Experiment No: 12

AIM: To demonstrate the use of multithreading.

Date:

CO mapped: CO-3

Objectives:

a) To effectively demonstrate the use of multithreading in software applications, including creating and managing multiple threads, synchronizing their execution, and leveraging the power of concurrent programming to improve performance, responsiveness, and resource utilization.

b) Demonstrating the use of multithreading is crucial for building responsive and efficient software applications, and this objective emphasizes understanding the concepts and practical implementation of multithreading to achieve these goals.

Background:

Multithreading in Java is a process of executing multiple threads simultaneously. A thread is a lightweight sub-process, the smallest unit of processing. Multiprocessing and multithreading, both are used to achieve multitasking. However, we use multithreading than multiprocessing because threads use a shared memory area. They don't allocate separate memory areas so saves memory, and context-switching between the threads takes less time than the process. Java Multithreading is mostly used in games, animation, etc.

Practical questions:

1. Write a program to create a thread extending Thread class and demonstrate the use of sleep() method.
2. Write a program to create a thread implementing Runnable interface and demonstrate the use of join() method.
3. Write a program that launches 10 threads. Each thread adds 1 to a variable sum that initially is 0. Define an Integer wrapper object to hold sum. Run the program with and without synchronization to see its effect.

Observations: Put Output of the program

Conclusion: (Sufficient space to be provided)

Quiz: (Sufficient space to be provided for the answers)

1. Can you explain the difference between a process and a thread in the context of multithreading?
2. What are the different states in the lifecycle of a Java thread, and how does a thread transition between them?
3. What is runnable interface? How can you use this interface in creating thread?

4. Explain the concept of thread synchronization and the role of the synchronized keyword.
5. Explain: wait, sleep, notify and notify all.

Suggested Reference:

1. https://www.tutorialspoint.com/java/
2. https://www.geeksforgeeks.org/
3. https://www.w3schools.com/java/
4. https://www.javatpoint.com/