

Question: What is the difference between an Interface and an Abstract class?

Answer: An abstract class can have instance methods that implement a default behavior. An Interface can only declare constants and instance methods, but cannot implement default behavior and all methods are implicitly abstract. An interface has all public members and no implementation. An abstract class is a class which may have the usual flavors of class members (private, protected, etc.), but has some abstract methods.

---

Question: What is the purpose of garbage collection in Java, and when is it used?

Answer: The purpose of garbage collection is to identify and discard objects that are no longer needed by a program so that their resources can be reclaimed and reused. A Java object is subject to garbage collection when it becomes unreachable to the program in which it is used.

---

Question: Describe synchronization in respect to multithreading.

Answer: With respect to multithreading, synchronization is the capability to control the access of multiple threads to shared resources. Without synchronization, it is possible for one thread to modify a shared variable while another thread is in the process of using or updating same shared variable. This usually leads to significant errors.

---

Question: Explain different way of using thread?

Answer: The thread could be implemented by using runnable interface or by inheriting from the Thread class. The former is more advantageous, 'cause when you are going for multiple inheritance..the only interface can help

---

Question: What are pass by reference and passby value?

Answer: Pass By Reference means the passing the address itself rather than passing the value. Passby Value means passing a copy of the value to be passed.

---

Question: What is the difference between a constructor and a method?

Answer: A constructor is a member function of a class that is used to create objects of that class. It has the same name as the class itself, has no return type, and is invoked using the new operator. A method is an ordinary member function of a class. It has its own name, a return type (which may be void), and is invoked using the dot operator.

---

Question: What is an Iterator?

Answer: Some of the collection classes provide traversal of their contents via a java.util.Iterator interface. This interface allows you to walk through a collection of objects, operating on each object in turn. Remember when using Iterators that they contain a snapshot of the collection at the time the Iterator was obtained; generally it is not advisable to modify the collection itself while traversing an Iterator.

---

Question: State the significance of public, private, protected, default modifiers both singly and in combination and state the effect of package relationships on declared items qualified by these modifiers.

Answer: public : Public class is visible in other packages, field is visible everywhere (class must be public too)  
private : Private variables or methods may be used only by an instance of the same class that declares the variable or method, A private feature may only be accessed by the class that owns the feature. protected : Is available to all classes in the same package and also available to all subclasses of the class that owns the protected feature.This access is provided even to subclasses that reside in a different package from the class that owns the protected feature. default :What you get by default ie, without any access modifier (ie, public private or protected).It means that it is visible to all within a particular package.

---

Question: What is an abstract class?

Answer: Abstract class must be extended/subclassed (to be useful). It serves as a template. A class that is abstract may not be instantiated (ie, you may not call its constructor), abstract class may contain static data. Any class with an abstract method is automatically abstract itself, and must be declared as such. A class may be declared abstract even if it has no abstract methods. This prevents it from being instantiated.

---

Question: What is static in java?

Answer: Static means one per class, not one for each object no matter how many instance of a class might exist. This means that you can use them without creating an instance of a class. Static methods are implicitly final, because overriding is done based on the type of the object, and static methods are attached to a class, not an object. A static method in a superclass can be shadowed by another static method in a subclass, as long as the original method was not declared final. However, you can't override a static method with a nonstatic method. In other words, you can't change a static method into an instance method in a subclass.

Question: What is final?

Answer: A final class can't be extended ie., final class may not be subclassed. A final method can't be overridden when its class is inherited. You can't change value of a final variable (is a constant).

---

Question: What if the main method is declared as private?

Answer: The program compiles properly but at runtime it will give "Main method not public." message.

---

Question: What if the static modifier is removed from the signature of the main method?

Answer: Program compiles. But at runtime throws an error "NoSuchMethodError".

---

Question: What if I write static public void instead of public static void?

Answer: Program compiles and runs properly.

---

Question: What is the first argument of the String array in main method?

Answer: The String array is empty. It does not have any element. This is unlike C/C++ where the first element by default is the program name.

