30 Java Questions

*Q1. How could Java classes direct program messages to the system console, but error messages, say to a file?

A. The class System has a variable out that represents the standard output, and the variable err that represents the standard error device. By default, they both point at the system console. This how the standard output could be re-directed:
Stream st = new Stream(new FileOutputStream("output.txt")); System.setErr(st); System.setOut(st);

*Q2. What's the difference between an interface and an abstract class?

A. An abstract class may contain code in method bodies, which is not allowed in an interface. With abstract classes, you have to inherit your class from it and Java does not allow multiple inheritance. On the other hand, you can implement multiple interfaces in your class.

*Q3. Why would you use a synchronized block vs. synchronized method?

A. Synchronized blocks place locks for shorter periods than synchronized methods.

*Q4. Explain the usage of the keyword transient?

A. This keyword indicates that the value of this member variable does not have to be serialized with the object. When the class will be de-serialized, this variable will be initialized with a default value of its data type (i.e. zero for integers).

*Q5. How can you force garbage collection?

A. You can't force GC, but could request it by calling System.gc(). JVM does not guarantee that GC will be started immediately.

*Q6. How do you know if an explicit object casting is needed?

A. If you assign a superclass object to a variable of a subclass's data type, you need to do explicit casting. For example:
Object a; Customer b; b = (Customer) a;
When you assign a subclass to a variable having a supeclass type, the casting is performed automatically.

*Q7. What's the difference between the methods sleep() and wait()

A. The code sleep(1000); puts thread aside for exactly one second. The code wait(1000), causes a wait of up to one second. A thread could stop waiting earlier if it receives the notify() or notifyAll() call. The method wait() is defined in the class Object and the method sleep() is defined in the class

Thread.

*Q8. Can you write a Java class that could be used both as an applet as well as an application?

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

*Q9. What's the difference between constructors and other methods?

A. Constructors must have the same name as the class and can not return a value. They are only called once while regular methods could be called many times.

*Q10. Can you call one constructor from another if a class has multiple constructors

A. Yes. Use this() syntax.

*Q11. Explain the usage of Java packages.

A. This is a way to organize files when a project consists of multiple modules. It also helps resolve naming conflicts when different packages have classes with the same names. Packages access level also allows you to protect data from being used by the non-authorized classes.

*Q12. If a class is located in a package, what do you need to change in the OS environment to be able to use it?

A. You need to add a directory or a jar file that contains the package directories to the CLASSPATH environment variable. Let's say a class Employee belongs to a package com.xyz.hr; and is located in the file c:\dev\com\xyz\hr\Employee.java. In this case, you'd need to add c:\dev to the variable CLASSPATH. If this class contains the method main(), you could test it from a command prompt window as follows:

```
c:\>java com.xyz.hr.Employee
```

*Q13. What's the difference between J2SDK 1.5 and J2SDK 5.0?

A.There's no difference, Sun Microsystems just re-branded this version.

*Q14. What would you use to compare two String variables - the operator == or the method equals()?

A. I'd use the method equals() to compare the values of the Strings and the == to check if two variables point at the same instance of a String object.

*Q15. Does it matter in what order catch statements for FileNotFoundException and IOExceptipon are written?

A. Yes, it does. The FileNoFoundException is inherited from the IOException. Exception's subclasses have to be caught first.

*Q16. Can an inner class declared inside of a method access local variables of this method?

A. It's possible if these variables are final.

*Q17. What can go wrong if you replace && with & in the following code:
String a=null; if (a!=null && a.length()>10) {...}
A. A single ampersand here would lead to a NullPointerException.

*Q18. What's the main difference between a Vector and an ArrayList

A. Java Vector class is internally synchronized and ArrayList is not.

*Q19. When should the method invokeLater()be used?

A. This method is used to ensure that Swing components are updated through the event-dispatching thread.

*Q20. How can a subclass call a method or a constructor defined in a superclass?

A. Use the following syntax: super.myMethod(); To call a constructor of the superclass, just write super(); in the first line of the subclass's constructor

For senior-level developers:

**Q21. What's the difference between a queue and a stack?

A. Stacks works by last-in-first-out rule (LIFO), while queues use the FIFO rule

**Q22. You can create an abstract class that contains only abstract methods. On the other hand, you can create an interface that declares the same methods. So can you use abstract classes instead of interfaces?

A. Sometimes. But your class may be a descendent of another class and in this case the interface is your only option.

**Q23. What comes to mind when you hear about a young generation in Java?

A. Garbage collection.

**Q24. What comes to mind when someone mentions a shallow copy in Java?

A. Object cloning.

**Q25. If you're overriding the method equals() of an object, which other method you might also consider?

A. hashCode()

**Q26. You are planning to do an indexed search in a list of objects. Which of the two Java collections should you use:
ArrayList or LinkedList?

A. ArrayList

**Q27. How would you make a copy of an entire Java object with its state?

A. Have this class implement Cloneable interface and call its method clone().
**Q28. How can you minimize the need of garbage collection and make the memory use more effective?

A. Use object pooling and weak object references.
**Q29. There are two classes: A and B. The class B need to inform a class A when some important event has happened. What Java technique would you use to implement it?

A. If these classes are threads I'd consider notify() or notifyAll(). For regular classes you can use the Observer interface.

*Q30. What access level do you need to specify in the class declaration to ensure that only classes from the same directory can access it?

A. You do not need to specify any access level, and Java will use a default package access level.