Core Java: Garbage Collection, I/O Stream, Serialization and Networking

1) What is Garbage Collection?

Garbage collection is a process of reclaiming the runtime unused objects. It is performed for memory management.

2) What is gc()?

gc() is a daemon thread.gc() method is defined in System class that is used to send request to JVM to perform garbage collection.

3) What is the purpose of finalize() method?

finalize() method is invoked just before the object is garbage collected.It is used to perform cleanup processing.

4) Can an unreferenced objects be referenced again?

Yes.

5) What kind of thread is the Garbage collector thread?

Daemon thread.

6) What is difference between final, finally and finalize?

final: final is a keyword, final can be variable, method or class.You, can't change the value of final variable, can't override final method, can't inherit final class.

finally: finally block is used in exception handling. finally block is always executed.

finalize():finalize() method is used in garbage collection.finalize() method is invoked just before the object is garbage collected.The finalize() method can be used to perform any cleanup processing.

7) What is the purpose of the Runtime class?

The purpose of the Runtime class is to provide access to the Java runtime system.

8) How will you invoke any external process in Java?

By Runtime.getRuntime().exec(?) method.

9)What is the difference between the Reader/Writer class hierarchy and the InputStream/OutputStream class hierarchy?

The Reader/Writer class hierarchy is character-oriented, and the InputStream/OutputStream class hierarchy is byte-oriented.

10) What an I/O filter?

An I/O filter is an object that reads from one stream and writes to another, usually altering the data in some way as it is passed from one stream to another.

11) What is serialization?

Serialization is a process of writing the state of an object into a byte stream. It is mainly used to travel object's state on the network.

12) What is Deserialization?

Deserialization is the process of reconstructing the object from the serialized state. It is the reverse operation of serialization.

13) What is transient keyword?

If you define any data member as transient, it will not be serialized.

14) What is Externalizable?

Externalizable interface is used to write the state of an object into a byte stream in compressed format. It is not a marker interface.

15) What is the difference between Serializable and Externalizable interface?

Serializable is a marker interface but Externalizable is not a marker interface. When you use Serializable interface, your class is serialized automatically by default. But you can override writeObject() and readObject() two methods to control more complex object serialization process. When you use Externalizable interface, you have a complete control over your class's serialization process.

16) How do I convert a numeric IP address like 192.18.97.39 into a hostname like java.sun.com?

By InetAddress.getByName("192.18.97.39").getHostName() where 192.18.97.39 is the IP address.

17) What is reflection?

Reflection is the process of examining or modifying the runtime behavior of a class at runtime. It is used in:

- IDE (Integrated Development Environment) e.g. Eclipse, MyEclipse, NetBeans.
- Debugger
- Test Tools etc.

18) Can you access the private method from outside the class?

Yes, by changing the runtime behavior of a class if the class is not secured.

Core Java: Java Collections

19) What is the difference between ArrayList and Vector?

ArrayList	Vector
ArrayList is not synchronized.	Vector is synchronized.
ArrayList is not a legacy class.	Vector is a legacy class.
ArrayList increases its size by 50% of the array size.	Vector increases its size by doubling the array size.

20) What is the difference between List and Set?

List can contain duplicate elements whereas Set contains only unique elements.

21) What is the difference between ArrayList and LinkedList?

ArrayList	LinkedList
ArrayList uses a dynamic array.	LinkedList uses doubly linked list.
ArrayList is not efficient for manipulation because a lot of shifting is required.	LinkedList is efficient for manipulation.
ArrayList is better to store and fetch data.	LinkedList is better to manipulate data.

22) What is the difference between Iterator and ListIterator?

Iterator traverses the elements in forward direction only whereas ListIterator traverses the elements in forward and backward direction.

Iterator	ListIterator
Iterator traverses the elements in forward direction only.	ListIterator traverses the elements in backward and forward directions both.
Iterator can be used in List, Set and Queue.	ListIterator can be used in List only.