Question: If I do not provide any arguments on the command line, then the String array of Main method will be empty or null?

Answer: It is empty. But not null.

---

Question: How can one prove that the array is not null but empty using one line of code?

Answer: Print args.length. It will print 0. That means it is empty. But if it would have been null then it would have thrown a NullPointerException on attempting to print args.length.

---

Question: Can an application have multiple classes having main method?

Answer: Yes it is possible. While starting the application we mention the class name to be run. The JVM will look for the Main method only in the class whose name you have mentioned. Hence there is not conflict amongst the multiple classes having main method.

---

Question: Can I have multiple main methods in the same class?

Answer: No the program fails to compile. The compiler says that the main method is already defined in the class.

---

Question: What are Checked and UnChecked Exception?

Answer: A checked exception is some subclass of Exception (or Exception itself), excluding class RuntimeException and its subclasses. Making an exception checked forces client programmers to deal with the possibility that the exception will be thrown. eg, IOException thrown by java.io.FileInputStream's read() method. Unchecked exceptions are RuntimeException and any of its subclasses. Class Error and its subclasses also are unchecked. With an unchecked exception, however, the compiler doesn't force client programmers either to catch the exception or declare it in a throws clause. In fact, client programmers may not even know that the exception could be thrown. eg, StringIndexOutOfBoundsException thrown by String's charAt() method. Checked exceptions must be caught at compile time. Runtime exceptions do not need to be. Errors often cannot be.

---

Question: What is Overriding?

Answer: When a class defines a method using the same name, return type, and arguments as a method in its superclass, the method in the class overrides the method in the superclass. When the method is invoked for an object of the class, it is the new definition of the method that is called, and not the method definition from superclass. Methods may be overridden to be more public, not more private.

---

Question: What type of parameter passing does Java support?

Answer: In Java the arguments are always passed by value .

---

Question: Primitive data types are passed by reference or pass by value?

Answer: Primitive data types are passed by value.

---

Question: Objects are passed by value or by reference?

Answer: Java only supports pass by value. With objects, the object reference itself is passed by value and so both the original reference and parameter copy both refer to the same object .

---

Question: What is serialization?

Answer: Serialization is a mechanism by which you can save the state of an object by converting it to a byte stream.

Question: What is the common usage of serialization?

Answer: Whenever an object is to be sent over the network, objects need to be serialized. Moreover if the state of an object is to be saved, objects need to be serilazed.

---

Question: In Java what happens to the static fields of a class during serialization?

Answer: There are three exceptions in which serialization doesnot necessarily read and write to the stream. These are 1. Serialization ignores static fields, because they are not part of ay particular state state. 2. Base class fields are only hendled if the base class itself is serializable. 3. Transient fields.

---

Question: What are wrapper classes?

Answer: Java provides specialized classes corresponding to each of the primitive data types. These are called wrapper classes. They are e.g. Integer, Character, Double etc

---

Question: What is the difference between error and an exception?

Answer: An error is an irrecoverable condition occurring at runtime. Such as OutOfMemory error. These JVM errors and you can not repair them at runtime. While exceptions are conditions that occur because of bad input etc. e.g. FileNotFoundException will be thrown if the specified file does not exist. Or a NullPointerException will take place if you try using a null reference. In most of the cases it is possible to recover from an exception (probably by giving user a feedback for entering proper values etc.).

---

Question: What is BOXING and UNBOXING in C#?

Answer: BOXING in C# is the conversion of a VALUE type on stack to a OBJECT type on the heap. Vice-versa the conversion from an OBJECT type back to a VALUE type is known as UNBOXING and it requires type casting.

Question: In how many ways you can create new copies of an existing string in C#?

Answer: There are two ways: 1. Using overloaded = operator like - string s2 = s1; 2. Using the static Copy method like - string s2 = string.Copy(s1);

---

Question: In how many ways you can compare two strings in C# using overloaded methods and operators?

