1. What is singleton pattern ?

(A)Singleton pattern restricts the instantiation of a class and ensures that only one instance of the class exists in the java virtual machine.

2. What are the rules for immutability?

(A) Declare the class as final so it can’t be extended.

Make all fields private so that direct access is not allowed.

Don’t provide setter methods for variables

Make all **mutable fields final** so that it’s value can be assigned only once.

Initialize all the fields via a constructor performing deep copy.

Perform cloning of objects in the getter methods to return a copy rather than returning the actual object reference.

3.What are the methods of Map.entry?

(A) **equals (Object o)** – It compares the object (invoking object) with the Object   
o for equality.

**K getKey()**– Returns the key for the corresponding map entry.

**V getValue()** – Returns the value for the corresponding map entry.

**int hashcode()** – Returns the hashcode for the corresponding map entry.

**V setValue(V v)** – Sets the value of the map with specified value v.

4.What is the difference between Hashmap and HashTable?

(A) Hashtable is [synchronized](https://stackoverflow.com/questions/1085709/what-does-synchronized-mean), whereas HashMap is not. This makes HashMap better for non-threaded applications, as unsynchronized Objects typically perform better than synchronized ones.

Hashtable does not allow null keys or values. HashMap allows one null key and any number of null values.

One of HashMap's subclasses is LinkedHashMap, so in the event that you'd want predictable iteration order (which is insertion order by default), you could easily swap out the HashMap for a LinkedHashMap. This wouldn't be as easy if you were using Hashtable.

5.What is the logic behind Hashset ?

(A) Hashmap is the logic behind Hashset.

6. What is vector?

(A) The Vector class implements a growable array of objects. Vectors basically fall in legacy classes but now it is fully compatible with collections.

1.Vector implements a dynamic array that means it can grow or shrink as required. 2.Like an array, it contains components that can be accessed using an integer index

3.They are very similar to ArrayList but Vector is synchronised and have some legacy method which collection framework does not contain.

4.It extends **AbstractList** and implements **List** interfaces.

7. What is queue and Dequeue?

(A) Queue follows Last in first order for performing operations.

Element deletion is known as Dequeue.

8. What are the differences between Comparable & Comparator?

(A) (a) Comparable interface can be used to provide single way of sorting whereas Comparator interface is used to provide different ways of sorting.

(b)For using Comparable, Class needs to implement it whereas for using Comparator we don’t need to make any change in the class.

(c)Comparable interface is in java.lang package whereas Comparator interface is present in java.util package.

(d)We don’t need to make any code changes at client side for using Comparable, Arrays.sort() or Collection.sort() methods automatically uses the compareTo() method of the class. For Comparator, client needs to provide the Comparator class to use in compare() method.