**Multithreading Assignments by gaurav diwan**

**Java with DSA and System design**

Q1] what do you mean by multithreading and why is it important?

Ans. Multithreading is the concurrent execution of multiple threads (smaller units of a process) within a single program.

Importance:

1. Concurrency: Enables tasks to run in parallel, improving program efficiency.

2. Responsiveness: Keeps a program responsive by allowing simultaneous execution of tasks.

3. Resource Utilization: Efficiently utilizes system resources, enhancing overall performance.

Q2] what are the benefits of multithreading?

1. **Concurrency:** Allows multiple tasks to run in parallel.
2. **Responsiveness:** Keeps programs responsive during task execution.
3. **Resource Utilization:** Efficient use of system resources.
4. **Improved Performance:** Enhances overall program performance.

Q3] what is thread in java?

Ans. In Java, a thread is the smallest unit of execution that can execute independently. It is a lightweight, concurrent process.

Q4] what are the two ways to implement thread in java?

1. **Extending Thread Class:** Creating a class that extends the **Thread** class.
2. **Implementing Runnable Interface:** Implementing the **Runnable** interface and passing an instance to the **Thread** constructor.

Q5] what’s the difference between thread and process ?

Ans. Thread:

- Smallest unit of execution.

- Shares memory space with other threads in the same process.

Process:

- Independent program execution unit.

- Has its own memory space and resources.

Q6] how can we create daemon threads ?

Ans. To create a daemon thread in Java, set the thread as a daemon before starting it:

Thread daemonThread = new Thread(() -> {

// Thread logic

});

daemonThread.setDaemon(true); // Set as daemon

daemonThread.start();

Q7] what are the wait() and sleep() methods ?

Ans. - `wait()`: Used for inter-thread communication. Releases the lock and waits until another thread notifies.

- `sleep()`: Pauses the execution of the current thread for a specified amount of time. Doesn't release the lock.

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***