Answer: There are three ways: 1. Overloaded Compare() method 2. Overloaded Equal() method 3. Overloaded == operator.

---

Question: What is Machine Language ?

Answer: Machine Language refers to the "ones and zeroes" that digital processors use as instructions. Give it one pattern of bits (such as 11001001) and it will add two numbers, give it a different pattern (11001010) and it will instead subtract one from the other. In as little as a billionth of second. The instruction sets within a CPU family are usually compatible, but not between product lines. For example, Intel's x86/Pentium language and Motorola's PPC/Gx language are completely incompatible. Machine Language is painfully difficult to work with, and almost never worth the effort anymore. Instead programmers use the higher-level languages below, which are either compiled or interpreted into machine language by the computer itself.

---

Question: What is Pascal ?

Answer: Pascal [mathematician/philosopher Blaise Pascal] was designed primarily as a tool for teaching good programming skills, but - thanks largely to the availability of Borland's inexpensive Pascal compiler for the early IBM PC - it has become popular outside of the classroom. Unlike many languages, Pascal requires a fairly structured approach, which prevents the kinds of indecipherable "spaghetti code" and easily-overlooked mistakes that plague programmers using languages such as Fortran or C. Free and commercial tools are available from various sources for DOS, Windows, Mac, OS/2, AmigaOS, and Unix-like systems. The web site editor BBEdit is written in Pascal.

---

Question: What is C ?

Answer: C [successor to the language "B"] offers an elegant compromise between the efficiency of coding in assembly language and the convenience and portability of writing in a structured, high-level language. By keeping many of its commands and syntax analagous to those of common machine languages, and with several generations of optimising compilers behind it, C makes it easy to write fast code without necessarily sacrificing readability. But it still tempts you write code that only a machine can follow, which can be a problem when it comes time to debug it or make changes. Free and commercial tools (most of which now also support C++) are available from various sources for just about every operating system.

---

Question: What is C# ?

Answer: C# ["C++" with the plus signs overlapping, pronounced "C sharp", equivalent to D] is actually Microsoft's answer to Java. They originally tried to release "Java" development tools that would produce apps that weren't truly portable; you could only use them on Windows. But this violated their licensing agreement with Sun (creators of Java), who successfully put a stop to that. So Microsoft turned around and produced a language with similar features that effectively is tied to Windows. Although they are submitting the language to a standard-setting body, for all practical purposes it's just a proprietary variant of C++ whose specs they'll dictate, available only from Microsoft, and practical only for Windows.

---

Question: What is Delphi ?

Answer: Delphi [home of the Greek oracle Pythia] is a non-standard, object-oriented version of Pascal developed by Borland for their rapid application development tool of the same name. The Delphi environment was designed to compete with Microsoft's Visual Basic tools, freeing the programmer from having to write all the code for the user interface by letting her drag and drop objects and attach functions to various buttons and other on-screen elements. Its ability to manipulate databases is another strength. Commercial tools are available from Borland for Windows and Linux.

---

Question: What is Oberon ?

Answer: Oberon [a moon of Uranus] is a later sibling of Pascal and Modula (sharing a parent, creator Niklaus Wirth). The name also refers to the operating environment in which the language was designed to be used.

---

Question: What is Basic ?

Answer: BASIC ["Beginner's All-purpose Symbolic Instruction Code"] is the first language that most early microcomputer users learned. The BASIC interpreters on those machines weren't very sophisticated or fast, largely due to the memory and speed limitations of the hardware, and the language encouraged sloppy coding. As an unknown pundit put it: "BASIC is to computer languages what Roman numerals are to arithmetic" Modern versions of BASIC are more structured and often include compilers for greater speed. Free and commercial tools are available from various sources for DOS, Windows, Mac, and Unix-like systems.

---

Question: What is Visual Basic ?

