**1. What is meant by a thread in runnable state?**

Resources are allocated to the process, thread is alive and running

**2. Which method invokes the run() method?**

The start() method

**3. What is the time unit of sleep method?**

milliseconds

**4. What are the advantages / reasons of using threads instead of processes?**

* Better utilization of resources (threads share address space)
* Easier to manipulate threads than process by the OS
* Threads are faster than a process

**Note: Disadvantage – stronger synchronization required**

**5. Which are the two ways to implement threads?**

By extending the thread class or implementing runnable and then overloading each methods run method.

**6. Which scheduling algorithm Java uses?**

Preemptive priority scheduling

**7. What is NORM\_PRIORITY (default priority) in Java?**

Min = 1, max = 10, and norm priority is 5 (default) in java.

**8. Explain Try Catch block inside sleep method.**

In the wait() or sleep() method, if a process is interrupted and then blocked, the method that called it throws an interruptexception object.

**9. Thread Methods**

**(deprecated = not recommend to use)**

* Suspend()
* Wait()
* Sleep()
* Start()
* isAlive()
* join()

**10. Advantages / Disadvantages****of SA’s (S.A. = Scheduling Algorithm)**

*Non-preemptive Scheduling:*

*Shortest Job First:*

* Advantages-
* Disadvantages-

*FCFS:*

* The ready queue contains the PCB of the process and it is treated like a FIFO queue.
* Advantages- Easy to implement because there are no computations
* Disadvantages- Large processes in front of the queue can create convoy effect🡪 creates poor average waiting/ turnaround time

*Preemptive Scheduling:* The schedule can forcibly remove a process from the CPU/context switching

*Shortest Remaining Time:*

* Process with the shortest CPU burst is scheduled. If a new process (P2) arrives in the ready queue with a shorter service time, P2 is scheduled and P1 is preempted.
* Advantages- Optimal average waiting time (among all preemptive SA’s)
* Disadvantages- Hard to predict CPU burst, not suitable for CPU.

*Round Robin:*

* Used for Time-sharing environments. Processes are dispatched in a FIFO sequence but each process can only run for a limited amount of time, known as a **time-slice** or **quantum**.
* Advantages- fair in using CPU time
* Disadvantages- poor average wait and turnaround times
* Special case at Arrival time zero, where multiple processes can arrive at this time.

Threads:

* Multithreading allows a process to have more than 1 thread.
* Heavy Weight process has 1 thread and 1 task.
* Light Weight process consists of a program counter, register set and a stack space. All LWP use same address space.

Main Memory of threads contain:

* Thread Control Block
* Execution Stack
* Minimal Resources (Set of Registers)
* Threads are efficient because they work as a team and they are sharing the same resources.
* Advantages:
* Better utilization of resources (threads share address space)
* Easier to control and manipulate threads than process
* Creating a thread is faster than creating another process.

Disadvantages:

* Operating System needs to support multithreaded environment. Synchronization is difficult and stricter because threads are sharing resources.
* Process Thread is the main thread and executes in main memory. If main thread terminates all threads terminate.

Thread types:

* Kernel level thread 🡪 communicate directly with kernel, system calls.
* User level thread 🡪 above kernel, library calls.

Thread Process States:

**New:** an object for the thread is created. No system resources have been allocated yet.

start()Make a thread runnable. Resources are allocated to the thread; the thread goes into the Runnable state.

**Runnable**: yield()

**Blocked:** wait(), sleep(), join()

**Terminate**: exit(), abnormal exit by user, completion

**Dead state**: the thread exits (terminates).

If thread is in wait(); notify() or notifyall() must be called.

isAlive() – returns a Boolean value that determines if a Thread is in the Dead state or not.

### Windows

Provides time slicing for threads with same priority

**Show Round Robin diagram and Shortest Job First Diagrams:**