**1 What is JVM and explain me the Java memory allocation**

JVM is a java virtual machine which is the runtime engine to run java applications. It interprets the bytecode for the computers processor to perform the java program instructions. JVM memory has two memory allocations called stack and heap. Heap stores the actual object and the reference to the object is stored in stack

**2. What is Polymorphism and encapsulation?**

Polymorphism is the ability of an object to perform a single method in different ways.

Dynamic Polymorphism —The process where the decision to execute a method is determined during run time. An example of this is method overriding.

Static Polymorphism — The process where the decision to execute a method is determined during compile time. An example of this is method overloading.

**3. What is method overloading and Method over riding?**

Overloading – It occurs when two or more methods in one class have the same method name but different parameters.

Overriding - having two methods with the same method name and parameters (i.e., method signature). One of the methods is in the parent class and the other is in the child class. Overriding allows a child class to provide a specific implementation of a method that is already provided its parent class.

**4. Why string is Immutable?**

If a string is mutable, changing the string with one reference will lead to the wrong value for the other references. It causes security threat so string is immutable.

**5. What is the difference between String and String buffer?**

String – It is Immutable. It overrides the equals() method and uses equals() method to compare the contents of the string.

StringBuffer – It is mutable. StringBuffer does not overrides the equals() method

**6. What is the difference between array and array list?**

Array is static (fixed length data structure) where as ArrayList is dynamic (Resizable). Array can contain both primitives and objects where as ArrayList can contain only objects.

**7. What is the difference between hash map and Hash table?**

HashMap allows one Null Key and multiple Null values but HashTable doesn’t allow any Null values or keys. HashMap is not synchronized, it is not Thread-safe. HashTable is synchronized and Thread-safe.

**8. What is a vector in Java?**

Vector is an implementation of List. Vector is a legacy class. It is synchronized and Thread-safe. It has both Iterator and enumeration for looping.

**9. What is set in java?**

Set is a collection and does not contains duplicate values, no ordering is guaranteed. It has three types of implemtations such as hashSet, Linked HashSet, TreeSet.

**10. What is an abstract class**?

Abstract classes are classes that contain one or more abstract methods. An abstract method is a method that is declared, but contains no implementation. Abstract classes may not be instantiated, and require subclasses to provide implementations for the abstract methods.

**11. What is an interface?**

An interface is similar to class. Writing an interface is similar to writing a class. But a class describes the attributes and behaviors of an object. And an interface contains behaviors that a class implements. A class implements an interface, and inherits the abstract methods of the interface

**12.  Why Java is Platform independent?**

We can say that Java is platform independent because the java interpretor also known as java virtual machine ,converts the java class file into java byte code . This java byte code is machine independent,i.e,it does not depends on which machine it is run. The java compiler converts this java byte code into respective machine code according to the machine it is run on. Since program in java neither depends  on software or hardware of the machine it is running, hence it is called platform independent.

**13. What are access modifiers? Give me an example?**

The access modifiers in java specifies accessibility (scope) of a data member, method, constructor or class.

There are 4 types of java access modifiers:

1. Private - The private access modifier is accessible only within class.
2. Default - If you don't use any modifier, it is treated as **default** bydefault. The default modifier is accessible only within package.
3. Protected - The **protected access modifier** is accessible within package and outside the package but through inheritance only.

The protected access modifier can be applied on the data member, method and constructor. It can't be applied on the class.

1. Public - The **public access modifier** is accessible everywhere. It has the widest scope among all other modifiers.

**14. What are java exceptions? Give me an example**

An Exception is an unwanted event that interrupts the normal flow of the program. When an exception occurs program execution gets terminated. By handling the exceptions (try/catch block, throws) we can provide a meaningful message to the user .

Example: while using FileReader class in your program to read data from a file, if the file specified in its constructor doesn't exist, then a FileNotFoundException occurs.

**15. What is the difference between throws and throwable?**

throws is also a keyword in java which is used in the method signature to indicate that this method may throw mentioned exceptions. The caller to such methods must handle the mentioned exceptions either using try-catch blocks or using throws keyword.

Throwable is a super class for all types of errors and exceptions in java. This class is a member of java.langpackage

**16. What is the difference between Error and exception?**

Exceptions in java are divided into two categories – checked and unchecked. Where as all Errors belongs to only one category i.e unchecked. Recovering from Error is not possible. The only solution to errors is to terminate the execution. Recovering from Exception is possible by using either try-catch blocks or throwing exception back to caller.

