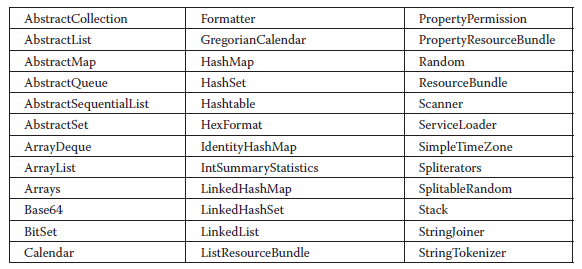
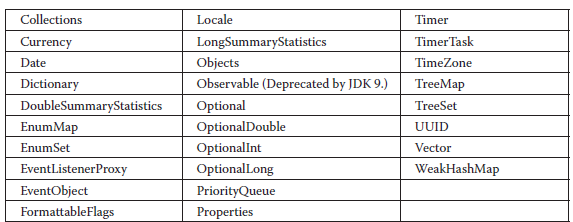
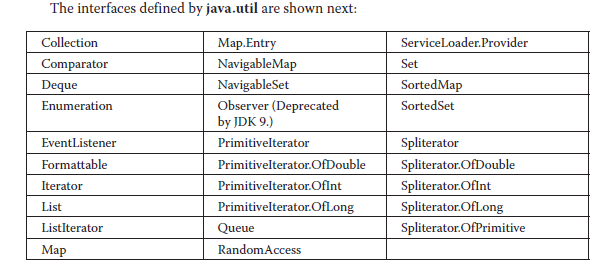
the *Collections Framework*. The Collections Framework is a sophisticated hierarchy of interfaces and classes that provide state-of-the-art technology for managing groups of objects.

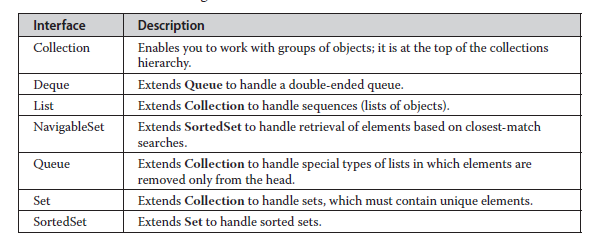






The Java Collections Framework standardizes the way in which groups of objects are handled by your programs.

The interfaces that underpin collections are summarized in the following table:



**The List Interface**

The **List** interface extends **Collection** and declares the behavior of a collection that stores a sequence of elements

Elements can be inserted or accessed by their position in the list, using a zero-based index. A list may contain duplicate elements. **List** is a generic interface that has this declaration:

interface List<E>

Here, **E** specifies the type of objects that the list will hold.

**The Set Interface**

The **Set** interface defines a set. It extends **Collection** and specifies the behavior of a collection that does not allow duplicate elements. Therefore, the **add( )** method returns **false** if an attempt is made to add duplicate elements to a set.

**The SortedSet Interface**

The **SortedSet** interface extends **Set** and declares the behavior of a set sorted in ascending order. **SortedSet** is a generic interface that has this declaration:

interface SortedSet<E>

**The NavigableSet Interface**

The **NavigableSet** interface extends **SortedSet** and declares the behavior of a collection that supports the retrieval of elements based on the closest match to a given value or values. **NavigableSet** is a generic interface that has this declaration:

interface NavigableSet<E>

**The Queue Interface**

The **Queue** interface extends **Collection** and declares the behavior of a queue, which is often a first-in, first-out list. However, there are types of queues in which the ordering is based upon other criteria. **Queue** is a generic interface that has this declaration:

interface Queue<E>

**The Deque Interface**

The **Deque** interface extends **Queue** and declares the behavior of a double-ended queue. Double-ended queues can function as standard, first-in, first-out queues or as last-in, first-out stacks. **Deque** is a generic interface that has this declaration:

interface Deque<E>

**The Collection Classes**

The core collection classes are summarized in the following table:

