**Arrays and Collection Difference**

**Q 1) Difference between the Arrays and Collection**

**Array- Arrays are fixed in size. With respect to memory Arrays are not recommended to use. With respect to performance Arrays are recommended to use. It hold only homogeneous data type elements. There is no underlying data structure and hence readymade methods support is not available.**

**Can hold both primitive & object types.**

**Collection—it is an interface.Collection are growable in nature, we can increase or decrease the size. With respect to memory Collection are recommended to use. With respect to performance Collection are not recommended to use. It Can hold both homogeneous and heterogeneous data type elements. Every collections class is implemented based on some standard data structure. It Can hold only objects but not primitives**

**Q 2) Difference between List and Set interface—**

**In List interface it allows Duplicate value .Insertion order must be preserved. List having classes are** Array List (1.2), Linked List(1.2), Vector (1.0) and 4) Stack (1.0).it allows any number of null values.it has one legacy class vector.

**In Set interface** duplicate elements are not allowed and **Insertion order not preserved.** it allows only one null values.it doesn’t have any legacy class. It implements -- **HashSet , Linked HashSet and Tree set. Set having classes like -HashSet , Linked HashSet and Tree set. And it has two child interface 1) Sorted interface and 2) Navigable interface.**

**Q3) Difference ArrayList and Linked List ?**

**Array List-** Adding and **frequent operation is** retrieval the element ArrayList is best choice. **frequent operation is retrieval. And it is the worst choice if our frequent operation is insertion or deletion, It is implemented as a resizable array.It’s elements can be accessed directly by using the getter & setter Method.** ArrayList is **better for storing and accessing** data.

**Linked List-- It is the Worst choice if our frequent operation is retrieval. And it is the Best choice if our frequent operation is insertion or deletion. It is implemented as a double linked list. It is worse on get and set methods.** LinkedList is **better for manipulating** data.To remove element Linkedlist is best choice.

**Q4) Difference HashSet and Linked Hashset ?**

**HashSet- It implements set interface. It doesn’t maintain insertion order. We can store homogeneous value. It allows null values.**

**Linked Hashset—It extends Hash set class. It maintain insertion order . it doesn’t allows duplicate values. Internally creates LinkHashMap. It implements set, cloneable and Serializable interface.**

**Q4) Difference between Set and Map Interface?**

**Set -Is the child interface of collection.** If we want to represent a group of individual objects as a single entity where duplicates are not allowed and insertion order not preserved then we should go for Set interface. Set Interface having class- HashSet, Linked Hashset and Tree set.

**Map interface** is not the child interface of collection. If we want to represent a group of individual objects **as key value pairs** then we should go for Map interface. Both key and value are objects, keys are not duplicated but values can be duplicated.

Q5) **Difference between ArrayList and Vector?**

ArrayList Non Synchronize, it is not thread safe (at a time multiple threads are allowed to operate an ArrayList Object) and hence relatively performance is high. It is non legacy class because it introduced in 1.2 version.

**Vector** Synchronize, it is thread safe(only one thread is allowed) and hence relatively performance is low. It is legacy class because it introduced in 1.0 version that means old version.

1. **What is difference between array and array-list?**

* Arrays can contain primitive or Objects  whereas ArrayList can contain only Objects.
* Arrays is static with fixed size whereas Array List is not static but dynamic in size or resizable
* Arrays doesn’t provide a lot of features like Array List, such as addAll, removeAll, iterator etc.
* Array take same datatype as a value but ArrayList take Different type of value.
* Array work with primitive datatype but ArrayList works only with object type.

