**Q1  What is Collection ? What is a Collections Framework ? What are the benefits of Java Collections Framework ?**  
  
**Collection :** A collection is an object  that is used for representing group of objects as a single entity.  
  
**Collections Framework :**Collections framework provides unified architecture for storing and manipulating group of objects.  
  
**Benefits of Collections Framework :**  
1. Improves program quality and speed  
2. Increases the chances of reusability of software  
  
**Q2 What is the root interface in collection hierarchy ?**  
  
Root interface in collection hierarchy is **Collection interface .**Few interviewer may argue that Collection interface extends **Iterable interface**. So iterable should be the root interface. But you should reply iterable interface present in java.lang package not in java.util package .It is clearly mentioned in [Oracle Collection  docs](http://docs.oracle.com/javase/7/docs/api/java/util/Collection.html) , that Collection interface is a member of the Java Collections framework.  For [Iterable interface Oracle doc](https://docs.oracle.com/javase/7/docs/api/java/lang/Iterable.html" \t "_blank) , iterable interface is not mentioned as a part of the Java Collections framework .So if the question includes  collection hierarchy , then you should answer the question as Collection interface (which is found in java.util package).  
 **Q3 What is the difference between Collection and Collections ?**  
  
Collection is an interface while Collections is a java class , both are present in java.util package and  part of java collections framework.

**Q4 Which collection classes are synchronized or thread-safe ?**  
  
Stack, Properties , Vector and Hashtable can be used in multi threaded environment because they are synchronized classes (or thread-safe).

**Q5 Name the core Collection  interfaces ?**Important : Collection , Set , Queue , List , Map  
  
Other interface also in the list :  SortedSet, SortedMap , Deque, ListIterator etc.  
 **Q6 What is the difference between List and Set ?**  
  
List allows duplicated values whereas set is not allowed duplicate values.  
Set is unordered while List is ordered. List maintains the order in which the objects are added.  
  
**Q7 What is the difference between Map and Set ?**  
  
Set contain only values where as Map contains entry i.e both key and value.

**Q8 What are the classes implementing List and Set interface ?**  
  
***Class implementing List interface :***  ArrayList , Vector , LinkedList ,  
  
***Class implementing Set interface :***HashSet , TreeSet

**Q9 What is an iterator ?**  
  
Iterator is an interface . It is found in java.util package. It is used to retrieve elements from any collection object.  
  
**Q10 What is the difference between Iterator and Enumeration ?**  
  
The main difference between Iterator and Enumeration is that Iterator has remove() method while Enumeration doesn't.  
Hence , using Iterator we can manipulate objects by adding and removing the objects from the collections. Enumeration behaves like a read only interface as it can only traverse the objects and fetch it .  
  
**Q12 Which methods you need to override to use any object as key in HashMap ?**  
  
To use any object as key in HashMap , it needs to implement equals() and hashCode() method .  
  
**Q13 What is the difference between Queue and Stack ?**  
  
Queue is a data structure which is based on FIFO ( first in first out ) property . An example of Queue in real world is buying movie tickets in the multiplex or cinema theaters.  
  
Stack is a data structure which is based on LIFO (last in first out) property. An example of Stack in real world is insertion or removal of CD from the CD case.  
  
**Q14 How to reverse the List in Collections ?**  
  
There is a built in reverse method in Collections class . reverse(List list) accepts list as parameter.  
  
**Collections.reverse(listobject);**  
  
**Q15 How to convert the array of strings into the list ?**  
  
Arrays class of java.util package contains the method asList() which accepts the array as parameter.So,  
  
**String[]  wordArray =  {"Love Yourself"  , "Alive is Awesome" , "Be in present"};**  
**List wordList =  Arrays.asList(wordArray);**  
  
**16 What is the difference between ArrayList and Vector ?**

It is one of the frequently asked collection interview question , the main differences are  
Vector is synchronized while ArrayList is not . Vector is slow while ArrayList is fast . Every time when needed, Vector increases the capacity twice of its initial size while ArrayList increases its ArraySize by 50%. find detailed explanation   [ArrayList vs Vector](http://javahungry.blogspot.co.uk/2013/12/difference-between-arraylist-and-vector-in-java-collection-interview-question.html" \t "_blank)  .  
  
