Top 40 Multithreading Interview Questions and Answers (2022)

1) What is a Thread?

A Thread is a concurrent unit of execution. We can say that it is part of the process which can easily run concurrently with other parts of the process.

2) What is multithreading?

Multithreading is a process of executing more than one thread simultaneously. The main advantage is:

- Threads share the same address space
- Thread remains lightweight
- Cost of communication between threads is low.

3) What are the states associated with the thread?

- Ready
- Running
- Waiting
- Dead state

4) Discuss a Thread Life Cycle?

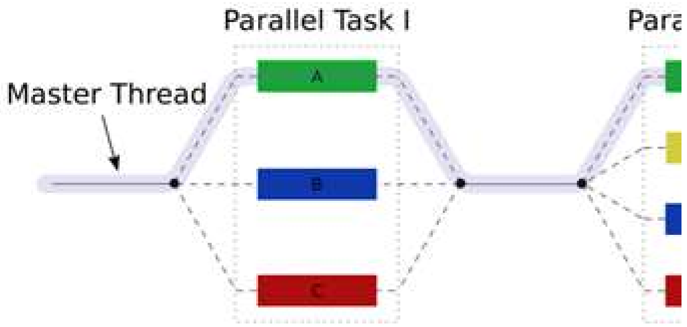
The life cycle of a thread is similar to the life cycle of processes running in an operating system. During its life cycle, the thread can move from one state to another. However, it depends on the operation performed on it.

5) What are the thread states?

Following are the different thread states:

New: A thread which is just instantiated is in the new state. When a start() method is invoked, the thread becomes the ready state. Then it is moved to the runnable state by the thread scheduler.

- **Runnable:** A thread which is ready to run
- **Running:** A thread which is executing is in running state.
- **Blocked:** A blocked thread is waiting for a monitor lock is in this state. This thing can also happen when a thread performs an I/O operation and moves to the next state.
- **Waiting:** It is a thread that is waiting for another thread to do the specific action.
- **Timed_waiting:** It is a thread that is waiting for another thread to perform.
- **Terminated:** A thread that has exited is in this state.



6) What are the major differences between Thread and Process?

The thread is a subset of process. The process can contain multiple threads. The process can run on on different memory space, but all threads share the same memory space.

7) What is deadlock?

Deadlock is a situation when a thread is waiting for an object lock, that is acquired by another thread and second thread also waiting for an object lock that is acquired by the first thread. As both threads are waiting for each other to release this condition is called deadlock.

8) What is LiveLock?

Livelock occurs when all threads are blocked and not able to execute because of the unavailability of required resources, and non-existence of any unblocked thread.

Livelock can occur in the following conditions:

- When all the threads in a program executed on an object with zero parameters. The program is live-locked and never processed until one or more threads call

Object.notify () or Object.notifyAll() on the relevant objects.
- Livelock also happens when all the threads in a program are stuck in infinite loops.

9) What is a monitor?

- The monitor is a body of code that can be executed by only one thread at a time.
- If any other thread attempts to get access at the same time, it will be suspended until the current thread releases the Monitor.

10) What do you mean by thread starvation?

In the situation when a thread does not have sufficient CPU for its execution Thread starvation happens.

However, it may happen in the following scenarios

- Low priority threads will get less CPU compared to high priority threads. Lower priority thread can starve away waiting to get more CPU space to perform calculations.
- The thread may be waiting indefinitely for a lock on object's monitor but notify() may repeatedly be awakening some other threads. In that case, also thread starves away.

11) What is the meaning of busy spin in multi-threading?

Busy spin is a technique that concurrent programmers employ to make a thread wait on certain condition. This is quite different from traditional methods like wait() and sleep()which all involves relinquishing CPU control. This method does not require abandon CPU, instead it the just runs the empty loop.

12) What is context-switching in multi-threading?

It is the process of storing and restoring of CPU state. This helps to resume thread execution from the same point at a later point in time. It is one of the essential features for multitasking operating system and support for the multi-threaded environment.

13) Why Thread behavior is unpredictable?

We can say that thread behavior is unpredictable because the execution of Threads depends on Thread scheduler. One should remember that every thread scheduler has a different implementation on different platforms like Windows, Unix, etc

14) How can you pause the execution of a Thread for a certain amount of time?

sleep () method is used to pause the execution of the thread for a certain amount of time. However, this will not stop the processing of thread for a specific time. However, when the thread awake from sleep its state changes to runnable and based on thread scheduling, it will be executed.

15) What are numerous ways in which a thread can enter the waiting state?

A thread can enter the waiting state by the following ways

- Using sleep() method
- Blocking on I/O
- Unsuccessfully trying to acquire an object's lock
- By invoking an object's wait() method

16) What happens if we don't override a run method?