Answer: Visual Basic [a version of "BASIC" for graphical environments] is Microsoft's Jack of all Trades language. It's a cross between BASIC, the various macro languages of Microsoft Office, and some rapid application development tools. The idea was to get people started writing macros using VBA (Visual Basic for Applications), then sell them the whole VB programming tool when they run into the limits of that approach. Unfortunately VB apps are impossible to port to other environments, and you're at the mercy of Microsoft's changing specs for the language. (Programs written in VB6 or earlier will not run properly in VB.NET!) Available only from Microsoft, only for Windows.

Question: What is Euphoria ?

Answer: Euphoria ["a state of bliss"] is described by its creator as "simple, flexible, and easy-to-learn", and its supportive community of users agree. Although powerful, it's not prone to allowing "dangerous" mistakes. It's an interpreted language, but quicker than most, and it can be converted into C code, which can then be compiled using an optimising compiler. The interpreter and C converter (for both Windows/DOS and Linux/FreeBSD) are available free, or with some bonus features when registered for a modest fee.

---

Question: What is ada ?

Answer: Ada [proto-programmer Lady Ada Byron] Based largely on Pascal, it was commissioned by the U.S. Dept. of Defense to create a standard language to replace the polyglot they had amassed over the decades. It's commonly accused of being typical of government committee work, but has many strengths, including its error-handling and the ease of maintaining and modifying programs. Furthermore, government studies :) indicate that it's more cost-effective than C++, Pascal, or Fortran. The current version includes object-oriented features.

---

Question: What is Small Talk ?

Answer: Smalltalk ["easy conversation"] The object-oriented, graphical Smalltalk development environment is what inspired Steve Jobs and later Bill Gates to "invent" the Mac OS and Windows interfaces. Focusing on the superficial aspects of it (windows and mice) they missed the real gem: the language.

---

Question: What is Assembly Language ?

Answer: Assembly Language is as close as you can come to writing in machine language, but has the advantage that it's also human-readable... using a small vocabulary of words with one syllable. Each written instruction (such as MOV A,B) typically corresponds to a single machine-language instruction (such as 11001001). An assembler makes the translation before the program is executed. Back when CPU speed was measured in kiloHertz and storage space was measured in kiloBytes, Assembly was the most cost-efficient way to implement a program. It's used less often now (with all those kilo's replaced by mega's or giga's, and even tera's on the horizon, it seems no one cares anymore about efficiency), but if you need speed and/or compactness above all else, Assembly is the solution.

---

Question: What is C++ ?

Answer: C++ ["C" with the C instruction to "increment"] is probably the most widely-supported language today, and most commercial software is written in C++. The name reflects why: when it was introduced it took all the benefits of the then-reigning development language (C) and incrementally added the next set of features programmers were looking for (object oriented programming). So programmers didn't have to throw anything out and re-do it... but they could add those techniques to their repertoire as needed. OO purists hate the results, but it's difficult to argue with that success. Free and commercial tools are available from various sources for just about every operating system. (Objective-C is an alternate approach to adding OO characteristics to C (borrowing directly from SmallTalk), which hasn't attracted as large a community of users.)

Question: What is Java ?

Answer: Java [slang for "coffee"] is kind of a streamlined version of C++, designed for portability. Its key advantage is that Java programs can be run on any operating system for which a Java "virtual environment" is available. (Programs in most other languages have to be modified and recompiled to go from one OS to another.) The language is defined by Sun and widely licenced to other companies, making it possible to run Java apps in web browsers, portable phones, desktop computers, web servers, etc. It isn't as fast as applications written in a compiled language like C++, however. Free and commercial tools are available from various sources for most current operating systems. Although Microsoft is removing support for Java from the default setup of new versions of Windows, it can easily be added back in.

---

Question: What is Icon ?

Answer: Icon [perhaps from "iconoclastic", pre-dating the current use of the term in graphical interfaces] is a high-level, imperative, procedural language. Text-handling is one of its strengths. Storage is allocated dynamically. Full versions exist for Unix-like systems, Windows, and Java environments; older versions exist for pre-OS-X Macs and other operating systems.