**Q17 What is the difference between HashMap and Hashtable ?**  
  
It is one of the most popular collections interview question for java developer . Make sure you go through this once before appearing for the interview .  
Main differences between HashMap and Hashtable are :  
  
a. Hash Map allows one null key and any number of null values while Hashtable does not allow null keys and null values.  
b. Hash Map is not synchronized or thread-safe while Hashtable is synchronized or thread-safe .  
  
**Q19 What is the difference between Iterator and ListIterator.**  
  
Using Iterator we can traverse the list of objects in forward direction . But ListIterator can traverse the collection in both directions that is forward as well as backward.  
  
**Q20 What is the difference between Array and ArrayList in Java ?**  
  
Main differences between Array and ArrayList are :  
a. Array is static in size while ArrayList is dynamic in size.  
b. Array can contain primitive data types while ArrayList can not contain primitive data types.

**Q21 What is the difference between HashSet and TreeSet ?**  
  
Main differences between HashSet and TreeSet are :  
a.  HashSet maintains the inserted elements in random order while TreeSet maintains elements in the sorted order  
b. HashSet can store null object while TreeSet can not store null object.  
  
**Q23 What is the difference between HashMap and ConcurrentHashMap ?**  
  
This is also one of the most popular java collections interview question . Make sure this question is in your to do list before appearing for the interview .  
Main differences between HashMap and ConcurrentHashMap are :  
a. HashMap is not synchronized while ConcurrentHashMap is synchronized.  
b. HashMap can have one null key and any number of null values while ConcurrentHashMap does not allow null keys and null values .

**Q24 Arrange the following in the ascending order (performance):**  
**HashMap , Hashtable , ConcurrentHashMap and Collections.SynchronizedMap**  
  
Hashtable  <  Collections.SynchronizedMap  <  ConcurrentHashMap  <  HashMap

**Q26 What is the difference between LinkedList and ArrayList in Java ?**  
  
Main differences between LinkedList and ArrayList are :  
a. ArrayList uses dynamic array where as Linked List uses Doubly Linked List  
b. ArrayList is the best choice if our frequent operation is retrieval while Linked list is the best choice if our frequent operation is insertion or deletion in middle  
  
**Q28 Why Map interface does not extend the Collection interface in Java Collections Framework ?**  
  
One liner answer : **Map interface is not compatible with the Collection interface.**  
Explanation : Since Map requires key as well as value , for example , if we want to add key-value pair then we will use put(Object key , Object value) . So there are two parameters required to add element to the HashMap object  . In Collection interface add(Object o) has only one parameter. 

**Q29 When to use ArrayList and when to use LinkedList in application?**  
  
ArrayList is the best choice if our frequent operation is retrieval operation.  
  
Linked list is the best choice if our frequent operation is insertion or deletion in middle.  
  
**Q30 Write the code for iterating the list in different ways in java ?**  
There are two ways to iterate over the list in java :  
a. using Iterator  
b. using for-each loop  
  
**Q31 How HashSet works internally in java ?**  
  
This is one of the popular interview question. Hash Set internally uses HashMap to maintain the uniqueness of elements.   
**Q32 What is CopyOnWriteArrayList ?  How it is different from  ArrayList in Java?**  
  
[CopyOnWriteArrayList](https://docs.oracle.com/javase/7/docs/api/java/util/concurrent/CopyOnWriteArrayList.html) is a thread safe variant of ArrayList in which all mutative operations like add , set are implemented by creating a fresh copy of the underlying array.  
It guaranteed not to throw ConcurrentModificationException.  
It permits all elements including null. It is introduced in jdk 1.5 .

