CODERS LODGE

JAVA PROGRAMMING

VIVA QUESTIONS

Kindly read the instructions carefully

1. **All these questions are important for examination and interview point of view, so practice them well.**
2. **If you have any doubt or facing any problem regarding these questions you can mail us at coderslodgeofficial@gmail.com or drop a message in our WhatsApp or telegram group.**
3. **If you want to support us, give your valuable feedback so that next time we can improve while interacting with you.**
4. ***Reminder*-Practice all questions well it will build your concept clear and you can easily score good in your exams.**

Connect with us

- If you want to join our **WhatsApp** community then mail us at: - coderslodgeofficial@gmail.com
- Join our **Telegram group**: - https://t.me/coderslodgeofficial
- Like our **Facebook page**: - https://www.facebook.com/coderslodge
- Follow us on **Instagram**:- https://www.instagram.com/coderslodge/
- Follow us on **Twitter**: - https://twitter.com/CodersLodge
- Follow us on **LinkedIn**:- https://www.linkedin.com/company/coderslodge

1. Why is Java a platform independent language?

Java language was developed in such a way that it does not depend on any hardware or software due to the fact that the compiler compiles the code and then converts it to platform-independent byte code which can be run on multiple systems.

The only condition to run that byte code is for the machine to have a runtime environment (JRE) installed in it.

2. Briefly explain the concept of constructor overloading.

Constructor overloading is the process of creating multiple constructors in the class consisting of the same name with a difference in the constructor parameters. Depending upon the number of parameters and their corresponding types, distinguishing of the different types of constructors is done by the compiler.

3. Differentiate between method overloading and method overriding.

No.	Method Overloading	Method Overriding
1)	Method overloading is used *to increase the readability* of the program.	Method overriding is used *to provide the specific implementation* of the method that is already provided by its super class.
2)	Method overloading is performed *within class.*	Method overriding occurs *in two classes* that have IS-A (inheritance) relationship.

3)	In case of method overloading, *parameter must be different*.	In case of method overriding, *parameter must be same*.
4)	Method overloading is the example of *compile time polymorphism*.	Method overriding is the example of *run time polymorphism*.
5)	In java, method overloading can't be performed by changing return type of the method only. *Return type can be same or different* in method overloading. But you must have to change the parameter.	*Return type must be same or covariant* in method overriding.

4. What part of memory - Stack or Heap - is cleaned in garbage collection process?

Heap

5. What are the differences between JVM, JRE and JDK in Java?

Criteria	JDK	JRE	JVM
Abbreviation	Java Development Kit	Java Runtime Environment	Java Virtual Machine
Definition	JDK is a complete software development kit for developing Java	JRE is a software package providing Java class libraries, JVM and all the	JVM is a platform-dependent, abstract machine comprising of 3 specifications - document describing the

Criteria	JDK	JRE	JVM
	applications. It comprises JRE, JavaDoc, compiler, debuggers, etc.	required components to run the Java applications.	JVM implementation requirements, computer program meeting the JVM requirements and instance object for executing the Java byte code and provide the runtime environment for execution.
Main Purpose	JDK is mainly used for code development and execution.	JRE is mainly used for environment creation to execute the code.	JVM provides specifications for all the implementations to JRE.
Tools provided	JDK provides tools like compiler, debuggers, etc for code development	JRE provides libraries and classes required by JVM to run the program.	JVM does not include any tools, but instead, it provides the specification for implementation.
Summary	JDK = (JRE) + Development tools	JRE = (JVM) + Libraries to	JVM = Runtime environment to

Criteria	JDK	JRE	JVM
		execute the application	execute Java byte code.

6. What is the difference between an object and an instance?

An Object May not have a class definition. eg int a[] where a is an array.An Instance should have a class definition. eg MyClass my=new MyClass();my is an instance.

7. What is meant by the term OOPs?

OOPs refers to Object-Oriented Programming. It is the programming paradigm that is defined using objects. Objects can be considered as real-world instances of entities like class, that have some characteristics and behaviors.

8. What is Compile time Polymorphism and how is it different from Runtime Polymorphism?

Compile Time Polymorphism: Compile time polymorphism, also known as Static Polymorphism, refers to the type of Polymorphism that happens at compile time. What it means is that the compiler decides what shape or value has to be taken by the entity in the picture.