23) What is the difference between Iterator and Enumeration?

Iterator	Enumeration
Iterator can traverse legacy and non-legacy elements.	Enumeration can traverse only legacy elements.
Iterator is fail-fast.	Enumeration is not fail-fast.
Iterator is slower than Enumeration.	Enumeration is faster than Iterator.

24) What is the difference between HashSet and TreeSet?

HashSet maintains **no order** whereas TreeSet maintains **ascending order**.

25) What is the difference between Set and Map?

Set contains values only whereas Map contains key and values both.

26) What is the difference between HashSet and HashMap?

HashSet contains only values whereas HashMap contains entry (key, value). HashSet can be iterated but HashMap need to convert into Set to be iterated.

27) What is the difference between HashMap and TreeMap?

HashMap maintains **no order** but TreeMap maintains **ascending order**.

28) What is the difference between HashMap and Hashtable?

HashMap	Hashtable
HashMap is not synchronized.	Hashtable is synchronized.
HashMap can contain one null key and multiple null values.	Hashtable cannot contain any null key or null value.

29) What is the difference between Collection and Collections?

Collection is an interface whereas Collections is a class. Collection interface provides normal functionality of data structure to List, Set and Queue. But, Collections class is to sort and synchronize collection elements.

30) What is the advantage of Properties file?

If you change the value in properties file, you don't need to recompile the java class. So, it makes the application easy to manage.

31) What is the advantage of generic collection?

If we use generic class, we don't need typecasting. It is typesafe and checked at compile time.

32) What is the difference between Comparable and Comparator?

Comparable	Comparator
Comparable provides only one sort of sequence.	Comparator provides multiple sort of sequences.
It provides one method named compareTo().	It provides one method named compare().
It is found in java.lang package.	it is found in java.util package.
If we implement Comparable interface, actual class is modified.	Actual class is not modified.

33) What does the hashCode() method?

The hashCode() method returns a hash code value (an integer number).

The hashCode() method returns the same integer number, if two keys (by calling equals() method) are same.

But, it is possible that two hash code numbers can have different or same keys.

34) Why we override equals() method?

The equals() method is used to check whether two objects are same or not. It needs to be overridden if we want to check the objects based on property.

For example, Employee is a class that has 3 data members: id, name and salary. But, we want to check the equality of employee object on the basis of salary. Then, we need to override the equals() method.

35) What is hash-collision in Hashtable and how it is handled in Java?

Two different keys with the same hash value is known as hash-collision. Two different entries will be kept in a single hash bucket to avoid the collision.

36) What is the Dictionary class?

The Dictionary class provides the capability to store key-value pairs.

37) How to synchronize List, Set and Map elements?

Yes, Collections class provides methods to make List, Set or Map elements as synchronized:

```
public static List synchronizedList(List l){}
public static Set synchronizedSet(Set s){}
public static SortedSet synchronizedSortedSet(SortedSet s){}
public static Map synchronizedMap(Map m){}
public static SortedMap synchronizedSortedMap(SortedMap m){}
```

38) What is the default size of load factor in hashing based collection?

The default size of load factor is **0.75**. The default capacity is computed as initial capacity * load factor. For example, 16 * 0.75 = 12. So, 12 is the default capacity of Map.

Core Java: Java Multithreading

39) What is multithreading?

Multithreading is a process of executing multiple threads simultaneously. Its main advantage is:

- Threads share the same address space.
- Thread is lightweight.
- Cost of communication between process is low.

40) What is thread?

A thread is a lightweight sub process. It is a separate path of execution. It is called separate path of execution because each thread runs in a separate stack frame.

41) Is it possible to start a thread twice?

No, there is no possibility to start a thread twice. If we do, it throws an exception.

42) What is the difference between preemptive scheduling and time slicing?

