1. **What do you mean by Multithreading? Why is it important?**
   - **Multithreading** is a programming concept where multiple threads are executed concurrently within a single program. It is important because it allows for more efficient use of resources, improves performance, and enables the execution of multiple tasks simultaneously.

2. **What are the benefits of Multithreading?**
   Benefits of Multithreading include:
   - Improved performance and responsiveness
   - Better resource utilization
   - Simplified modeling of real-world problems
   - Enhanced application throughput

3. **What is Thread in Java?**
   A **Thread** in Java is a lightweight process that can run concurrently with other threads within the same program. It is the smallest unit of execution in a Java program.
4. **What are the two ways of implementing thread in Java?**
   The two ways to implement threads in Java are:
   - **Extending the Thread class:** Create a new class that extends Thread and override the run() method.
   - **Implementing the Runnable interface:** Create a new class that implements the Runnable interface and pass an instance of this class to a Thread object.

5. **What's the difference between thread and process?**
   - A **Thread** is a smaller unit of a process that can run concurrently with other threads within the same process.
   - A **Process** is an independent program that runs in its own memory space. Threads within the same process share the same memory space, while processes do not.

6. **How can we create daemon threads?**
   - **Daemon threads** are low-priority threads that run in the background to perform tasks such as garbage collection. To create a daemon thread, use the setDaemon(true) method on a Thread object before starting it.

7. **What are the** wait() **and** sleep() **methods?**
    - The **wait()** method is used to make a thread wait until another thread invokes the notify() or notifyAll() methods on the same object. It is used for inter-thread communication.
    - The **sleep()** method is used to pause the execution of the current thread for a specified period. It is used to simulate delays or to give other threads a chance to execute.