Q 7) D**ifference between TreeSet and HashSet ?**

**HashSet— It implements set interface. It does not maintain insertion order.It allows null values. It is faster than Treeset. We can store homogeneous values.**

**Tree Set – It implements Navigable interface. It maintain and store the elements in natural order.It doesn’t allows null values.It is slower then Hashset. We can’t add only homogeneous elements.**

Q 8) D**ifference between TreeSet and Tree Map ?**

**Tree Map –It implements Map Interface. Duplicated elements are not allowed.Insert objects allowed for same type.**

**Tree Set – It implements Navigable interface. It maintain and store the elements in natural order.It doesn’t allows null values.It is slower then Hashset. We can’t add only homogeneous elements.**

Q9) D**ifference between HashMap and HashTable? [they are both implement Map interface]**

**HashMap—It is a Map based collection class. It is not synchronized. It is faster than Hash table and non Secured. It is Iterator. . It takes one null key and more than one null value.it came from in 1.2 version. It is not a legacy class. It is applicable in single thread application.**

**HashTable—is the child of Dictionary which is abstract class. It came from in 1.0 version that’s why it is a legacy class. It is synchronized and thread safe. It has no null keys and null values. It is slow and secured. It is Enumerator. It is applicable for multithreaded application .**

**10) What is Sorted Interface and Navigable Set?**

**Sorted Set**  -It the child interface of Set. D**uplicate elements are not allowed and all objects should be inserted according to the some sorting order .**.

**Navigable Set-**-It the child interface of Sorted Set .It defines several methods for navigation purpose.

Q11) D**ifference between Comparable interface and Comparator interface?**

**Comparable interface–It is a default natural sorting order present in Java.lang.package.**

**This interface defines only one method CompareTo(). All wrapper classes and String class implement Comparable interface.**

**Comparator interface- It is a customizes sorting order present in Java.util.package.**

**This interface defines two method Compare() and equals(). Collator and Rule base Collator class implement Comparable interface.**

**Q Difference between CopyOnArrayList and ArrayList In java?**

**ArrayList—Iterator of ArrayList is Fail fast and throws ConcurrentModificationException exception once detect any modification in list once iterator begins .It is not Thread safe. ArrayList iterator supports remove() operation.**

**CopyOnArrayList - It is a thread safe version collectione .Iterator of CopyOnArrayList is fail fast and doesn’t throws ConcurrentModificationException. It doesn’t supports remove operation.**

**Q Difference between the Fail Fast and Fail safe?**

**Fail- safe iterators means it will not throw an exception if the collections is modified while iterating over it .**

**Fail Fast iterator means it will throw an exception(ConcurrentModificationException)if the collections is modified while iterating over it. It is nothing but immediately report any failure occurs in the system.**

Q) **Difference between Enumerator(Ist level) Iterator and List Iterator ?**

**Three cursers of Java**

**If we want to retrieve the object one by one from the collection then we should go for curses. They are available in java. 1) Enumeration 2) Iterator 3) List Iterator**

**Enumeration**-this concept is available only for Legacy classes and hence it is not a universal cursor. By using this we can get only read access and we can’t perform remove operation. To overcome the limitations of Enumerator we should go for Iterator**.**

**Iterator- -It is not bidirectional Cursor because by using Iterator we can move only to the forward direction. It is universal cursor because we can apply this concept for any collection objects. Replacement of new objects perform is not possible only read and remove operation is possible**

**List Iterator-It is bidirectional Cursor because by using List Iterator we can move to the forward or backward direction. It is not universal cursor because it is implemented only for List implemented class objects. Replacement of new objects , read and remove operation is possible.**

**Enumeration (1.0)**-it is applicable for Legacy classes (1.0) and it has movement only to the forward direction(Single direction). only read access . it has only two methods 1)hasMoreElements(), nextElement() .

**Iterator(1.2)- -** it is applicable for any collection classes (1.0) and it has movement only to the forward direction(Single direction). Both read and remove access . it has three methods 1) hasNext(), Next() and remove().

**List Iterator-(1.2)** it is applicable for only List classes (1.0) and it has movement both forward and backward direction(bidirection). It has access read, remove, replace and addition of new objects. it has Nine methods 1) hasNext(), Next() and remove().

**List Iterator-It is bidirectional Cursor because by using List Iterator we can move to the forward or backward direction. It is not universal cursor because it is implemented only for List implemented class objects. Replacement of new objects , read and remove operation is possible.**

**Exception handling and Wrapper classes**