**LinkedHashSet Class in Java with Example**

[**JAVA COLLECTIONS**](http://beginnersbook.com/category/java-collections/)

Earlier we have shared tutorials on [**HashSet**](http://beginnersbook.com/2013/12/hashset-class-in-java-with-example/) and [**TreeSet**](http://beginnersbook.com/2013/12/treeset-class-in-java-with-example/).[**LinkedHashSet**](http://docs.oracle.com/javase/6/docs/api/java/util/LinkedHashSet.html) is also an implementation of Set interface, it is similar to the HashSet and TreeSet except the below mentioned differences:

1. HashSet doesn’t maintain any kind of order of its elements.
2. TreeSet sorts the elements in ascending order.
3. LinkedHashSet maintains the insertion order. Elements gets sorted in the same sequence in which they have been added to the Set.

**Example of LinkedHashSet:**

import java.util.LinkedHashSet;

public class LinkedHashSetExample {

public static void main(String args[]) {

// LinkedHashSet of String Type

LinkedHashSet<String> lhset = new LinkedHashSet<String>();

// Adding elements to the LinkedHashSet

lhset.add("Z");

lhset.add("PQ");

lhset.add("N");

lhset.add("O");

lhset.add("KK");

lhset.add("FGH");

System.out.println(lhset);

// LinkedHashSet of Integer Type

LinkedHashSet<Integer> lhset2 = new LinkedHashSet<Integer>();

// Adding elements

lhset2.add(99);

lhset2.add(7);

lhset2.add(0);

lhset2.add(67);

lhset2.add(89);

lhset2.add(66);

System.out.println(lhset2);

}

}

**Output:**

[Z, PQ, N, O, KK, FGH]

[99, 7, 0, 67, 89, 66]

Observe the output: Both types of LinkedHashSet have preserved the insertion order.