Runtime Polymorphism: Runtime polymorphism, also known as Dynamic Polymorphism, refers to the type of Polymorphism that happens at the run time. What it means is it can't be decided by the compiler. Therefore what shape or value has to be taken depends upon the execution. Hence the name Runtime Polymorphism.

9. What is Abstraction?

If you are a user, and you have a problem statement, you don't want to know how the components of the software work, or how it's made. You only want to know how the software solves your problem. Abstraction is the method of hiding unnecessary details from the necessary ones. It is one of the main features of OOPs.
For example, consider a car. You only need to know how to run a car, and not how the wires are connected inside it. This is obtained using Abstraction

10. What is meant by Inheritance?

The term “inheritance” means “receiving some quality or behavior from a parent to an offspring.” In object-oriented programming, inheritance is the mechanism by which an object or class (referred to as a child) is created using the definition of another object or class (referred to as a parent). Inheritance not only helps to keep the implementation simpler but also helps to facilitate code reuse

11. Is it mandatory for a catch block to be followed after a try block?

No, it is not necessary for a catch block to be present after a try block. - A try block should be followed either by a catch block or by a finally block. If the exceptions likelihood is more, then they should be declared using the throws clause of the method.

12. Differentiate between string and stringbuffer.

No.	String	StringBuffer
1)	The String class is immutable.	The StringBuffer class is mutable.

2)	String is slow and consumes more memory when we concatenate too many strings because every time it creates new instance.	StringBuffer is fast and consumes less memory when we concatenate t strings.
3)	String class overrides the equals() method of Object class. So you can compare the contents of two strings by equals() method.	StringBuffer class doesn't override the equals() method of Object class.
4)	String class is slower while performing concatenation operation.	StringBuffer class is faster while performing concatenation operation.
5)	String class uses String constant pool.	StringBuffer uses Heap memory

13. What is the static variable?

The static variable is used to refer to the common property of all objects (that is not unique for each object), e.g., The company name of employees, college name of students, etc. Static variable gets memory only once in the class area at the time of class loading. Using a static variable makes your program more memory efficient (it saves memory). Static variable belongs to the class rather than the object.

14. Can we override the static methods?

No, we can't override static methods.

15. What are access modifiers in Java?

In Java, access modifiers are special keywords which are used to restrict the access of a class, constructor, data member and method in another class. Java supports four types of access modifiers:

a. Default
b. Private
c. Protected
d. Public

16. What is encapsulation in Java?

Encapsulation is a mechanism where you bind your data(variables) and code(methods) together as a single unit. Here, the data is hidden from the outer world and can be accessed only via current class methods. This helps in protecting the data from any unnecessary modification. We can achieve encapsulation in Java by:

Declaring the variables of a class as private.

Providing public setter and getter methods to modify and view the values of the variables.

17. What is JDBC Driver?

JDBC Driver is a software component that enables java application to interact with the database. There are 4 types of JDBC drivers:

JDBC-ODBC bridge driver

Native-API driver (partially java driver)

Network Protocol driver (fully java driver)

Thin driver (fully java driver)

18. What are the steps to connect to a database in java?

- Registering the driver class
- Creating connection
- Creating statement

- Executing queries
- Closing connection

19. What do you understand by JDBC Statements?

JDBC supports 3 types of statements:

Statement: Used for general purpose access to the database and executes a static SQL query at runtime.

PreparedStatement: Used to provide input parameters to the query during execution.

CallableStatement: Used to access the database stored procedures and helps in accepting runtime parameters.

20.What is Serialization?

Java provides mechanism called serialization to persists java objects in a form of ordered or sequence of bytes that includes the object’s data as well as information about the object’s type and the types of data stored in the object.So if we need to serialize any object then it can be read and deserialize it using object’s type and other information so we can retrieve original object.Classes ObjectInputStream and ObjectOutputStream are high-level streams that contain the methods for serializing and deserializing an object.

21.What is need of Serialization?

Serialization is usually used when there is need to send your data over network or to store in files. By data I mean objects and not text.

22. Can you Serialize static variables?

No,you can't.As you know static variable are at class level not at object level and you serialize a object so you can't serialize static variables.

23. What is wrapper class?

The wrapper class in Java provides the mechanism to convert primitive into object and object into primitive.

24. What is inner class?

Java inner class or nested class is a class that is declared inside the class or interface.