In Java, when we call start() method on a thread, It internally calls run() method with newly created thread. So, if we don't override run( ) method, then newly created thread won't be called so nothing will happen.
Example:

```
class MyThread extends Thread {

//don't override run() method

}
public class DontOverrideRun {

public static void main(String[] args) {

System.out.println("main has started.");

MyThread thread1=new MyThread();
thread1.start();

System.out.println("main has ended.");
}

}
```

17) What is the major difference between Thread.start() & Thread?run() method?

Thread.start() method (native method) of Thread class does the job of running the Thread.A run() method in a thread. So, if we directly call Thread.The run() method also execute in the same thread. Thus it will never solve the purpose of creating a new thread.

18) How to create a thread in java?

There are two methods to create a thread in java.

- First by implementing Runnable interface and then create a thread object from it.
- The second method is to extend the thread class.

19) What is the meaning of Thread Priority?

Every thread has a priority. However, a higher priority also gets precedence in execution. However, it also depends on Thread Scheduler implementation which is OS dependent. It is possible to the change the priority of the thread, but it does not give assurance that higher priority thread will get executed first.

20) What join() method does?

The join() method waits for a thread to die. It forces all the running threads to stop executing till the time the thread joins to complete its job.

21) What is Java Shutdown Hook?

The Java shutdown hook is used to clean up resources when JVM shuts down. Clean resources mean closing log file, sending some alerts or something else. Shutdown hook needs to be used to execute code before JVM shuts down.

22) What are the two main uses of volatile in Java?

Threads are allowed to hold the values of variables in local memory If a variable is marked as volatile, then every time the same variable is used, it must be read from the main memory.

In the same way, every time the variable is written, the value must be stored in main memory.

23) How can you share data between two thread in Java?

We can get data between threads by using a shared object, or concurrent data structure like BlockingQueue. It implements a producer-consumer pattern using wait and notifies methods. It also involves sharing objects between two threads.

24) How can multiple threads be controlled simultaneously?

Multiple threads can be simultaneously controlled if they are created in a ThreadGroup object.

25) What is a blocking method in Java?

In Java blocking method is a method which blocks until the task is done. For example, accept () method of ServerSocket blocks until the time a client is connected. Here, blocking refers anything that control will not return to the caller until the task is over.

26) What is the main difference between wait () and sleep () method?

Wait()	Sleep()
This method is defined in the Object class	The method is defined in Thread class
Wait() method releases the lock	This method never releases the lock.

27) What is an immutable object? How can it help in writing a concurrent application?

Any object can be considered unchallengeable if its state does not change after it is constructed. Immutable Objects are used in creating simple, reliable, and concurrent applications.

For creating object immutable, it is important to make the class and its member final so that once objects get created, it’s state doesn’t modify.

28) Tell me the difference between yielding and sleeping?

When a task invokes its yield() method, it returns to the ready state.

When a task invokes its sleep() method, it returns to the waiting state.

29) What is ThreadPool?

ThreadPool is a pool of threads that reuses a fixed number of threads to execute the specific task.

30) What is the use of Synchronized keyword?

Synchronized keyword can be applied either to the static or non-static method. Using Synchronized only one thread can access synchronized methods. However, in the situation where there are multiple threads which are trying to access the same method. At that time, other threads have to wait for the execution thread. It also provides a lock on the object to prevent a race condition.

```
public void synchronized method1(){}

public void synchronized staticmethod1(){}

public void myMethod(){

synchronized (this){

//synchronized keyword on block of code

}

}
```

31) What is a volatile keyword?

Volatile keyword is a qualifier which is applied to a variable when it is declared. It tells the compiler that the value of the variable may change at any time–without any action being taken by the code.

32) What are the main differences between notify and notifyAll in Java?

Notify () method doesn't provide any way to choose a particular thread, that's why it's only useful when a single While notifyAll() sends a notification to all threads. It also allows them to compete for locks. It also ensures that at least one thread will proceed further.

33) Which JVM parameter is used to control the stack size of a thread?

To control the stack size of Thread in Java Xss parameter is used.

34) Can you start a thread twice in Java?

No, once a thread is started, it can't be started the second time.

35) What is the purpose of using the yield method of thread class?

Yield method is the simplest way to request current thread to relinquish CPU so that other thread. It is a static method and only guarantees that the current thread will relinquish the CPU but doesn't tell anything about which other thread also impacts the CPU.

36) When can we say that threads are not lightweight process in java?

Threads are not lightweight process when threads of the same process are executing simultaneously. Although, if threads of different processes are executing concurrently at that time threads become a heavyweight process.

37) Can It is possible to synchronize the constructor of a Java Class?

As Java Standard, constructors cannot be synchronized as other threads cannot see the object before the thread creation process has finished it. There is no need of Java Objects constructor to be synchronized since as it would lock the object being constructed.

38) What is transient variable?

A transient variable is a variable that can't be serialized during serialization. It is initialized to its default value during the serialization.