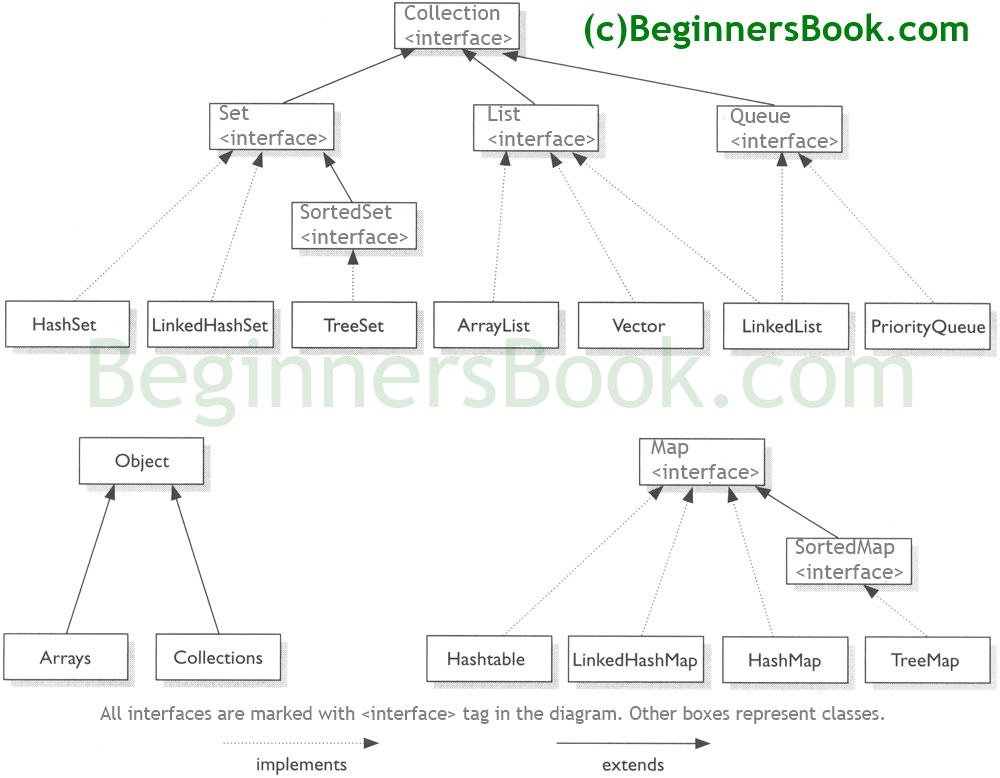
**Collections Framework hierarchy**



## Java Collections – Table of Contents

1. [ArrayList](https://beginnersbook.com/java-collections-tutorials/#1)  
2. [LinkedList](https://beginnersbook.com/java-collections-tutorials/#2)  
3. [Vector](https://beginnersbook.com/java-collections-tutorials/#3)  
4. [HashSet](https://beginnersbook.com/java-collections-tutorials/#4)  
5. [LinkedHashSet](https://beginnersbook.com/java-collections-tutorials/#5)  
6. [TreeSet](https://beginnersbook.com/java-collections-tutorials/#6)  
7. [HashMap](https://beginnersbook.com/java-collections-tutorials/#7)  
8. [TreeMap](https://beginnersbook.com/java-collections-tutorials/#8)  
9. [LinkedHashMap](https://beginnersbook.com/java-collections-tutorials/#9)  
10. [Hashtable](https://beginnersbook.com/java-collections-tutorials/#10)  
11. [Iterator and ListIterator](https://beginnersbook.com/java-collections-tutorials/#11)  
12. [Comparable and Comparator](https://beginnersbook.com/java-collections-tutorials/#12)  
13. [Java Collections Interview Questions](https://beginnersbook.com/java-collections-tutorials/#13)

**1)ArrayList**

**HashMap** - Key value pair and is not sorted i.e. we dont get values in the same sequence as entered

Syntax : HashMap<Integer, String> hmap = new HashMap<Integer, String>();

Adding elements to hashMap :hmap.put("");

Iterating HashMap :

Method 1 :Using for loop

System.out.println("For Loop:");

for (Map.Entry me : hmap.entrySet()) {

System.out.println("Key: "+me.getKey() + " & Value: " + me.getValue());

}

Method 2 :Using While and iterator

Iterator iterator = hmap.entrySet().iterator();

while (iterator.hasNext()) {

Map.Entry me2 = (Map.Entry) iterator.next();

System.out.println("Key: "+me2.getKey() + " & Value: " + me2.getValue());

}

Size of hashMap : hmap.size();

Delete element from hashMap : hmap.delete(key of element to be deleted);

Delete all elements from hashMap : hmap.clear();

Check if hashMap is empty : hmap.isEmpty()

**HashSet** -Doesn’t maintain any order, doesn’t allow duplicates(Duplicate values are overwritten),allows single NULL value entry.

Declaration syntax :

HashSet<String> hset =

new HashSet<String>();

Add elements to HashSet : hset.add(“element”);

Size of HashSet: hset.size();

Delete element from HashSet: hset.delete(key of element to be deleted);

Delete all elements from hashMap : hset.clear();

Check if HashSet is empty : hset.isEmpty()