**Tricky questions java**

**1.Why java is object Oriented?**

Java is an object-oriented programming language. Everything in Java is an object. Object-oriented means we organize our software as a combination of different types of objects that incorporate both data and behaviour.

**2.why java is platform independent?**

We write a code once and run anywhere languages. The platform is software or hardware environment as program runs.

**3.Features of java?**

1. Simple
2. Object oriented
3. Platform independent
4. Secured
5. Robust

**4. public static void main(String[] args)- meaning**

* **class** keyword is used to declare a class in Java.
* **public** keyword is an access modifier that represents visibility. It means it is visible to all.
* **static** is a keyword. If we declare any method as static, it is known as the static method. The core advantage of the static method is that there is no need to create an object to invoke the static method. The main() method is executed by the JVM, so it doesn't require creating an object to invoke the main() method. So, it saves memory.
* **void** is the return type of the method. It means it doesn't return any value.
* **main** represents the starting point of the program.
* **String[] args or String args[]** is used for command line argument.
* **System.out.println()** is used to print statement. Here, System is a class, out is an object of the PrintStream class, println() is a method of the PrintStream class.

**5. Can you have multiple classes in a java source file?**

Yes.

**6.What happens at run time?**

While in runtime, the java virtual machine loads the . class file in memory and executes that class to generate the output of program.

**7. Can you save a Java source file by another name than the class name?**

Yes, if the class is not public.

**8.Difference b/w JDK, JRE, JVM?**

**9.local variable?**

A variable declared inside the body of the method is called local variable. You can use this variable only within that method and the other methods in the class aren't even aware that the variable exists.

. A local variable cannot be defined with "static" keyword.

**10.instance variable?**

A variable declared inside the class but outside the body of the method, is called an instance variable. It is not declared as static.

**11.Static variable?**

A variable that is declared as static is called a static variable. It cannot be local. You can create a single copy of the static variable and share it among all the instances of the class.

**12. what is keyword?**

These are predefined words by Java so they cannot be used as a variable or object name or class name.

**13.java control statements or control flow in java?**

**Three types:**

**1.decision making statements**

* **If statement**
* **Switch statement**

**2.looping statements**

* **For loop**
* **For each loop**
* **While loop**
* **Do while loop**

**3.jumping statements**

* **Break statements**
* **Continue statements**

**14.** **Decision-Making statements?**

decision-making statements decide which statement to execute and when.

**15. Nested if-statement?**

In nested if-statements, the if statement can contain a if or if-else statement inside another if or else-if statement.

**16.switch statement?**

In Java, Switch statements are similar to if-else-if statements. The switch statement contains multiple blocks of code called cases and a single case is executed based on the variable which is being switched.

**17. what is class?**

Collection of objects is called class.

**18.what is polymorphism?**

If one task is performed in different ways, it is known as polymorphism.

Ex: shapes, square, rectangle

**19. what is coupling?**

Coupling refers to the knowledge or information or dependency of another class. It arises when classes are aware of each other.

**20.private constructor?**

🡪It does not allow to create an object outside the class.

🡪If a class has a private constructor and when we try to extend the class, a compile-time error occurs.

🡪We cannot access a private constructor from any other class.

**21.Difference b/w Constructor and Method**

|  |  |
| --- | --- |
| A constructor is used to initialize the state of an object | Method is used to expose the behaviour of an object |
| It does not have any return type. | Method must have a return type |
| Constructor must invoked implicitly | Method must invoked explicitly |
| CONSTRUCTOR has same name as in the class name | Method has may or may not same as a class name. |

**22. Why is the Java main method static?**

It is because the object is not required to call a static method. If it were a non-static method, JVM creates an object first then call main() method that will lead the problem of extra memory allocation.

**23. Can we execute a program without main() method?**

No, one of the ways was the static block, but it was possible till JDK 1.6. Since JDK 1.7, it is not possible to execute a Java class without the main method.

**24. Why Method Overloading is not possible by changing the return type of method only?**

In java, method overloading is not possible by changing the return type of the method only because of ambiguity.

**25. Can we overload java main() method?**

Yes, by method overloading. You can have any number of main methods in a class by method overloading. But JVM calls main() method which receives string array as arguments only.

**26.instance initializer block?**

Instance Initializer block is used to initialize the instance data member. It run each time when object of the class is created.

**27. Is final method inherited?**

Yes, final method is inherited but you cannot override it.

**28. What is blank or uninitialized final variable?**

A final variable that is not initialized at the time of declaration is known as blank final variable.

