

Java 多线程并发

基础

为什么会有多线程

- 摩尔定律失效
- 处理器核心数量越来越多
- 更快的响应时间
- 更好的编程模型

线程创建过程

Thread#start()

线程的不同形式

- Thread
- Runnable

线程的状态

- new
- runnable
- blocked
- waiting
- timed_waiting
- terminated

volatile

- 保证线程间可见性
- 不能保证原子性

happens-before 原则

- 程序次序规则
- 锁定规则
- volatile 变量规则
- 传递规则
- 线程启动规则
- 线程中断规则
- 线程终结规则
- 对象终结规则

synchronized

线程池

Java 并发包 (JUC)

锁机制类

- Lock
- Condition
- ReentrantLock
- ReadWriteLock
- LockSupport

原子操作类

- AtomicInteger
- AtomicLong
- LongAdder

线程池相关类

- Future
- Callable
- Executor
- ExecutorService

信号量工具类

- CountDownLatch
- CyclicBarrier
- Semaphore

并发集合类

- CopyOnWriteArrayList
- ConcurrentMap