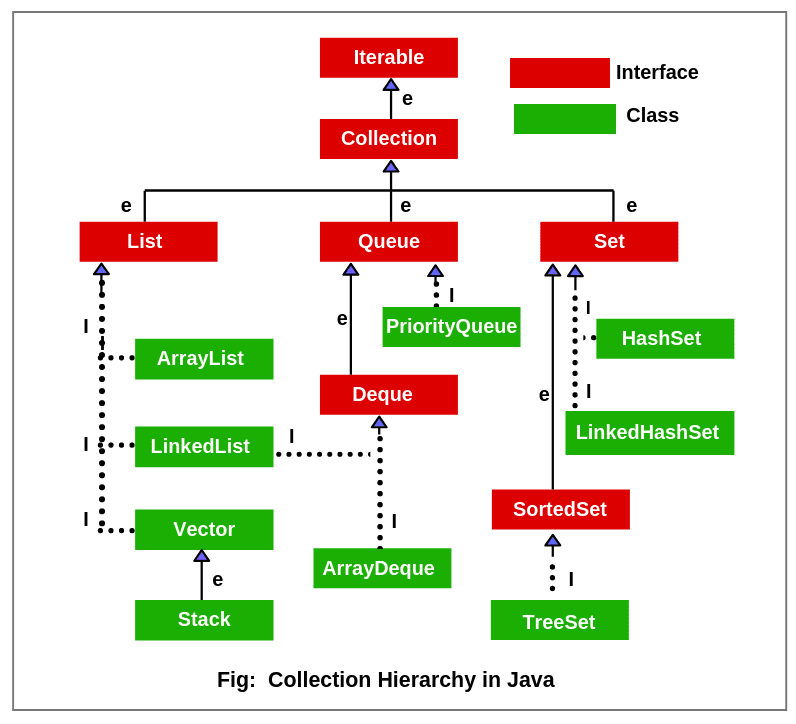
6) Describe the collection interface hierarchy in java with diagrammatic representation

The hierarchy of the entire collection framework consists of four core interfaces such as Collection, List, Set, Map, and two specialized interfaces named SortedSet and SortedMap for sorting.

All the interfaces and classes for the collection framework are located in [java.util.package](https://docs.oracle.com/javase/8/docs/api/java/util/package-summary.html). The diagram of Java collection hierarchy is shown in the below figure.



**Collection Interface in Java**

The basic interface of the collections framework is the Collection interface which is the root interface of all collections in the API (Application programming interface). It is placed at the top of the collection hierarchy in java. It provides the basic operations for adding and removing elements in the collection.

The Collection interface extends the iterable interface. The iterable interface has only one method called iterator (). The function of the iterator method is to return the iterator object. Using this iterator object, we can iterate over the elements of the collection.

List, Queue, and Set have three component which extends the Collection interface. A map is not inherited by Collection interface.

**List Interface**

This interface represents a collection of elements whose elements are arranged sequentially ordered.

List maintains an order of elements means the order is retained in which we add elements, and the same sequence we will get while retrieving elements.

We can insert elements into the list at any location. The list allows storing duplicate elements in Java.

ArrayList, Vector, and LinkedList are three concrete subclasses that implement the list interface.

**Set Interface**

This interface represents a collection of elements that contains unique elements. i.e., it is used to store the collection of unique elements.

Set interface does not maintain any order while storing elements and while retrieving, we may not get the same order as we put elements.  All the elements in a set can be in any order.

Set does not allow any duplicate elements.

HashSet, LinkedHashSet, TreeSet classes implements the set interface and sorted interface extends a set interface.

It can be iterated by using Iterator but cannot be iterated using ListIterator.

**SortedSet Interface**

This interface extends a set whose iterator transverse its elements according to their natural ordering.

TreeSet implements the sorted interface.

**Queue Interface**

A queue is an ordered of the homogeneous group of elements in which new elements are added at one end(rear) and elements are removed from the other end(front). Just like a queue in a supermarket or any shop.

This interface represents a special type of list whose elements are removed only from the head.

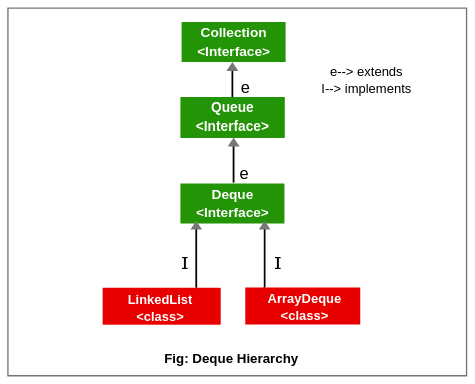
LinkedList, Priority queue, ArrayQueue, Priority Blocking Queue, and Linked Blocking Queue are the concrete subclasses that implements the queue interface.

**Deque Interface**

A deque (double-ended queue) is a sub-interface of queue interface. It is usually pronounced “deck”.

This interface was added to the collection framework in Java SE 6.

Deque interface extends the queue interface and uses its method to implement deque. The hierarchy of the deque interface is shown in the below figure.



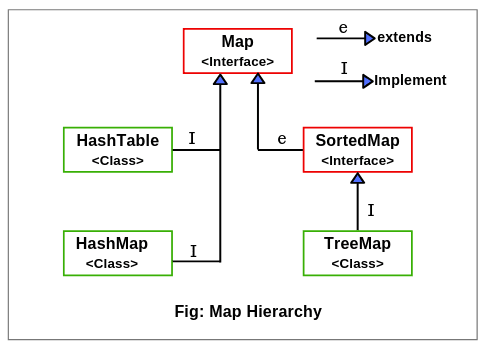
It is a linear collection of elements in which elements can be inserted and removed from either end. i.e., it supports insertion and removal at both ends of an object of a class that implements it.

LinkedList and ArrayDeque classes implement the Deque interface.

**Map Interface**

Map interface is not inherited by the collection interface. It represents an object that stores and retrieves elements in the form of a Key/Value pairs and their location within the Map are determined by a Key.

The hierarchy of the map interface is shown in the below figure.



Map uses a hashing technique for storing key-value pairs.

It doesn’t allow to store the duplicate keys but duplicate values are allowed.

HashMap, HashTable, LinkedHashMap, TreeMap classes implements Map interface.

**SortedMap Interface**

This interface represents a Map whose elements are stored in their natural ordering. It extends the Map interface which in turn is implemented by TreeMap classes.