

# Parallel and Concurrent Programming

## Java Parallel Streams

Java Parallel Streams divide a running process into multiple streams that execute in parallel on separate cores, returning a combined result of all the individual outcomes.

## Parallelism

Parallelism is the act of splitting tasks into smaller subtasks and processing those subtasks in parallel, for instance across multiple CPUs at the exact same time.

## Java Streams

A Java Stream is used to process a collection of objects which can be pipelined to produce a desired result.

## Executor Framework

The Executor framework implements thread pooling through an `Executor` interface.

## Concurrency

Concurrency is the act of processing more than one task at seemingly the same time on the same CPU, requiring the ability to switch between tasks.

## Memory Consistency Errors

Memory consistency errors occur when different threads have inconsistent views of what should be the same data.

## Fork-Join Framework

The Fork-Join framework uses `ForkJoinPool` to distribute a task across several worker threads and then wait for a result.

## Thread Interference

Thread interference can occur when one thread overwrites the results of another thread in an unpredictable way, which can cause unexpected results when reading the altered data.

## Thread Pools

A thread pool manages a pool of worker threads which connect to a work queue of `Runnable` tasks waiting to be executed.