**Q33  How HashMap works in Java ?**  
  
We are repeating this question , as it is one of the most important question for java developer.HashMap works on the principle of Hashing . please find the detailed answer here [hashmap internal working in java](http://javahungry.blogspot.co.uk/2013/08/hashing-how-hash-map-works-in-java-or.html" \t "_blank) .  
  
**Q34 How remove(key) method works in HashMap ?**  
  
Using the following method we can remove the elements from the hash map.

Object remove (Object key)

**Q36 How TreeMap works in Java ?**  
  
TreeMap internally uses Red-Black tree to sort the elements in natural order. In other words , it sorts the TreeMap object keys using Red-Black tree algorithm.  
  
**Q38 What is the difference between Fail- fast iterator and Fail-safe iterator ?**  
This is one  of the most popular interview question for the higher experienced java developers .  
Main differences between Fail-fast and Fail-safe iterators are :  
a. Fail-fast throw ConcurrentModificationException while Fail-safe does not.  
b. Fail-fast does not clone the original collection list of objects while Fail-safe creates a copy of the original collection list of objects.  
  
**Q39 How ConcurrentHashMap works internally in Java?**  
  
ConcurrentHashMap also works on the principle of hashing  
  
  
**Q42 Explain the importance of hashCode() and equals() method ? Explain the contract also ?**  
HashMap object uses Key object hashCode() method and equals() method to find out the index to put the key-value pair. If we want to get value from the HashMap same both methods are used . Somehow, if both methods are not implemented correctly , it will result in two keys producing the same hashCode() and equals() output. The problem will arise that HashMap will treat both output same instead of different and overwrite the most recent key-value pair with the previous key-value pair.  
Similarly all the collection classes that does not allow the duplicate values use hashCode() and equals() method to find the duplicate elements.So it is very important to implement them correctly.  
  
**Contract of hashCode() and equals() method**  
a.If  object1.equals(object2) , then  object1.hashCode() == object2.hashCode() should always be true.  
  
b. If object1.hashCode() == object2.hashCode() is true does not guarantee object1.equals(object2)  
**Q44 What are concurrentCollectionClasses?**   
In jdk1.5 , Java Api developers had introduced new package called java.util.concurrent that have thread-safe collection classes as they allow collections to be modified while iterating . The iterator is fail-fast that is it will throw ConcurrentModificationException.  
Some examples of concurrentCollectionClasses are :  
a. CopyOnWriteArrayList  
b. ConcurrentHashMap  
  
**Q45 How do you convert a given Collection to SynchronizedCollection ?**  
One line code :    Collections.synchronizedCollection(Collection collectionObj) will convert a given collection to synchronized collection.  
  
**Q46  What is IdentityHashMap ?**  
 **IdentityHashMap**  
[IdentityHashMap](http://docs.oracle.com/javase/7/docs/api/java/util/IdentityHashMap.html) is exactly same as HashMap except the following difference in the case of HashMap JVM uses equals() method to identify the duplicate keys , but in the case of [IdentityHashMap](http://docs.oracle.com/javase/7/docs/api/java/util/IdentityHashMap.html) JVM uses “= =” operator to identify the duplicate keys.

**Q47 What is  WeakHashMap ?**  
 **WeakHashMap :**  
[WeakHashMap](http://docs.oracle.com/javase/7/docs/api/java/util/WeakHashMap.html)  is exactly same as HashMap except the following difference s.

HashMap dominates Garbage Collector where as [WeakHashMap](http://docs.oracle.com/javase/7/docs/api/java/util/WeakHashMap.html) does not dominates the Garbage Collector

**Q48 How will you make Collections readOnly ?**

We can make the Collection readOnly by using the following lines code:

General : Collections.unmodifiableCollection(Collection c)  
Collections.unmodifiableMap(Map m)

Collections.unmodifiableList(List l)

Collections.unmodifiableSet(Set s)

**Q49  What is UnsupportedOperationException?**  
This exception is thrown to indicate that the requested operation is not supported.  
Example of UnsupportedOperationException:  
In other words, if you call add() or remove() method on the readOnly collection . We know readOnly collection can not be modified . Hence , UnsupportedOperationException will be thrown.