**29. Can we initialize blank final variable?**

Yes, but only in constructor.

**30. What is final parameter?**

If you declare any parameter as final, you cannot change the value of it.

**31. Can we declare a constructor final?**

No, because constructor is never inherited.

**32.Upcasting?**

If the reference variable of Parent class refers to the object of Child class, it is known as upcasting.

**33.what is binding?**

Connecting a method call to the method body is known as binding.

**34.two types of binding?**

* Static binding(early binding)
* Dynamic binding(late binding)

**35.static binding?**

When type of the object is determined at compiled time (by the compiler), it is known as static binding.

**36.dynamic binding?**

When type of the object is determined at run-time, it is known as dynamic binding.

**37.java instanceOf?**

The instanceof in java is also known as type *comparison operator* because it compares the instance with type. It returns either true or false.

Ex:

1. **class** Simple1{
2. **public** **static** **void** main(String args[]){
3. Simple1 s=**new** Simple1();
4. System.out.println(s **instanceof** Simple1);//true
5. }
6. }

**38.Difference b/w abstract class and interface?**

|  |  |
| --- | --- |
| Abstract | Interface |
| 1) Abstract class can have abstract and non-abstract methods. | Interface can have only abstract methods. Since Java 8, it can have default and static methods also |
| Abstract class doesn't support multiple inheritance | Interface supports multiple inheritance |
| Abstract class can have final, non-final, static and non-static variables | Interface has only static and final variables. |
| An abstract class can be extended using keyword "extends" | An interface can be implemented using keyword "implements" |

**39.Advantages of encapsulation?**

* Read only or write-only. In other words, you can skip the getter or setter methods.
* control over the data
* data hiding

**40.Adavntages of arrays?**

* **Code Optimization**: It makes the code optimized, we can retrieve or sort the data efficiently.
* **Random access**: We can get any data located at an index position

**41.Object cloning in java?**

The object cloning is a way to create exact copy of an object. The clone() method of Object class is used to clone an object.

1. Student18 s1=**new** Student18(101,"amit");
3. Student18 s2=(Student18)s1.clone();
5. System.out.println(s1.rollno+" "+s1.name);
6. System.out.println(s2.rollno+" "+s2.name);

**42. Why use clone() method?**

The clone() method saves the extra processing task for creating the exact copy of an object. If we perform it by using the new keyword, it will take a lot of processing time to be performed that is why we use object cloning.

**43.adavntages of clone?**

* You don't need to write lengthy and repetitive codes.
* It is the easiest and most efficient way for copying objects.
* Clone() is the fastest way to copy array.

**44.java math class?**

Java Math class provides several methods to work on math calculations like min(), max(), avg(), sin(), cos(), tan(), round(), ceil(), floor(), abs() etc.

**45.Difference b/w array and arrayList?**

|  |  |
| --- | --- |
| Array | ArrayList |
| Size should be given at the time of array declaration.  String[] name = new String[2] | Size may not be required. It changes the size dynamically.  ArrayList name = new ArrayList |
| To put an object into array we need to specify the index.  name[1] = “book” | No index required.  name.add(“book”) |

**46.wrapper class?**

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

**47. Java Strictfp Keyword?**

Java strictfp keyword ensures that you will get the same result on every platform if you perform operations in the floating-point variable.

**48. Difference between object and class**

|  |  |
| --- | --- |
| Object | Class |
| Object is instance of class | Class is a blueprint of an object |
| It is a physical entity | Logical entity |
| Object is created many times as per requirement. | Class is declared once. |
| Object allocates memory when it is created | Class doesn't allocated memory when it is created. |

**49. Difference between method overloading and method overriding in java?**

|  |  |
| --- | --- |
| Method overloading is performed within class. | Method overriding occurs in two classes that have IS-A (inheritance) relationship |
| Method overloading is the example of compile time polymorphism. | Method overriding is the example of run time polymorphism. |

**50. How to create a string object?**

There are two ways to create String object:

1. By string literal

2. By new keyword

**51. Difference between String and String Buffer?**

|  |  |
| --- | --- |
| **String buffer** | **String** |
| mutable | immutable |
| stringBuffer is fast | Strig is slow |
| StringBuffer doesn’t override the equals() method in object class. | String override the equals() method in object class. |

**52. How to create Immutable class?**

There are many immutable classes like String, Boolean, Byte, Short, Integer, Long, Float, Double etc. In short, all the wrapper classes and String class is immutable.

