1. **What is an Exception in java?**

**Answer:** An exception is an event, which occurs during the execution of a program that disrupts the normal flow of the program's instructions. The Java programming language uses exceptions to handle errors and other exceptional events. Java programming language provides two broad categories of exceptions known as checked and unchecked exceptions.

1. **What is Assertion?**

Assertion is a subclass of Error class. By the assertion we can test unusual condition during testing of a program.

1. **What is Stream?**

Answer: A stream is a flow of data from a source to a sink. Sources called input stream & we can only read from it. Sinks called output stream & we can only write to a stream.

There are two classify of stream:

* Byte Stream
* InputStream
* OutputStream
* Character Stream
* Reader
* Writer

1. **What is Serialization?**

Answer: Serialization: The process of writing an object to somewhere is called object serialization. The basic concept of object serialization is the ability to read & write objects to the byte streams. To make the objects of a class serializable , the class must implements the **Serializable** interface.

1. **What is wrapper class?**

Java programming language provides wrapper classes to manipulate primitive data elements as objects. Such data elements are wrapped in an object created around them.Each primitive data types has corresponding wrapper class in the java.lang package.

The following two statements illustrate the difference between a primitive data type and an object of a wrapper class:

int x = 25;

Integer y = new Integer(33);

1. **Describe the equals() method.**

Equals():

equals() method checks the equality of the content.

1. **What is collection?**

Answer: A collection is a single object managing a group of object. A group of objects known as elements. Collection allows us to a specific ordering and duplicate are permitted.

There are two type of collection:

Set-An unordered collection & no duplicate are permitted.

List-An ordered collection & duplicates are permitted.

Another type of collection :

Map-Another collection of map which store pair keys against values.

1. **What is auto boxing in java?**

**Answer:** During assignment, the automatic transformation of primitive type(int, float, double etc.) into their object equivalents or wrapper type(Integer, Float, Double,etc) is known as Autoboxing.

int x = 25;

Integer y = x; //It is autoboxing.

1. **What kind of ways can you create a thread? Which way is better and why?**

Ans: There are two way to create a thread:

1. using the thread class: Define the thread by writing your class that extends the Thread class and by overriding its run() method and instantiate the thread by instantiating your class. Then start the thread by executing the start() method.
2. Using the runable interface: Write the class that implements the Runnable interface, and implements the run() method of the Runnable interface. Then make an object of the thread class by passing instance in the argument of thread constructor of your runable class. And start the thread by invoking the start() method.
3. **Write the Lifecycle method of a thread?**

Answer: The lifecycle method of a thread is given below:

* New
* Runnable
* Running
* Nonrunnable states
* Blocked
* Sleeping
* Waiting
* Dead

1. **Define**

**Wait()**

**Notify()**

**NotifyAll()**

**Run()**

**Start()**

**Sleep()**

Ans: wait(): the java.lang.Object class provides wait for thread. A thread executes synchronized code that contains a wait call on a particular object. Thread is placed int the wait pool for that object and wait releases that object’s lock flag automatically.

Sleep(): The sleep is a static method in the Thread class. Thread.sleep() call, asking the thread to pause deliberately for a fixed period of time. The sleep method’s argument specifies the minimum number of milliseconds for which the thread must be made inactive.

1. **What do you mean by polymorphism?**

**Answer:** Polymorphism is the feature that allows one interface to be used for general class actions. It enables the same method to behave differently on different class.

1. **What do you mean by short-circuit logical operator?**

the conditional or short-circuit operator (&&) will not bother to evaluate the right hand operand if the left hand operand false.

1. **What do tou mean by Interface?**

An interface is a collection of abstract methods. A class implements an interface, thereby inheriting the abstract methods of the interface.

An interface is not a class. Writing an interface is similar to writing a class, but they are two different concepts. A class describes the attributes and behaviors of an object. An interface contains behaviors that a class implements.

The **interface** keyword is used to declare an interface.

An interface can extend another interface.

1. **What do you mean by garbage collection?**

**Answer:** The essential purpose of garbage collection is to reclaim the memory space that was occupied by the objects that are no longer required for the application. Garbage collection is done in java by a program called garbage collector. An object in the application is subjected to garbage collected when it is unreachable to the application.

1. **What do you mean by method overriding?**

**Answer: Method overriding:** It means the ability of a subclass to override a method allows a class to inherit from a superclass whose behavior is "close enough" and then to modify behavior as needed.

The overriding method has the same name, number and type of parameters, and return type as the method it overrides. When overriding a method, you might want to use the @Override annotation

1. **What is a constructor?**

A class contains constructors that are invoked to create objects from the class blueprint. Constructor declarations look like method declarations—except that they use the name of the class and have no return type. For example

Class Constructor() {

String symbol;

Constructor(String stockSymbol) {symbol = stockSymbol}

}

Every class has at least one constructor. Default constructor is a no argument constructor. If we do not declare a constructor the Java programming language provides one for us. This constructor takes no arguments and has an empty body. For example

Class Constructor() {

String symbol;

//no constructor declared

}