17. What is the difference between Error, throwable and exception?

**18. What are collection APIs, give me an example**

Collection is a framework or container of set of classes and interfaces to implement the collection data structure. Collection is s top level interface in java collection framework. Collections is the utility class which has utility methods to operate the collection object.

Example: List, Set

**19. What is the difference between final and finally?**

Final is a keyword.

Final is used to apply restrictions on class, method and variable. Final class can't be inherited, final method can't be overridden and final variable value can't be changed.

Finally is a block.

Finally is used to place important code, it will be executed whether exception is handled or not.

**20. Will java supports multiple inheritance?**

No, Java does not supports multiple inheritance. It causes ambiguity.

a class cannot extend more than one class. However, a class can implement one or more interfaces, which has helped Java get rid of the impossibility of multiple inheritance.

**21.  What are the different types of interface? (Ans List, set, Queue)**

**22. What are wrapper class? Give me an example**

A Wrapper class is a class where we can wrap a primitive value into a wrapper class object.

Example: Autoboxing, Auto unboxing

**23. What is boxing and unboxing in Java? Explain with an example**

Automatic conversion of primitive types to the object of their corresponding wrapper classes is known as autoboxing. Eg: conversion of int to Integer, long to Long, double to Double

**24. Explain for each loop**

For-each is another array traversing technique like for loop, while loop, do-while loop

* It starts with the keyword **for** like a normal for-loop.
* Instead of declaring and initializing a loop counter variable, you declare a variable that is the same type as the base type of the array, followed by a colon, which is then followed by the array name.
* In the loop body, you can use the loop variable you created rather than using an indexed array element.
* It’s commonly used to iterate over an array or a Collections class (eg, ArrayList)

**25. What are iterators, explain with an example**

Iterator is used for iterating (looping) various collection classes such as [HashMap](https://beginnersbook.com/2013/12/hashmap-in-java-with-example/), [ArrayList](https://beginnersbook.com/2013/12/java-arraylist/" \o "ArrayList in java with example programs – Collections Framework" \t "_blank), [LinkedList](https://beginnersbook.com/2013/12/linkedlist-in-java-with-example/) etc.

**26. How do you access Private variables in different class ?**

Using getters and setters methods

**What is static and Dynamic Polymorphism ?**

Dynamic Polymorphism —The process where the decision to execute a method is determined during run time. An example of this is method overriding.

Static Polymorphism — The process where the decision to execute a method is determined during compile time. An example of this is method overloading.

**What is multithreading in Java?**

Multithreading in java is a process of executing multiple threads simultaneously.

**Serializatio**n is the process of converting the objects into sequence of bytes which can be written into file or db. A class must implement serializable interface to serialize its objects

**Generics** - Using Java Generic concept, we might write a generic method for sorting an array of objects, then invoke the generic method with Integer arrays, Double arrays, String arrays and so on, to sort the array elements.

1. **Dead lock in Java ?**

**Deadlock** describes a situation where two or more threads are blocked forever, waiting for each other

Solution: change the order of the lock(oredering the lock in a fixed sequence manner)

1. **What’s the difference between abstract class and interface?**

**Type of methods** : Interface can have only abstract methods. Abstract class can have abstract and non-abstract methods

**Type of variables** : Abstract class can have final, non-final, static and non-static variables. Interface has only static and final variables.

**Implementation** : Abstract class can provide the implementation of interface. Interface can’t provide the implementation of abstract class.

A Java interface can be implemented using keyword “implements” and abstract class can be extended using keyword “extends”.

1. **What is a constructor?**

A **constructor**is a special method that is used to to initialize the objects at the time of object creation

1. It has the **same name** as the class
2. It should not return a value not even ***void***
3. **What’s the difference between constructor and a method?**

The constructor is used to initialize an object whereas method is used to exhibits functionality of an object.

Constructors are invoked implicitly whereas methods are invoked explicitly.

The constructor does not return any value where the method may/may not return a value.

In case constructor is not present, a default constructor is provided by java compiler. In case of the method, no default method is provided.

A constructor should be of the same name as that of class. Method name should not be of the same name as that of class.

1. **What’s the difference between assert and verify?**

When an “assert” command fails, the test execution will be aborted. So when the Assertion fails, all the test steps after that line of code are skipped. The solution to overcoming this issue is to use a try-catch block

When a “verify” command fails, the test will continue executing and logging the failure. Mostly, the Verify command is used to check non-critical things. In such cases where we move forward even though the end result of the check value is failed.