An ISO 9001:2015 Certified Company
NARESH i technologies

Top Core Java Interview Questions

Home / Blog, Interview Questions / Top Core Java Interview Questions

Core Java Interview Questions

1. Can we write static block within the method?

Answer:

No. Compiler will give error.

```
public class Test{
public static void main(String[] args) {
static{
}
}
}
```

Compiletime Error: Illegal start of expression.

2. Can we write return statement without value in constructors?

Answer: Yes.

```
public class Test{
    Test(){
        return;
    }
    public static void main(String[] args) {
        {
            System.out.println("non static block");
        }
    }
}
```

3.What is the difference between static and new keyword?

Answer:

New keyword is useful for providing memory for non-static data, it is individual, where as static keyword is useful for providing memory static data by default by JVM, it is common for all.

4.Is bellow code valid?

Answer:

```
public class Test{

public static void main(String[] args) {

final int a = 100;

a = 200;

}

}
```

No.

Once we initialize the final variable, we can’t reinitialize the variable again either with same value or different value.

5.What is the use of constructor?

Answer:

Constructor is useful for initialize the non-static variables meanwhile of object creation.

6.What is the use of preparer?

Answer:

Preparer is useful for providing default values for static variables in static loading phase.

7.What is the use of initializer?

Answer:

Initializer is useful for replacing default values with original values of static variables in static initialization phase.

8.What is the use of copy constructor?

Answer:

Mean while of creating object, if we want to copy the data from existed object to new creating object within the different memory location then we should use copy constructor.

9.Why should we go for constructor overloading?

Answer:

If we want perform different operations in different object creations mode then we should prefer constructor overloading.

10.What is use of this () in java?

Answer:

This () method is useful for making communication between one constructor to another constructor within the same class.

11.What is use of super () in java?

Answer:

Super () method is useful for making communication between one constructor to another constructor between super and sub classes.

12.What is singleton?

Answer:

A class which is allows to programmer to create only one object.

13.What is use of non-static block?

Answer:

To make our logic/functionalities to common for all objects, mean while of object creation then we should use non-static block.

14.What is the difference between constructor and non-static block?

Answer:

To make our logic/functionalities to common for all we should use non-static block, whereas constructor is useful for providing different functionalities for different object.

15.Can we this and super keywords in static context?

Answer:

No, we are unable to use, the reason are by default non-static final reference variables.

16.What is use of static methods?

Answer:

If we are unable to create object for a class, but still if we want to execute some logic and if we are not giving to subclasses to change our logic then we should prefer static methods.

17.Can we write return statement without value in void method?

Answer: Yes.

18.What the difference between void and non-void method?

Answer: The method which is unable to carry information from called area to calling area is called void method the method which is able to carry the information from called area to calling area is called non-void method.

19.What is factory method?

Answer: A method which is returns either same class memory other class memory is called factory method.

20.What is factory class?

Answer: A class which is ready to returns memory for other classes is called factory class.

21.What is the use of abstract methods and concrete methods?

Answer: To providing rules or specifications to third party vendors we should use abstract methods, where as if we want provides implementations for specifications we should prefer concrete methods.

22.What is the use of default methods in java?

Answer: Without doing modifications to implementation classes if we want write logic to interface then we should use default methods. These are introduced in java 8.

23.How many interfaces do we have from java 1.7?

Answer:

- We have three interfaces from java 1.7.
- Normal interface/non-functional interface
- Functional interfaces
- Marker interfaces.

24.Can we write default methods in functional interface?

Answer:

Yes, we can write multiple default methods and java.lang.Object class methods also (abstract format only).

25.Can we override static, final, private methods?

Answer: No

26.What are required things to call methods?

Answer:

The following things are very much mandatory for calling any method. Those are

- Access modifier
- Modifier (static/non-static)
- Return type
- Method name
- Parameters(number of, type of , place of parameters)
- Throws keyword(Exceptions)

27.What is encapsulation?

Answer: The process of providing security for our data, if the end user is authorized then we should provides permissions to security data is called encapsuaiton.

28.What is marker interface and it usage?

Answer: The interface which contains zero abstract method is called marker interface, which is useful for making our object as specialized object to participated in special operations like serialization, cloning and etc…

29.How can we call abstract class constructor?

Answer: By creating object for its sub class.

30.Are interface talks about state of an object?

Answer: No. The reason is by default interface variables are public static final.

31.Is java supports multiple inheritance?

Answer: Yes. Java supports multiple inheritance only through interfaces but not through classes.

32.What is method overloading?

Answer: The process of writing same method multiple times with different parameters is called method overloading.

33.What is method overriding?

