

Multithreading

What is multithreading in Java?

Multithreading is a process of executing multiple threads concurrently. Multithreading enables programs to perform multiple tasks at the same time, it improves performance and responsiveness.

What is Multitasking?

Multitasking is a process of executing multiple tasks simultaneously.

What is a thread in Java?

A thread is a lightweight process that helps a program to perform multiple tasks concurrently. It shares the same memory space and resources but executes independently, and helps to achieve parallelism and responsiveness.

How do you create a thread in Java?

By extending the Thread class and overriding its run method.

By implementing the Runnable interface and passing it to a Thread object.

What is the difference between Thread and Runnable?

We can create a thread with both thread class and Runnable interface. if we're using Thread class. There is no chance of extending any other class. means we are missing Inheritance benefits. on the other side if we're using a Runnable interface we can use the benefits of Inheritance. Another difference is When we extend the Thread class, each of our threads creates a unique object and associates it with it. When we implement Runnable, it shares the same object to multiple threads.

What are the states of a thread in Java?

- New
- Runnable
- Blocked
- Waiting
- Timed Waiting
- Terminated

What is synchronization in Java?

Synchronization is the capability to control the access of multiple threads to any shared resource.

How do you synchronize a method in Java?

By using the synchronized keyword.

What is a deadlock?

A Deadlock is a situation where two or more threads are blocked forever, waiting for each other to release the resources they need. It generally occurs when multiple threads hold locks on different resources and are waiting for other resources to complete their task.

What is the wait method in Java?

It is used to pause the execution of a thread until another thread signals that it can resume.

What is the notify method in Java?

It is used to give the notification for only one thread which is waiting for a particular object.

What is the notifyAll method in Java?

It is used to give a notification to all waiting threads of a particular object.

What is the volatile keyword in Java?

The volatile keyword is used to indicate that a variable's value will be modified by different threads, ensuring visibility and ordering of changes.

What is the join method in Java?

The join method allows one thread to wait for the completion of another thread.

What is the ExecutorService in Java?

ExecutorService interface extends Executor by adding methods that help manage and control the execution of threads.

What is the Future of Java?

Future represents the result of an asynchronous computation. It provides methods to check if the computation is complete, wait for its completion, and retrieve the result. Asynchronous computation means running tasks in the background without blocking the main thread.

What is the Callable interface in Java?

Callable is similar to Runnable but can return a result and throw a checked exception.

What is a ThreadPool in Java?

A thread pool is a collection of threads that are managed by a thread pool manager. When a task is submitted to the thread pool, it assigns the task to an available thread from the pool.

What is the difference between sleep() and wait()?

sleep pauses the current thread for a specified time, while wait pauses the thread until another thread calls notify or notifyAll.

What is the start() and run() method of the Thread class?

start(): Creates a new thread and then calls the run method.

run(): Contains the thread's code; calling it directly does not create a new thread.

What's the difference between thread and process?

Process means any program is in execution.

Thread means executing different parts of the program at the same time.

What's the difference between the User thread and the Daemon thread?

User threads are high-priority threads. The JVM will wait for any user thread to complete its task before terminating it. On the other hand, daemon threads are low-priority threads whose only role is to provide services to user threads.

How do threads communicate with each other?

Threads can communicate using three methods i.e., wait(), notify(), and notifyAll().

What is thread starvation?

Thread starvation is a situation or condition where a thread won't be able to have regular access to shared resources and therefore is unable to proceed or make progress. This is because other threads have high priority and occupy the resources for too long.

Advantages of Multithreading?

- Allow the program to run continuously even if a part of it is blocked.

- Increase the use of CPU resources and reduce costs of maintenance.
- If an exception occurs in a single thread, it will not affect other threads as threads are independent.
- Allows to write effective programs that utilize maximum CPU time.

What is Livelock?

A livelock is a situation where two or more threads keep changing their states in response to each other but fail to make any progress.

What is thread priority?

Thread priority is an integer value from 1 to 10, with 1 being the lowest priority and 10 being the highest, representing thread priority. It helps the operating system determine the order in which threads are scheduled.

What is a race condition?

A race condition occurs when two or more threads can access shared data and they try to change it at the same time.

What is the difference between a synchronized block and a synchronized method?

[synchronized method](#): it means only one thread can execute it at a time.

[synchronized block](#): it provides more fine-grained control and potentially better performance.

What is the Executor framework in Java?

The Executor framework provides a higher-level replacement for working directly with threads. It simplifies thread management by providing various implementations like `ThreadPoolExecutor` and `ScheduledThreadPoolExecutor`.

What is the ConcurrentHashMap?

`ConcurrentHashMap` is a thread-safe variant of `HashMap` that allows concurrent read and write operations without locking the entire map, improving performance in a multi-threaded environment.

What are atomic classes in Java?

Atomic classes, such as `AtomicInteger`, `AtomicBoolean`, and `AtomicReference`, provide thread-safe operations on single variables without using synchronization.