

Q1. What do you mean by multithreading? Why is it important?

A1. Multithreading means multiple threads and it is considered as one of the most important feature of Java. As the name suggests. It's ability of a CPU to execute multiple threads independently at the same time but share the process resources simultaneously. Its main purpose is to provide simultaneous execution of multiple threads to utilize the CPU time as much as possible. It is a Java feature where one can subdivide the specific program into two or more threads to make the execution of the program fast and easy.

Q2. What are benefits of using Multithreading?

A2 The Benefits are as follows:-

- i. Increase CPU Efficiency.**
- ii. Increase use of CPU Resources.**
- iii. Saves time and parallelism tasks.**

Q3. What is a Thread in Java?

A3. Threads are basically the lightweight and smallest unit of processing that can be managed independently by a scheduler.

Q4. What are the two ways of implementing thread in Java?

A4. The two ways of implementing thread in Java are as follows :-

- i. Extending the Thread class,**
- ii. Implementing Runnable Interface in Java.**

Q5. What's the difference between Thread and Process?

A5. Thread: It simply refers to the smallest units of the particular process. It has the ability to execute different parts (referred to as thread) of the program at the same time.

Process: It simply refers to a program that is in execution i.e., an active program. A process can be handled using PCB (Process Control Block).

Q6. How can we create Daemon Threads?

A6. We can create daemon threads in java using the thread class setDaemon(true). It is used to mark the current thread as daemon thread or user thread. isDaemon() method is generally used to check whether the current thread is daemon or not. If the thread is a daemon, it will return true otherwise it returns false.

Q7. What are the wait() and sleep() methods?

A7. wait(): As the name suggests, it is a non-static method that causes the current thread to wait and go to sleep until some other threads call the notify () or notifyAll() method for the object's monitor (lock).

sleep(): As the name suggests, it is a static method that pauses or stops the execution of the current thread for some specified period.