Answer: The logic which is given by the java software or already existed, if that logic is not suitable for project requirement then we should write same method with different logic within the super and sub classes is called method overriding.

34.How many memories are given by the JVM meanwhile of subclass object creation?

Answer: JVM will provide only one memory for subclass but super class non-static data will come stored in sub class memory.

35.What is upcasting and its usage?

Answer: The process of placing sub class memory into super class reference variable is called upcasting.

If we are not aware about type of value and memory but still if we want hold then we should prefer upcasting.

36.What is downcasting and its usage?

Answer: The process of placing super class reference into sub class reference by using cast operator, but super class memory should contains sub class memory. With the support of upcasting reference variable we can able to communicating with super class data but not sub class data, if we want to communicating with sub class data then we should prefer downcasting.

37.What is the output of bellow code?

```
public class Test{
void m1(float x){
System.out.println(“m1-float: “+x);
}
void m1(int x){
System.out.println(“m1-int: “+x);
}
public static void main(String[] args) {
Test t = new Test();
t.m1(100);
}
}
```

Output: m1-int: 100

38.What is output of bellow code?

Answer:

```
public class Test{
void m1(Object x){
System.out.println(“m1-Object: “+x);
}
void m1(String x){
System.out.println(“m1-String: “+x);
}
public static void main(String[] args) {
Test t = new Test();
t.m1(“ram”);
}
}
```

Output: m1-Sstring: ram

39.What is static and dynamic polymorphism?

Answer:

- The process of executing the method by JVM at runtime, which is binds by the compiler at compilation time, is called dynamic polymorphism.
- The process of executing the method from subclass by JVM at runtime, which is not binds by the compiler at compilation time, is called dynamic polymorphism.

40.What is immutable object and examples?

Answer: The process of unable to change the state of the object is called immutable object.

Ex: String and Wrapper classes.

41.What is difference between StringBuffer and StringBuilder?

Answer: StringBuffer is thread-safe (synchronized), multiple threads are unable to processing the StringBuffer object at time, so it will gives less performance, where as StringBuilder is not a thread-safe so multiple objects can processing the object, which gives more performance.

42.How convert StringBuffer object to String object?

Answer: StringBuffer object we can convert into String object using toString().

43.What is difference between the following code?

Answer: String s = “ram”;

String s1 = new String(“ram”);

In the first statement, JVM only creates one object in String constant pool.

In the second statement, JVM will create two objects, one is in heap area and another one is in String constant pool area.

44.What is use of intern() in String class?

Answer: The intern () will go to memory of a String object and read the data later it will pointing to same data in string constant pool area.

45.What type of data holding by method area and heap area?

Answer: Method are always hold static type data where as heap area always holds objects, in that object non-static data will store.

46.Are String and wrapper classes participated in inheritance?

Answer: No, the reason is final classes are not participated in inheritance.

47.What are the rules followed to create java bean?

Answer: The java standard edition class which maintain the following rules.

- Class should implements either java.io.Serializable or Externalizable
- Variables should be private
- Public zero or default constructor.
- Public setters and getters
- Class should be public.

48.What is wrapper class?

Answer: The class which is representing the data in the form object is called wrapper class.

49.What is auto boxing and unboxing?

Answer: The process of converting primitive data to object type is called auto boxing where as converting object type data to primitive is called auto unboxing.

50.What is package?

Answer: The process of encapsulating the similar type of classes, interfaces and enum in a single unit is called package.

51.What is use of static import statement?

Answer: Static import statement is useful for loading only static type data.

52.How to compile and execute package program?

Answer:

Compilation: javac –d path filename.java

Execution: java packagename.classname

53.What is the use of throw keyword?

Answer: Throw keyword is useful for forward the exception object from one area to another area.

54.What is the use of throws keyword?

Answer: throws keyword is useful forward the checked exception from our program to JVM and giving information to compiler there may be chance of raising the exception in our program.

55.How many ways can able print exception message?

Answer: There are three ways to print exception messages.

- printStackTrace()
- toString()
- getMessage()

56.What is the difference between initCause(-) and getCause() method?

Answer: InitCause(-) will provides root cause or more meaningful information of the exception whereas getCause() will read exception information which added in the initCause(-) method.

57.What is different between byte streams and character streams?

Answer: Byte streams are sending and writing at a time one byte of data whereas character streams are sending and writing two byte of data at time. Byte streams are use full for working image related information whereas character streams are useful for working with textual file.

58.What is serialization?

Answer: The process of sending data from java application to file in the form of entire object is called serialization.

59.What is externalization?

Answer: The process of sending the data from java application to file in the form of some part of object is called externalization.