**53. What is Exception in Java?**

an exception is an event that disrupts the normal flow of the program. It is an object which is thrown at runtime.

**54. Java Multi-catch block?**

A try block can be followed by one or more catch blocks. Each catch block must contain a different exception handler.

**55. Java Nested try block?**

In Java, using a try block inside another try block is permitted. It is called as nested try block.

**56. Why use nested try block?**

Sometimes a situation may arise where a part of a block may cause one error and the entire block itself may cause another error. In such cases, exception handlers have to be nested.

**57. Why use Java finally block?**

finally block in Java can be used to put "cleanup" code such as closing a file, closing connection, etc. o

The important statements to be printed can be placed in the finally block.

**58.difference b/w throw and throws?**

|  |  |
| --- | --- |
| **Throw** | **Throws** |
| Throw followed by instance | Throws followed by class |
| Throw is used within the method. | Throws used within the method signature |

**59.what is multi threading?**

It will executing the multiple threads simultaneously.

**60.java garbage collection?**

**Garbage:**

In java, garbage means unreferenced objects.

**Garbage collection:**

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

**61.advantage of garbage collection?**

* Memory efficient
* Automatically done.

**62.what is collection?**

Collection is a single unit of object.

**63.what is framework in java?**

o It provides readymade architecture.

o It represents a set of classes and interfaces.

o It is optional.

**64.difference b/w arraylist and linkedlist?**

|  |  |
| --- | --- |
| ArrayList | LinkedList |
| Dynamic array to store the elements | Doubly linked list to store the elements |
| Act as a list | Act as a list and queue |
| Better for retrieving a elements | Better for inserting and deleting an elements |

**65.what is list?**

List in java to maintain the ordered collection.

**66.what is collections class?**

Collections class is used exclusively with static methods that operate on or return collections. Its inherited object class.

67. **Why Java is not 100% Object-oriented?**  
Java is not 100% Object-oriented because it makes use of eight primitive data types such as boolean, byte, char, int, float, double, long, short which are not objects.

**68.two types of constructor?**

* Default constructor

🡪does not take any inputs

* Parameterized constructor

🡪capable of initializing the instance variable and provide the values

**69.difference b/w hashset and treeset?**

|  |  |
| --- | --- |
| Hashset- doesnot maintain insertion order | Ascending order |
| DOESNOT allow duplicates | DOESNOT allow duplicates |
| All null elements | Does not allow null elements |
| Faster | slow |

**70.synchronized non access modifier?**

Synchronized is used to indicate that a method accessing by only one thread at a time.

**71. Variables used in a switch statement can be used with which datatypes?**

String, byte , short, int, enum , char.

**72. When parseInt() method can be used?**

This method is used to get the primitive data type of a certain String.

**73. Which package is used for pattern matching with regular expressions?**

java.util.regex package is used for this purpose.

**74. java.util.regex consists of which classes?**

java.util.regex consists of three classes − Pattern class, Matcher class and PatternSyntaxException class.

**75. Why Packages are used?**

Packages are used in Java in-order to prevent naming conflicts, to control access, to make searching/locating and usage of classes, interfaces, enumerations and annotations, etc., easier.

**76. What are the two ways in which Thread can be created?**

Thread can be created by: implementing Runnable interface, extending the Thread class.

**77. Define immutable object?**

An immutable object can’t be changed once it is created.

**78. What is JAR file?**

JAR files is Java Archive files and it aggregates many files into one. It holds Java classes in a library. JAR files are built on ZIP file format and have .jar file extension.

**79. What is the purpose of File class?**

It is used to create objects that provide access to the files and directories of a local file system.

80. What is the difference between the Reader/Writer class hierarchy and the InputStream/OutputStream class hierarchy?

The Reader/Writer class hierarchy is character-oriented, and the InputStream/OutputStream class hierarchy is byte-oriented.

**81.difference b/w c++ and java?**

|  |  |
| --- | --- |
| C++ platform independent | Java is platform independent |
| Fully supports pointers | There is no concept of pointers in java |
| Allow multiple inheritance | Does not allow multiple inheritance |

#### 82. Which class is a superclass of all classes?

Java.lang.The object is the root class for all the java classes and we don’t need to extend it.

**83.  What is the use of System class in Java?**

System class doesn’t provide any public constructors, so we can’t instantiate this class and that’s why all of its methods are static.

**84. What is an instanceof keyword?**

We can use instanceof keyword in java to check whether an object belongs to a class or not.

