

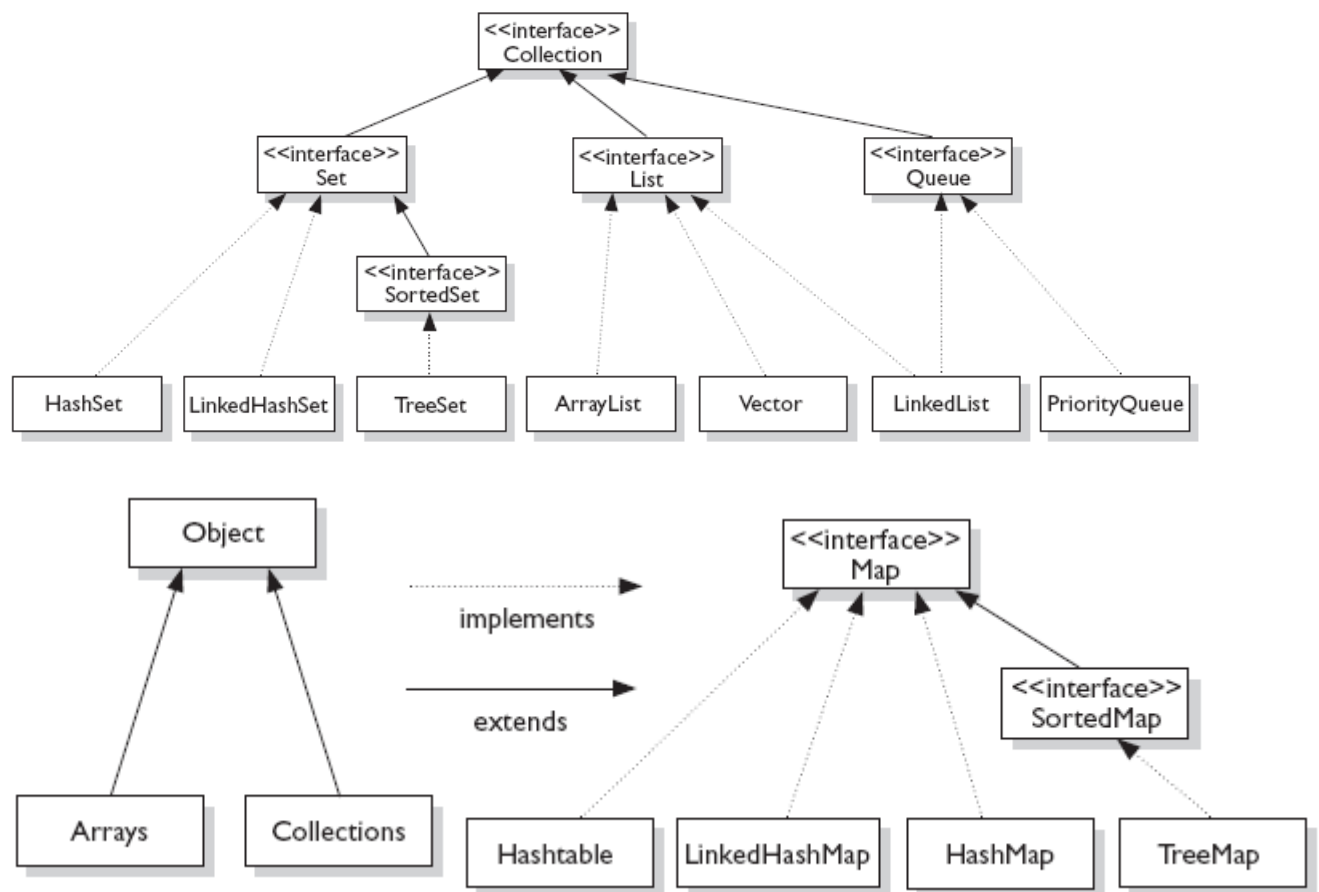
Contents

Collections

Collections

Collection is a group of objects.

Collection Interfaces & Implementations



Various Collections

Factors	Key-value or only values	Ordered or Sorted	Duplicates (Y/N)	Null values	Synchronize (Y/N)
ArrayList [Growable & powerful than String]	values	Ordered	Yes	Yes	No. Therefore, faster than vectors
HashSet [Uses hashcodes]	values	Not sorted, Not Ordered	No	Single null value	Yes
TreeSet	values	sorted	No		No
HashMap [Uses hashcodes]	Key-value	Not sorted, Not Ordered	Unique key. Duplicate values allowed	one null key & multiple null values	No
TreeMap	Key-value	Sorted on keys	Duplicate values allowed	No null key. One null value	No
LinkedHashMap	Key-value	Ordered			
LinkedHashSet	values	Ordered			
Vector	[growable & used instead of arrays]				Yes
HashTable [Uses hashcodes]	Key-value	Not sorted, Not Ordered	Duplicate values allowed	No null key. No null value	Yes

Collection Methods

<code>int size();</code>	Returns number of elements in collection
<code>boolean isEmpty();</code>	Returns true if collection is empty
<code>boolean contains(object element);</code>	Returns true if element is present in the collection
<code>boolean add(object element);</code>	Adds element to collection
<code>boolean remove(object element);</code>	Removes element to collection
<code>Iterator iterator();</code>	Returns and iterators from the collection
<code>void clear();</code>	Removes all elements from collection

Autoboxing & Unboxing

	Prior to Java 5	Since Java 5
Boxing	<pre>int iVal = 11; Integer iRef = new Integer(iVal);</pre>	<pre>int iVal = 11; Integer iRef = iVal;</pre>
Unboxing	<pre>int iVal = iRef.intValue();</pre>	<pre>iVal = iRef;</pre>