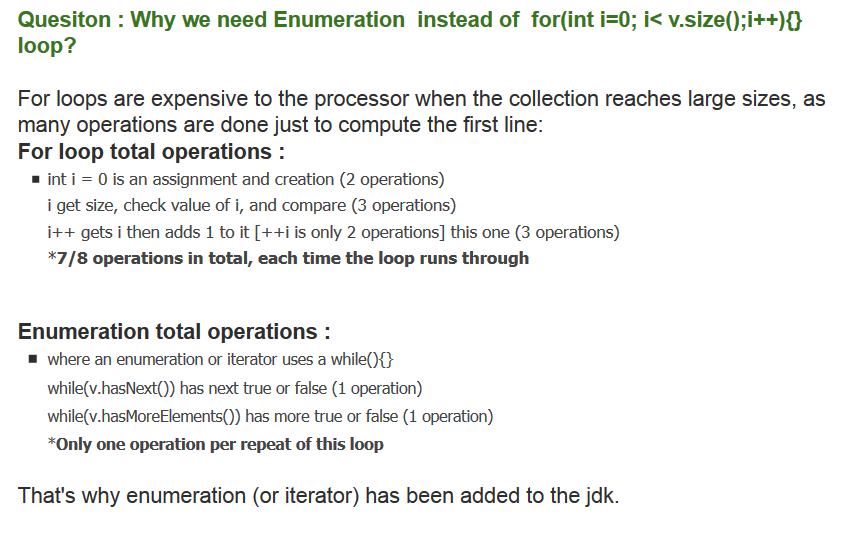
**Iterator and Enumeration with example**

The main difference between Iterator and Enumeration is removal of the element while traversing the collection.

|  |  |
| --- | --- |
| **Iterator** | **Enumeration** |
| Iterator can remove the element during traversal of collection as it has remove() method. | Enumeration does not have remove() method. |
| Iterator is the interface and found in the java.util package. It has three methods  \*hasNext() \*next() \*remove() | Enumeration is also an interface and found in the java.util package . It is used for passing through a collection, usually of unknown size.  It has following methods  \*hasMoreElements() \*nextElement() |
| Iterator has the remove() method | Enumeration does not have remove() method. |
| Enumeration interface acts as a read only interface, one can not do any modifications to Collection while traversing the elements of the Collection. | Iterator can do modifications (e.g using remove() method it removes the element from the Collection during traversal). |
| Iterator is added in jdk1.2 version. | Enumeration is added to the jdk1.0 version |
| Iterator is fail-fast in nature. It throws ConcurrentModificationException if a Collection is modified while iterating other than its own remove() method. | Enumeration is fail-safe in nature. It does not throw ConcurrentModificationException if Collection is modified during the traversal. |
| Iterator is not a legacy interface. Iterator can be used for the traversal of HashMap, LinkedList,  ArrayList, HashSet, TreeMap, TreeSet . | Enumeration is a legacy interface which is used for traversing Vector, Hashtable. |

**Note:**

The functionality of Enumeration  is duplicated by the iterator interface.Iterator adds an optional remove operation,and has shorter method names. New implementations should consider using Iterator in preference to Enumeration.

****