Difference between HashSet and LinkedHashSet

Contents

Overview

This document is intended to provide difference between HashSet and LinkedHashSet in java.

Difference between HashSet and LinkedHashSet

☑1. Order of Elements

- **HashSet** does **not guarantee any order** of elements. When you iterate over a `HashSet`, the order of elements is **unpredictable** and may change.

- **LinkedHashSet**, on the other hand, **maintains the order in which elements were inserted**. This means when you iterate over it, the elements will appear **exactly in the order they were added** to the set.

☑2. Performance

- `HashSet` is generally **faster** than `LinkedHashSet` because it doesn't maintain any order and has less overhead.

- `LinkedHashSet` is **slightly slower** due to the extra cost of maintaining a **linked list** for insertion order.

- Both have **constant-time performance** (O(1)) for basic operations like add, remove, and contains, but `LinkedHashSet` uses more memory.

☑3. Use Cases

Use Case	Best Choice
Fast access without needing element order	`HashSet`
Maintaining order of unique elements	`LinkedHashSet`
Implementing ordered sets or insertion history	`LinkedHashSet`

Summary Table:

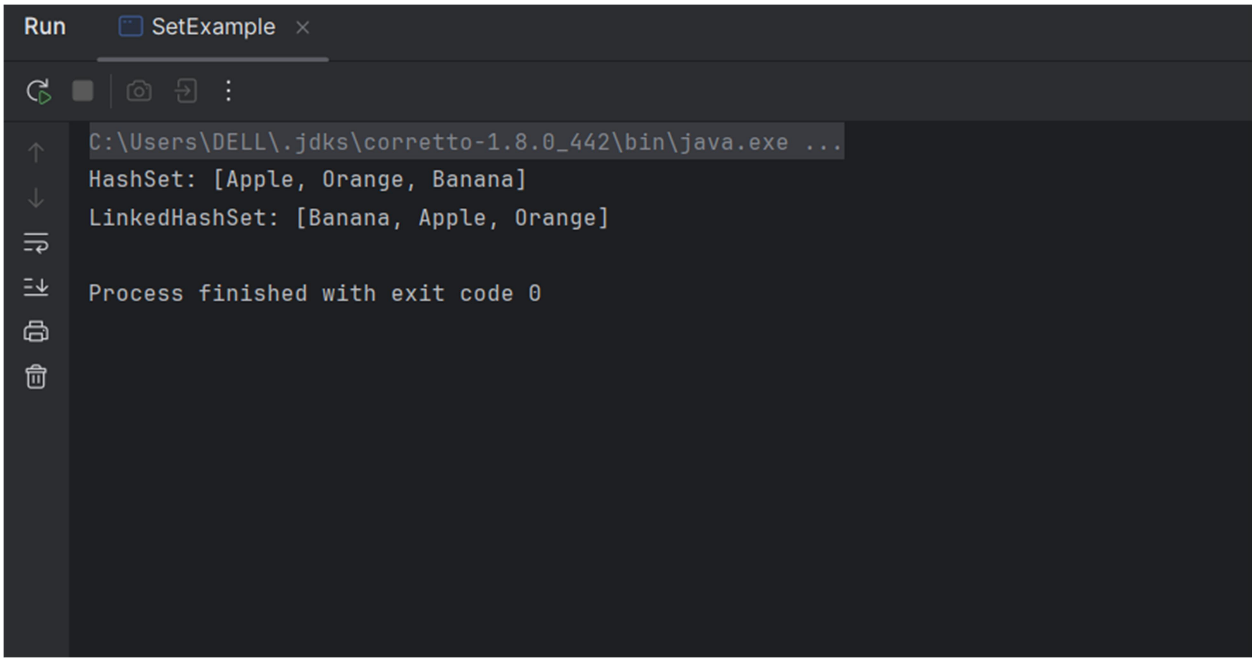
Feature	HashSet	LinkedHashSet
Maintains Order	No	Yes (insertion order)
Iteration Output	Unpredictable	Predictable (in insertion order)
Performance	Faster	Slightly slower
Memory Usage	Lower	Higher (due to linked structure)
Use Case	Fast, unordered set operations	Ordered set operations (e.g., history)

Example:

Java Code

```
import java.util.HashSet;
import java.util.LinkedHashSet;
import java.util.Set;

public class SetExample {
    public static void main(String[] args) {
        Set<String> hashSet = new HashSet<>();
        Set<String> linkedHashSet = new LinkedHashSet<>();

        hashSet.add("Banana");
        hashSet.add("Apple");
        hashSet.add("Orange");

        linkedHashSet.add("Banana");
        linkedHashSet.add("Apple");
        linkedHashSet.add("Orange");

        System.out.println("HashSet: " + hashSet);             // Random order
        System.out.println("LinkedHashSet: " + linkedHashSet); // Insertion order
    }
}
```

Output:



```
C:\Users\DELL\.jdks\corretto-1.8.0_442\bin\java.exe ...
HashSet: [Apple, Orange, Banana]
LinkedHashSet: [Banana, Apple, Orange]

Process finished with exit code 0
```