Vibhor Mehta DevOps Assignment-4

1.How to perform two tasks by two threads ?

To perform two tasks using two threads use the run() function.

2.How to perform multithreading by anonymous class ?

The first way is to extend the Thread class, override the run() method with the code you want to execute, then create a new object from your class and call start(). The second method is to pass an implementation of the Runnable interface to the constructor of Thread, then call start().

3.What is the Thread Scheduler and what is the difference between preemptive scheduling and

time slicing?

The thread scheduler is an operation done in the CPU where the threads are made in order such that they can be executed. Preemptive scheduling is a process putting all the high-priority threads first in line for execution whereas time slicing is how much time should the execution of a single thread take.

4.What happens if we start a thread twice ?

It will not run since we first need to finish the first thread only then can we run it a second there we cannot start two threads of the same type at once.

5.What happens if we call the run() method instead of start() method ?

The run() method is just an ordinary method. As with any other ordinary method and calling it directly will cause the current thread to execute run().The start() method will cause the JVM to spawn a new thread and make the newly spawned thread execute run().

6.What is the purpose of join method ?

Join method is used to join two threads where one is almost getting over and the other needs to start therefore making it a simultaneous start and end process.

7.Why JVM terminates the daemon thread if there is no user threads remaining?

Daemon thread is a low priority thread that runs in background to perform tasks such as garbage collection. They can not prevent the JVM from exiting when all the user threads finish their execution. .JVM terminates itself when all user threads finish their execution. If JVM finds running daemon thread, it terminates the thread and after that shutdown itself. JVM does not care whether Daemon thread is running or not. It is an utmost low priority thread.

8.What is the shutdown hook?

Shutdown Hooks are a special construct that allow developers to plug in a piece of code to be executed when the JVM is shutting down. This comes in handy in cases where we need to do special clean up operations in case the VM is shutting down.

9.What is garbage collection ?

Garbage Collection is process of reclaiming the runtime unused memory automatically. it is a way to destroy the unused objects. It makes java memory efficient because garbage collector removes the unreferenced objects from heap memory. It is automatically done by the garbage collector so we don't need to make extra efforts.

10.What is the purpose of finalize() method ?

The finalize() is called by the garbage collector on an object when garbage collection determines that there are no more references to the object. A subclass overrides the finalize method to dispose of system resources or to perform other cleanup.

11.What does gc() method ?

The gc() method runs the garbage collector in java.

12.What is synchronization and why use synchronization?

Synchronization is used to to protect access to resources that are accessed concurrently.

13.What is the difference between synchronized method and synchronized block?

Synchronized methods enable a simple strategy for preventing thread interference and memory consistency errors whereas synchronized block ensures atomicity of bunch of code statements.

14.What are the two ways to perform static synchronization?

* To create a private lock on the object
* To synchronize on the object itself.

15.How do threads communicate through wait() and notify()?

The wait() method causes a current thread to release the lock and wait for another thread to invoke the notify() method. The notify() method wakes up a single thread that is waiting on the object’s monitor.

16.What is interthread-communication or cooperation?

Interthread communication is allowing synchronized threads to communicate with each other. This is a mechanism in which a thread is paused running in its critical section and another thread is allowed to enter in the same situation to be executed.