**Difference between Collection<E> remove(Object obj) method and Iterator<E> remove() method.**

[Collection<E>](http://data-structure-learning.blogspot.com/2015/05/java-collections-part-4collection.html) interface has remove(Object obj) method that removes specified object from Collection.

[List<E>](http://data-structure-learning.blogspot.com/2015/05/java-collections-part-5list-interface.html) interface adds remove(int index) method that removes object from specified index.

[Iterator<E>](http://data-structure-learning.blogspot.com/2015/05/java-collections-part-8-iterator.html) interface has remove() method that removes last object returned by next() method.

Now, you can remove(Object obj) of Collection<E> interface and remove(int index) of List<E> interface to remove object from underlying Collection.

But, while iterating you cannot use them. Because if you do so them it will throw [ConcurrentModificationException](http://data-structure-learning.blogspot.com/2015/05/concurrentmodificationexception.html).

While iterating the collection we have to use remove() method of Iterator<E> interface. There is no exception to this rule.

So it is advised to use remove() method of Iterator<E> to remove object from collection while iterating.