer>` and print both to show difference.**
22. **Given a `List<String>` with some names, remove duplicates manually using only List operations (no Set).**
23. **Given two `List<String>` objects, print common elements (intersection) using enhanced for-loop.**
24. **Given a `List<Integer>` and a `Set<Integer>`, print all elements from the list that are not present in the set.**
25. **Create a `List<String>` and a `Set<String>`. Print elements that are present in both.**
26. **Given a `Set<String>` of words, create a new `Set<String>` with all words in lowercase.**
27. **Create a `List<String>` of full names (e.g. "John Doe"). Print only the first names.**
28. **Create a `Set<String>` of colors. Convert all elements to uppercase and store in a new set. Print both.**
29. **Create a `List<Integer>` with some values. Print all elements in reverse order using only List methods (no Collections.reverse).**
30. **Create a `List<String>` and remove all elements whose length is less than 4 using another list (don’t use removeIf or iterator).**

✅ Rules & Constraints

- **Only use enhanced for-loop (`for (Type element : collection)` syntax).**

- **Avoid using Stream API, traditional for-loops, or iterators.**
- **Use only Java Collections (`List`, `Set`, `ArrayList`, `HashSet`, etc.).**