1. What is object-oriented programming? Is Java an object-oriented language?

Ans: Object-oriented programming works based on the concept of objects and the objects are the containers which store the data in the form of fields and code in the form of procedures which implements the logic for performing the operations. Java utilizes eight primitive data types such as boolean, byte, char, int, float, long, short, double and these data types are not the objects, hence java cannot be considered as a 100% object-oriented language.

2. What are wrapper classes in Java?

Ans: Wrapper classes convert the Java primitives into the reference types which are the objects. Every primitive data type has a class dedicated to it. These are known as wrapper classes because they “wrap” the primitive data type into an object of that class.

3. Explain the difference between ‘finally’ and ‘finalize’ in Java?

Ans:

Finally:

It is used along with a try-catch block, the “finally” block ensures that a particular piece of code is always executed, even if the execution is thrown by the try-catch block.

Finalize:

It is a special method in the object class. The finalize() method is generally overridden to release system resources when garbage value is collected from the object.

4. What are the main uses of “this” keyword?

Ans: There are many uses of this keyword.

The keyword “this” can be used for referring to the current class instance variable.

The keyword “this” can be used to invoke the current class method (implicitly).

The method “this()” can be used to invoke the current class constructor.

The keyword “this” can be passed as an argument in the method call.

The keyword “this” can be passed as an argument in the constructor call.

The keyword “this” can be used to return the current class instance from the method.

5. Why is method overloading not possible by changing the return type in java?

Ans: It is not possible by changing the return type of the program due to avoid the ambiguity. The following example explains the ambiguity.

6. Can you use this() and super() both in a constructor?

Ans: No. It’s because

7. What are the differences between method Overloading and Overriding?

Ans:

**Method Overloading:**

It increases the readability of the program.

It occurs within a class.

In this scenario, the parameters have to be different.

**Method Overriding:**

It provides the specific implementation of the method that is already provided by its superclass.

It occurs in two classes that have IS-A relationship between them.

8. How to achieve encapsulation in Java? Give an example.

Ans: Apply these two steps for achieving encapsulation in java.

Declare the variable of the class as private.

Provide public set() and get() methods to modify the values of variables.

“this()” and “super()” has to be the first statement in the class constructor.

9. What is super in java?

Ans: The “super” keyword in Java is a reference variable that is used to refer to the immediate parent class object. Whenever an instance of the subclass is created, an instance of the parent class is created implicitly which is then referred to by the super reference variable. The “super()” method is called in the class constructor implicitly by the compiler if there is no super or this.

10. Why are Java Strings immutable in nature?

Ans: In Java, string objects are immutable which means once if the String object is created then its state cannot be changed. Java creates a new string object whenever you try to update the value of that object instead of updating the values of that particular object. String objects are generally cached in the String pool and hence Java String objects are immutable. Since String literals are usually shared between multiple clients, action from one client might affect the rest. It enhances security, caching, synchronization, and performance of the application.

11. Can we change the scope of the overridden method in the subclass?

Ans: Yes. It is possible to change the scope of the overridden method in the subclass. However, it should be noticed that the accessibility of the method cannot be decreased. The following measures must be taken care of while changing the accessibility of the method.

The private can be changed to protected, public, or default.

The protected can be changed to public or default.

The default can be changed to public.

The public will always remain public.

12. What is the difference between abstract classes and interfaces?

Ans:

Abstract class:

An abstract class provides complete, default code and the details that must be overridden.

In an abstract class, a class can extend only one abstract class.

An abstract class can have non-abstract methods and instance variables.

The visibility of an abstract class can be private, public or protected.

The abstract class performs fast and it can use the constructors.

Interface:

An interface doesn’t provide any code but only the signature.

A class can implements several interfaces.

All methods of an Interface are abstract and it doesn’t contain instance variables.

The visibility of an interface can be either public or none.

The interfaces will perform slow and it cannot use the constructors.

13. What is synchronization?

Ans:

Synchronization refers to multi-threading. A synchronized block of code is executed only by one thread at a time.

Java supports execution of multiple threads, so two or more threads can access the same fields or objects.

Synchronization is a process which keeps all concurrent threads in execution to be in sync.

Synchronization avoids the memory consistency errors which are caused due to the inconsistent view of shared memory.

When a method is declared as synchronized the thread holds the monitor for that method’s object. If another thread is executing the synchronized method the thread is blocked until that thread releases the monitor.

14. Define ClassLoader?

Ans. A ClassLoader is a component of Java Virtual Machine (JVM) that helps in loading class files when a program is executed. Moreover, it is the first to load a workable file in Java language.

15. What is JDK, JRE, JVM?

Ans. While downloading the Java file, we get the following two important things:-

JDK -

It refers to the Java Development Kit (JDK).

This is a dedicated kit only useful for developing software.

Further, it is Platform Dependent as the compiler generates the byte code.

The JDK package includes many tools useful for debugging and Developing.

JRE -

It refers to Java Runtime Environment (JRE) in Java.

JRE is a group of software and libraries designed for executing Java Programs.

This is similar to JDK in platform dependency.

The package supports various files and libraries for a runtime environment in Java language.

JVM (Java Virtual Machine) includes a Just-in-Time (JIT) compiler tool that turns all the source code in Java into the low-level machine language (ML). It is very compatible to make it run faster than the constant application.

JRE or Java Runtime Environment includes class libraries and other JVM supporting files. But it doesn’t include any Java development tool like a compiler or debugger.

JDK or Java Development Kit includes some tools necessary to write Java Programs. Also, it uses JRE to execute these programs. Further, JRE contains a compiler, an applet viewer, and a Java app launcher

16. Name the various Memory Allocations available in the Java language.

Ans. The following are the different types of memory allocations in Java.

Class

Stack

Heap

Native Method Stack

Program Counter (PC) Memory.

Stack memory is useful to store the order of method execution and local variables. It is created at the time of the creation of threads. On the other hand, Heap memory is used to store the objects as dynamic memory for allocating the objects & JRE classes in the runtime.

17. What is meant by Java String Pool?

Ans. A Java String Pool is a group of strings within Java's Heap memory. Whenever you try to build a new string object, then the JVM checks the presence of the object within the pool. If it is available, the same object reference is shared with the variable, otherwise it will create a new object.

18. What is Java?

Java is the high-level, object-oriented, robust, secure programming language, platform-independent, high performance, Multithreaded, and portable programming language. It was developed by James Gosling in June 1991.

19.What is the static block?

Static block is used to initialize the static data member. It is executed before the main method, at the time of classloading.

20.Can you tell the difference between equals() method and equality operator (==) in Java?

equals()

It is a binary operator in Java.

This method is used for checking the equality of

contents between two objects as per the specified business logic.

==

This is a method defined in the Object class.

This operator is used for comparing addresses (or references), i.e checks if both the objects are pointing to the same memory location.

21.

What is Garbage Collection(GC)

Garbage Collection is process of reclaiming the runtime unused memory automatically. In other words, it is a way to destroy the unused objects.