1. What is an immutable class?

A: Immutable class is a class which once created, its contents cannot be changed. Immutable

Objects are the objects whose state cannot be changed once constructed. e.g. String class

1. Why String is Immutable or Final in Java

A: String is Immutable in Java because String objects are cached in String pool. Since cached String literal is shared between multiple clients there is always a risk, where one client's action would affect all other client. For example, if one client changes value of String "Test" to "TEST", all other client will also see that value as explained in first example. Since caching of String objects was important from performance reason this risk was avoided by making String class Immutable.

1. What are the types of assertion and what are assertion in java

A: An assertion is a statement in the JavaTM programming language that enables you to test you’re

assumptions about your program. For example, if you write a method that calculates the speed of a

particle, you might assert that the calculated speed is less than the speed of light.

Each assertion contains a Boolean expression that you believe will be true when the assertion

executes. If it is not true, the system will throw an error. By verifying that the Boolean expression is

indeed true, the assertion confirms your assumptions about the behavior of your program,

increasing your confidence that the program is free of errors.

1. What is the default package in java

A: There are three packages are imported by default for each source file. First, the package with no name. Second, the java.lang package. And third, the current package (the package in which the current file is defined).

1. What is Singleton class

A: In [object-oriented programming](http://searchsoa.techtarget.com/definition/object-oriented-programming) , a singleton [class](http://whatis.techtarget.com/definition/class) is a class that can have only one [object](http://searchsoa.techtarget.com/definition/object)(an instance of the class) at a time.

1. Difference between singleton and static class

A: 1. Singleton object stores in Heap but, static object stores in stack

2. We can clone the object of Singleton but, we cannot clone the static class object

3. Singleton class follow the OOP (object oriented principles) but not static class

4. We can implement interface with Singleton class but not with Static class.

7. What is difference between .equals() , (==)

A: Both equals() and "==" operator in Java is used to compare objects to check equality but main difference between equals method and  == operator is that former is method and later is operator. Since [Java doesn’t support operator overloading](http://javarevisited.blogspot.sg/2011/08/why-java-does-not-support-operator.html), == behaves identical for every object but equals() is method, which can be overridden in Java and logic to compare objects can be changed based upon business rules. Another notable difference between == and equals method is that former is used to compare both primitive and objects while later is only used for objects comparison.

1. [CompareTo() vs. equals()](http://stackoverflow.com/questions/1551235/java-strings-compareto-vs-equals)

A: 1. Equals will take any Object as a parameter, but compareTo will only take Strings.

2. Equals only tells you whether they're equal or not, but compareTo gives information on how the Strings compare lexicographically.

9. What is Synchronization?

A: Synchronization in java is the capability of control the access of multiple threads to any shared

resource.

There are two types of synchronization

1. Process Synchronization
2. Thread Synchronization
3. When to use HashTable and HashMap

The basic difference between a Hashtable and an HashMap is that, Hashtable is synchronized while HashMap is not. Thus whenever there is a possibility of multiple threads accessing the same instance, one should use Hashtable. While if not multiple threads are going to access the same instance then use HashMap.

Difference between HashTable and HashMap

* One of the major differences between HashMap and Hashtable is that HashMap is non-synchronized whereas Hashtable is synchronized, which means Hashtable is thread-safe and can be shared between multiple threads but HashMap cannot be shared between multiple threads without proper synchronization. Java 5 introduced ConcurrentHashMap which is an alternative of Hashtable and provides better scalability than Hashtable in Java.Synchronized means only one thread can modify a hash table at one point of time. Basically, it means that any thread before performing an update on a hashtable will have to acquire a lock on the object while others will wait for lock to be released.
* The HashMap class is roughly equivalent to Hashtable, except that it permits nulls. (HashMap allows null values as key and value whereas Hashtable doesn’t allow nulls).
* One more notable difference between Hashtable and HashMap is that because of thread-safety and synchronization Hashtable is much slower than HashMap if used in Single threaded environment. So if you don’t need synchronization and HashMap is only used by one thread, it outperform Hashtable in Java.
* HashMap does not guarantee that the order of the map will remain constant over time

1. Main Difference Between Static And Non Static Methods In Java

A: We can call static methods directly while we cannot call non static methods directly. You need to create and instantiate an object of class for calling non static methods. [VIEW THIS POST](http://software-testing-tutorials-automation.blogspot.in/2014/04/selenium-webdriver-java-tutorials.html) to learn about object In Java.

Non static stuff (methods, variables) cannot be accessible Inside static methods Means we can access only static stuff Inside static methods. Opposite to It, Non static method do not have any such restrictions. We can access static and non-static both kind of stuffs inside non static methods.

Static method is associated with the class while non-static method Is associated with an object.