**85. What is an Iterator?**

Iterator interface provides methods to iterate over any collection. We can get iterator instance from a collection using iterator() method.\

86. **which collections are thread safe?**

* HashTable
* Stack
* Vector

#### 87. What happens when an exception is thrown by the main method?

When an exception is thrown by the main() method, Java Runtime terminates the program and print the exception message and stack trace in system console.

#### 88. Does the constructor return any value?

#### Yes, The constructor can implicitly return the current instance of the class and we cannot use explicit return type with the constructor.

#### 89. What is the usage of a default constructor?

#### The main use of the default constructor such as to assign the default value to the objects. The java compiler can create a default constructor in an implicit manner if the constructor does not hold any values in it.

#### 90.  How does Java enable high performance?

#### Java uses Just In Time compiler to enable high performance. It is used to convert the instructions into bytecodes.

#### 91. **What are the OOPs concepts?**

* Inheritance
* Encapsulation
* Polymorphism
* Abstraction
* Interface

#### 92. **Collections are used to perform the following operations?**

* Searching
* Sorting
* Manipulation
* Insertion
* Deletion

#### 93. **What are all the Classes and Interfaces that are available in the collections?**

**Interfaces:**

* Collection
* List
* Set
* Map
* Sorted Set
* Sorted Map
* Queue

**Classes:**

* Lists:
* Array List
* Vector
* Linked List

#### Set,map,queue

#### 94. **What is meant by Ordered and Sorted in collections?**

**Ordered:**It means the values that are stored in a collection is based on the values that are added to the collection. So we can iterate the values from the collection in a specific order.

**Sorted:**Sorting mechanisms can be applied internally or externally so that the group of objects sorted in a particular collection is based on the properties of the objects.

**95.vector?**

#### Maintain the insertion order

#### Allow duplicates

#### Same as arraylist

#### Methods are synchronized

#### Thread safe

#### 96.what is set?

#### Set cares about uniqueness.

#### 97. **What are the different ways to handle exceptions?**

#### Try..catch

#### declaring Throws keyword

#### 98. **What are the advantages of Exception handling?**

* The normal flow of the execution won’t be terminated if an exception gets handled
* We can identify the problem by using catch declaration

#### 99. **How do you make a thread in Java?**

**a) Extend Thread class:**Extending a Thread class and override the run method. The thread is available in java.lang.thread.

**Example:**

|  |
| --- |
| Public **class** Addition **extends** Thread {  **public** **void** run () {  }  } |

**b) Implement Runnable interface:**Another way is by implementing the runnable interface. For that, we should provide the implementation for the run () method which is defined in the interface.

**Example:**

|  |
| --- |
| Public **class** Addition **implements** Runnable {  **public** **void** run () {  }  } |

**100.** **Explain about join () method.**

**Answer:** Join () method is used to join one thread with the end of the currently running thread.

**Example:**

|  |
| --- |
| **public** **static** **void** main (String[] args){  Thread t = **new** Thread ();  t.start ();  t.join ();  } |

#### 101.difference b/w notify() and notifyall()

| **notify()** | **notifyAll()** |
| --- | --- |
|  |  |
| This method is used to send a signal to wake up a single thread in the waiting pool. | This method sends the signal to wake up all the threads in a waiting spool. |

#### 102. **How to stop a thread in java?**

**We can stop a thread by using the following thread methods:**

* Sleeping
* Waiting
* Blocked

**103.** **When to use the Runnable interface Vs Thread class in Java?**

**Answer:** If we need our class to extend some other classes other than the thread then we can go with the runnable interface because in java we can extend only one class.

If we are not going to extend any class then we can extend the thread class.

104. **Explain the thread life cycle in Java.**

**Answer:** **Thread has the following states:**

* New
* Runnable
* Running
* Non-runnable (Blocked)
* Terminated

105. **What is the purpose of a Volatile Variable?**

Volatile variable values are always read from the main memory and not from thread’s cache memory. This is used mainly during synchronization. It is applicable only for variables.

**Example:**

volatile int number;

### 106. What are the default values assigned to variables and instances in java?

* There are no default values assigned to the variables in java. We need to initialize the value before using it. Otherwise, it will throw a compilation error of (**Variable might not be initialized**).

### **107.** Can the static methods be overloaded?

Yes! There can be two or more static methods in a class with the same name but differing input parameters.

### **108.**  Why does the java array index start with 0?

It is because the 0 index array avoids the extra arithmetic operation to calculate the memory address.