We use inner classes to logically group classes and interfaces in one place to be more readable and maintainable.

Additionally, it can access all the members of the outer class, including private data members and methods.

25. What is package?

A java package is a group of similar types of classes, interfaces and sub-packages.

Package in java can be categorized in two form, built-in package and user-defined package.

There are many built-in packages such as java, lang, awt, javax, swing, net, io, util, sql etc.

26. What is use of super keyword in java?

The super keyword in Java is a reference variable which is used to refer immediate parent class object.

Whenever you create the instance of subclass, an instance of parent class is created implicitly which is referred by super reference variable.

27. What is interface in java?

An interface in Java is a blueprint of a class. It has static constants and abstract methods.The interface in Java is a mechanism to achieve abstraction

There can be only abstract methods in the Java interface, not method body. It is used to achieve abstraction and multiple inheritance in java.

In other words, you can say that interfaces can have abstract methods and variables. It cannot have a method body.Java Interface also represents the IS-A relationship.

28. What is the use of this keyword in java?

The this keyword refers to the current object in a method or constructor.

29. Briefly tell about multithreading?

- Multithreading in Java is a process of executing multiple threads simultaneously.
- A thread is a lightweight sub-process, the smallest unit of processing. Multiprocessing and multithreading, both are used to achieve multitasking.
- However, we use multithreading than multiprocessing because threads use a shared memory area. They don't allocate

separate memory area so saves memory, and context-switching between the threads takes less time than process.

- Java Multithreading is mostly used in games, animation, etc.

30. What are advantages of multithreading?

1) It doesn't block the user because threads are independent and you can perform multiple operations at the same time.

2) You can perform many operations together, so it saves time.

3) Threads are independent, so it doesn't affect other threads if an exception occurs in a single thread.

31. What is thread synchronization?

Thread synchronization is the concurrent execution of two or more threads that share critical resources. Threads should be synchronized to avoid critical resource use conflicts. Otherwise, conflicts may arise when parallel-running threads attempt to modify a common variable at the same time.

32. What is difference between Container and Component ?

Main difference between Container and Component is that former can hold other components e.g. JFrame which is used as container to hold other components e.g. JButton. This is rather a simple Swing question and mostly asked in telephonic or upto 2 years experienced programmers.

33. What is difference between AWT and Swing?

No.	Java AWT	Java Swing
1)	AWT components are **platform-dependent**.	Java swing components are **platform-independent**.
2)	AWT components are **heavyweight**.	Swing components are **lightweight**.
3)	AWT **doesn't support pluggable look and feel**.	Swing **supports pluggable look and feel**.
4)	AWT provides **less components** than Swing.	Swing provides **more powerful components** such as tables, lists, scrollpanes, colorchooser, tabbedpane etc.
5)	AWT **doesn't follows MVC**(Model View Controller) where model represents data, view represents presentation and controller acts as an interface between model and view.	Swing **follows MVC**.

34. What is JFC?

The Java Foundation Classes (JFC) are a set of GUI components which simplify the development of desktop applications.

35. What is an Applet?

Applet is a special type of program that is embedded in the webpage to generate the dynamic content. It runs inside the browser and works at client side.

36. Briefly explain about the term GUI.

GUI stands for Graphical User Interface, a term used not only in Java but in all programming languages that support the development of GUIs. A program's graphical user interface presents an easy-to-use visual display to the user. It is made up of graphical components (e.g., buttons, labels, windows) through which the user can interact with the page or application.

37. What is an event?

Changing the state of an object is known as an event. For example, click on button, dragging mouse etc.

38. Explain about Applet Life Cycle.

The applet life cycle can be defined as the process of how the object is created, started, stopped, and destroyed during the entire execution of its application. It basically has five core methods namely init(), start(), stop(), paint() and destroy().These methods are invoked by the browser to execute.

39. What is an exception?

Exception is an abnormal condition which occurs during the execution of a program and disrupts normal flow of the program. This exception must be handled properly. If it is not handled, program will be terminated abruptly.

40. How the exceptions are handled in Java? OR Explain exception handling mechanism in Java?

Exceptions in Java are handled using try, catch and finally blocks.

try block : The code or set of statements which are to be monitored for exception are kept in this block.

catch block : This block catches the exceptions occurred in the try block.

finally block : This block is always executed whether exception is occurred in the try block or not and occurred exception is caught in the catch block or not.