60.What is internal functionalities of JVM meanwhile of object creation?

Answer: JVM mainly follows five steps meanwhile of exception.

- Jvm checks the problem.
- Jvm will select predefine exception class
- Jvm will create object for selected exception class
- Jvm will add reason of the exception if required
- Finally that exception object handover to catch block if not existed then handover to console.

61.Why can't we write Object type parameter in catch block?

Answer:

- Catch block always executing meanwhile of exception raise in the try block.
- Try block always throwing exception objects, which are directly or indirectly subclass of Throwable class.
- So catch block parameter type required both Object class functionalities and Throwable class functionalities.

If we are writing

catch(Object o){

}

- Parameter 'o' only have Object class functionalities but does not Throwable functionalities.
- That's why we can't write Object type parameter in catch block.

62.Can we stop of finally block execution?

Answer: Yes**.** We can by writing the following statement.

Syatem.exit(0)

63.What is cloning?

Answer: The process of creating new object with the support of existed object data is called cloning.

64.What is use of hashCode(), equals(-) of Object class?

Answer: To get memory identification number which is given by the jvm without checking we should use hashCode(-). To check whether objects are same or not by using memory identification which is given by the JVM then we should use equals(-).

65.What are the difference between comparator and comparable interfaces?

Answer: Comparable interface to check whether the homogenous object are same or not where as comparator will check whether the heterogeneous objects are same or not.

Comparable having only one method like compareTo(Object).
Comparator having two method like compare(Object,Object) and equals(Object)

66.What is collection?

Answer: It is one well define mechanism, to group homogeneous, heterogeneous, duplicate and unique objects into a single container is called collection.

67.What is the difference between Vector and ArrayList?

Answer: By default vector is synchronized class, so its object never allows to multiple threads to processing, whereas ArrayList will allows to multiple thread to processing its object. The load factor of Vector is current capacity*2, whereas ArrayList load factor is current capacity * 3/2.

68.What is generic collection?

Answer: The process of providing type safety on collection object is called generic.

69.What is fail-fast and fail-safe?

Answer: Whenever we reading the data from ArrayList by using one thread, if we are doing modification on ArrayList by using another thread we will get exception that is fail-fast, to overcome this problem we have concurrent collection, that means meanwhile of reading the data from ArrayList by using one thread object, we can able to do modifications by using another thread object on same object is called fail-safe.

70.What are functionalities of TreeSet?

Answer: TreeSet not allows heterogenous, null objects, it will provide sorting technique.

71.What is the difference between new and newInstance()?

Answer: Meanwhile of creating a class if we are aware about for which class we are going to creating object then we should use new keyword, if we want to create an object for a class which is loading at runtime by using byte code then we should use newInstance().

72.Can we write compile and execute the program without class?

Answer: Yes. From java 1.5 onwards by using enum and from java 1.8 onwards by using interface we can able to compile and execute the program.

73.Why is String immutable?

Answer: While working with Hashtable, if our String object state changes then we should faces the problem like null or NullPointerExceptions, to avoiding this problem Java community people make String as immutable and also to maintain proper memory management in string constant pool area.

74.What is lambda expression?

Answer: It is one anonymous expression or function to provide implementation for functional interface in a concise and clear manner and within the less time.

With the help of lambda we can develop functional programming in java.

75.What is functional programming?

Answer: The process of sending functional as method argument is called functional programming.

76.Can we use peek() and pop() on empty stack and what is the difference between them?

Answer: Pop() will give top most element with deletion from Stack whereas peek() will give top most element without deletion from Stack.

Both the methods will throw exception like java.util.EmptyStackException if elements are not existed.

```
import java.util.Stack;
class A{
public static void main(String[] ss){
Stack s = new Stack();
//System.out.println(s.pop());
System.out.println(s.peek());
}
}
```

78.What is the advantage and disadvantage of ArrayList?

Answer: The advantage of ArrayList is we can able to read any index element within the same time. The drawback is whenever we insert elements in the middle of ArrayList the next position elements are going to shifted to right side, it will reduce performance.

79.What is synchronization in java?

Answer: The process of allowing only one thread at a time to execute entire work is called synchronization.

80.What is race condition?

Answer: The process of allowing multiple threads at a time to execute the task simultaneously is called race condition, it will give unreliable result.

81.What is dead lock?

Answer: Placing the thread into wait mode up to life time is called deadlock.

82.What is loosely coupling?

Answer: The process of holding multiple subclass type of objects with the single reference variable is called loosely coupling, but at a time reference variable can hold only one sub class object.

Read More: Java Interview Questions

Share this post