Under preemptive scheduling, the highest priority task executes until it enters the waiting or dead states or a higher priority task comes into existence. Under time slicing, a task executes for a predefined slice of time and then reenters the pool of ready tasks. The scheduler then determines which task should execute next, based on priority and other factors.

43) What does join() method?

The join() method waits for a thread to die. In other words, it causes the currently running threads to stop executing until the thread it joins with completes its task.

44) What is difference between wait() and sleep() method?

wait()	sleep()
The wait() method is defined in Object class.	The sleep() method is defined in Thread class.
wait() method releases the lock.	The sleep() method doesn't releases the lock.

45) Can we call the run() method instead of start()?

Yes, but it will not work as a thread rather it will work as a normal object so there will not be context-switching between the threads.

46) What about the daemon threads?

The daemon threads are basically the low priority threads that provide the background support to the user threads. It provides services to the user threads.

47) Can we make the user thread as daemon thread if thread is started?

No, if you do so, it will throw IllegalThreadStateException

48) What is shutdown hook?

The shutdown hook is basically a thread i.e. invoked implicitly before JVM shuts down. So we can use it perform clean up resource.

49) When should we interrupt a thread?

We should interrupt a thread if we want to break out the sleep or wait state of a thread.

50) What is synchronization?

Synchronization is the capability of control the access of multiple threads to any shared resource. It is used:

1. To prevent thread interference.
2. To prevent consistency problem.

51) What is the purpose of Synchronized block?

- Synchronized block is used to lock an object for any shared resource.
- Scope of synchronized block is smaller than the method.

52) Can Java object be locked down for exclusive use by a given thread?

Yes. You can lock an object by putting it in a "synchronized" block. The locked object is inaccessible to any thread other than the one that explicitly claimed it.

53) What is static synchronization?

If you make any static method as synchronized, the lock will be on the class not on object.

54) What is the difference between notify() and notifyAll()?

The notify() is used to unblock one waiting thread whereas notifyAll() method is used to unblock all the threads in waiting state.

55) What is deadlock?

Deadlock is a situation when two threads are waiting on each other to release a resource. Each thread waiting for a resource which is held by the other waiting thread.

Core Java: Java Miscellaneous

56) What are wrapper classes?

Wrapper classes are classes that allow primitive types to be accessed as objects.

57) What is a native method?

A native method is a method that is implemented in a language other than Java.

58) What is the purpose of the System class?

The purpose of the System class is to provide access to system resources.

59) What comes to mind when someone mentions a shallow copy in Java?

Object cloning.

60) What is singleton class?

Singleton class means that any given time only one instance of the class is present, in one JVM.

61) Which containers use a border layout as their default layout?

The Window, Frame and Dialog classes use a border layout as their default layout.

62) Which containers use a FlowLayout as their default layout?

The Panel and Applet classes use the FlowLayout as their default layout.

63) What are peerless components?

The peerless components are called light weight components.

64) is the difference between a Scrollbar and a ScrollPane?

A Scrollbar is a Component, but not a Container. A ScrollPane is a Container. A ScrollPane handles its own events and performs its own scrolling.

65) What is a lightweight component?

Lightweight components are the one which doesn't go with the native call to obtain the graphical units. They share their parent component graphical units to render them. For example, Swing components.

66) What is a heavyweight component?

For every paint call, there will be a native call to get the graphical units. For Example, AWT.

67) What is an applet?

An applet is a small java program that runs inside the browser and generates dynamic contents.

68) Can you write a Java class that could be used both as an applet as well as an application?

Yes. Add a main() method to the applet.

69) What is Locale?

A Locale object represents a specific geographical, political, or cultural region.

70) How will you load a specific locale?

By ResourceBundle.getBundle(?) method.

71) What is a JavaBean?

JavaBean are reusable software components written in the Java programming language, designed to be manipulated visually by a software development environment, like JBuilder or VisualAge for Java.

72) Can RMI and Corba based applications interact?

Yes they can. RMI is available with IIOP as the transport protocol instead of JRMP.