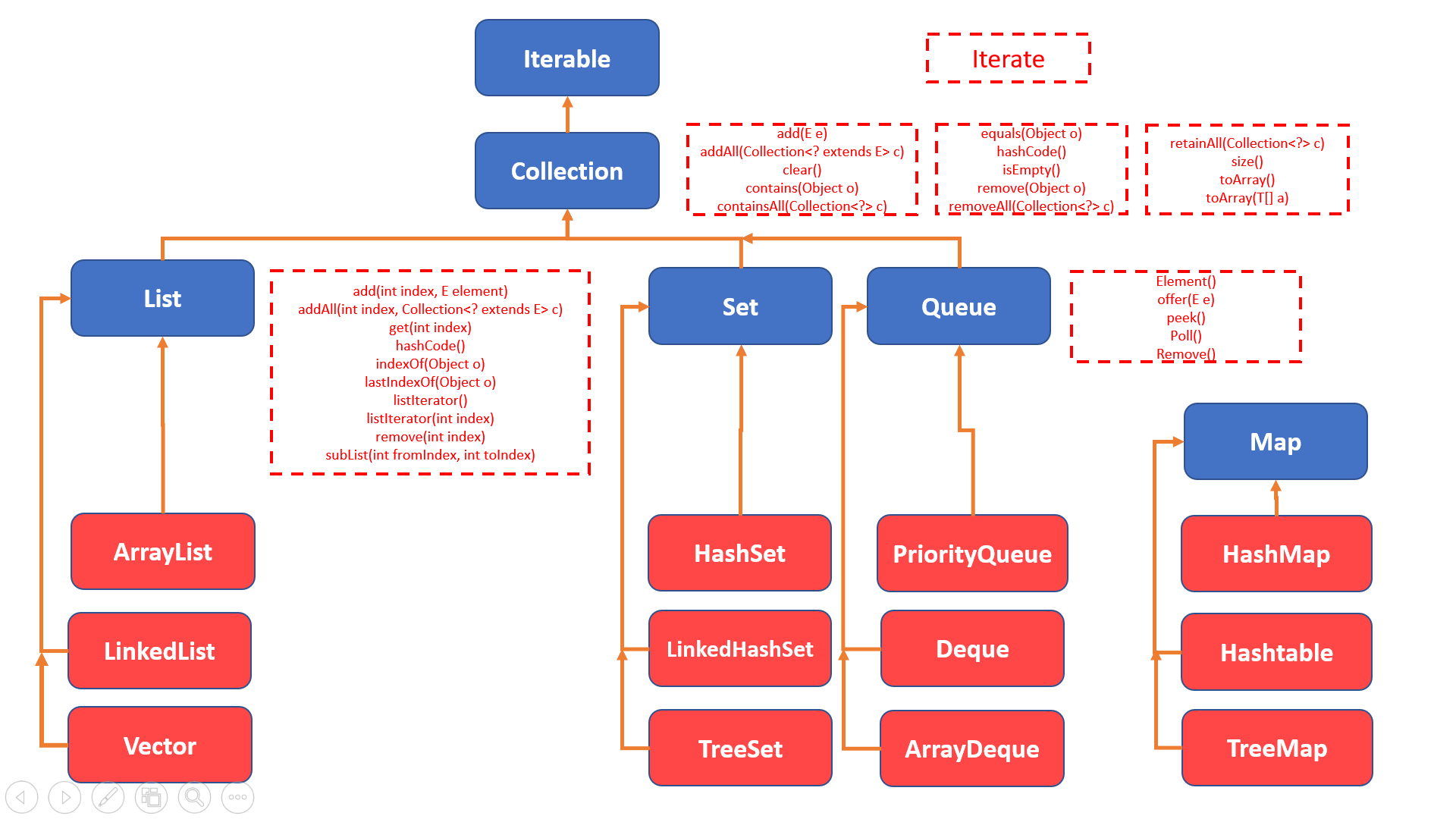
**Java Collections Framework:**

The main disadvantage with Arrays is, they are of fixed size. If we declare an array of length 3, we can add only 3 elements into it. We can’t change the size of the array. We can’t increase it or decrease it.

If we want to have more or less elements into it, we need to have an extendable or shrinkable arrays and Collections framework is an answer for that.

List, Set, Queue and Map are four important interfaces of Collections Framework. Map doesn’t implement Collection interface but it is a part of Collections Framework.



The first interface in collections framework is “Iterable” interface and it defined a method “Iterate” which is to iterate over the collections.

The second one is “Collection” interface which has defined additional methods.

The next interfaces are List, Set and Queue and all of these are extending the “Collection” framework and defined some more additional methods.

Different classes like ArrayList, HashSet etc., were developed which implement these interfaces.

Map doesn’t extend Collection interface but it is still under collections framework.

You can have a look at the following link (oracle docs) for more information about the interfaces and classes developed.

<https://docs.oracle.com/javase/7/docs/api/java/lang/Iterable.html>

**Differences between List, Set, Queue and Map:**

List in java provides ordered and indexed collection which may contain duplicates.

The Set interface provides an unordered collection of unique objects ie Set doesn’t allow duplicates.

Queue works on FIFO (First In First Out) basis. In FIFO the first element is removed frist and the last element is last.

Map provides a data structure based on key value pair.

**List – Differences between ArrayList and LinkedList**

**ArrayList:**

1. ArrayList internally uses dynamic array to store the elements
2. Manipulation with ArrayList is **slow** because it internally uses array. If any element is removed from the array, all the bits are shifted in memory.
3. ArrayList is better for storing and accessing data.

**LinkedList:**

1. LinkedList internally uses doubly linked list to store the elements.
2. Manipulation with LinkedList is faster than ArrayList because it uses doubly linked list so no bit shifting is required in memory.
3. LinkedList is better for manipulating data.