



面试虐我千百遍，Java 并发真讨厌

小马哥 (mercyblitz)



关于我

小马哥，Java 劝退师，Apache 和 Spring Cloud 等知名开源架构成员。通过 SUN Java (SCJP、SCWCD、SCBCD) 以及 Oracle OCA 等的认证。

主要线上分享

慕课网 - Spring Boot 2.0深度实践之核心技术篇

慕课网 - Spring Boot 2.0深度实践之生态整合篇 (即将上线...)

著书

《Spring Boot 编程思想》(京东预售...)

Github : <https://github.com/mercyblitz>

议程



心态篇

自我否定
摇尾乞怜
幡然醒悟



实战篇

Java 多线程：从操作系统底层说起
Java 并发框架：基础，基础，还是基础
Java 内存模型：没你想得那么简单



职业篇

提升阶段：如何学好 Java
成熟阶段：多元化思考
放弃阶段：学会放弃 Java

PART 01

心态篇

面试嘛，总在懵逼和牛逼之间徘徊



面试如同恋爱，总让人患得患失



自我否定

面试尚未成功
职业面临自宫



摇尾乞怜

考官手下留点情
给个机会行不行



幡然醒悟

面试造火箭
工作拧螺丝

PART 02

实战篇

“From Java programming language to thinking”



Java 并发实战

Java 多线程

Java 并发集合框架

Java 并发框架

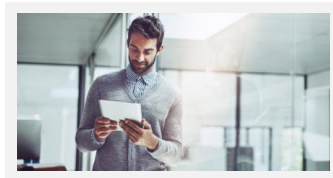


1、线程创建

01

基本版

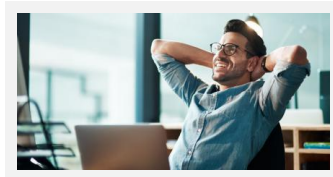
有哪些方法创建线程？



02

进阶版

如何通过 Java 创建进程？



03

劝退版

如何销毁一个线程？

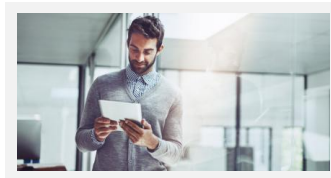


2、线程执行

01

基本版

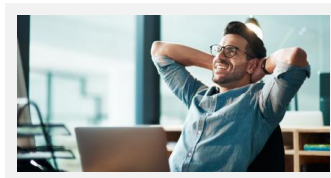
如何通过 Java API 启动线程？



02

进阶版

当有线程 T1、T2 以及 T3，如何实现 T1 -> T2 -> T3 的执行顺序？



03

劝退版

以上问题请至少提供另外一种实现？

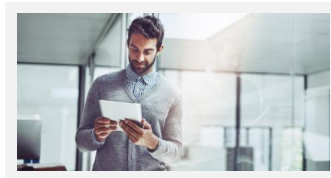


3、线程中止

01

基本版

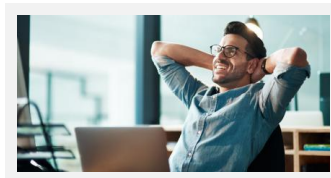
如何停止一个线程？



02

进阶版

为什么 Java 要放弃 Thread 的 stop() 方法？



03

劝退版

请说明 Thread interrupt()、isInterrupted() 以及 interrupted() 的区别以及意义？

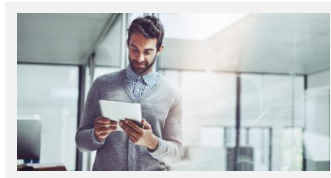


4、线程异常

01

基本版

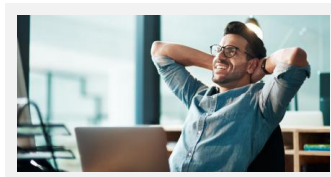
当线程遇到异常时，到底发生了什么？



02

进阶版

当线程遇到异常时，如何捕获？



03

劝退版

当线程遇到异常时，ThreadPoolExecutor 如何捕获异常？

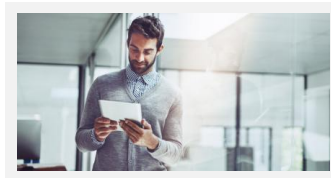


5、线程状态

01

基本版

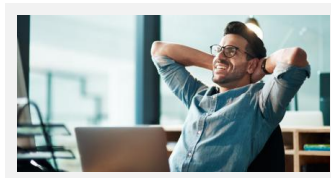
Java 线程有哪些状态，分别代表什么含义？



02

进阶版

如何获取当前 JVM 所有的线程状态？



03

劝退版

如何获取线程的资源消费情况？

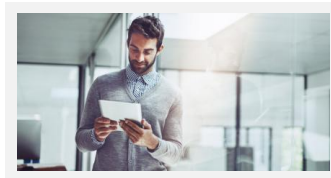


6、线程同步

01

基本版

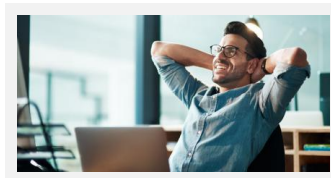
请说明 `synchronized` 关键字在修饰方法与代码块中的区别？



02

进阶版

请说明 `synchronized` 关键字与 `ReentrantLock` 之间的区别？



03

劝退版

请解释偏向锁对 `synchronized` 与 `ReentrantLock` 的价值？

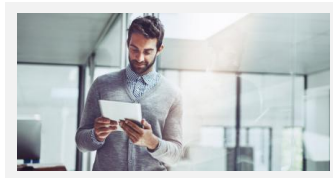


7、线程通讯

01

基本版

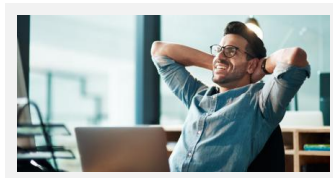
为什么 `wait()` 和 `notify()` 以及 `notifyAll()` 方法属于 `Object` , 并解释它们的作用 ?



02

进阶版

为什么 `Object` `wait()` 和 `notify()` 以及 `notifyAll()` 方法必须 `synchronized` 之中执行 ?



03

劝退版

请通过 Java 代码模拟实现 `wait()` 和 `notify()` 以及 `notifyAll()` 的语义 ?

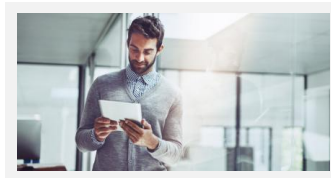


8、线程退出

01

基本版

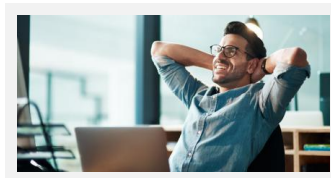
当主线程退出时，守候子线程会执行完毕吗？



02

进阶版

请说明 ShutdownHook 线程的使用场景，以及如何触发执行？



03

劝退版

如何确保在主线程退出前，所有线程执行完毕？



资源推荐



次灵均阅



《Spring Boot 编程思想 - 核心篇》
京东预售链接

次灵均阅

Java 并发实战

Java 多线程

Java 并发集合框架

Java 并发框架

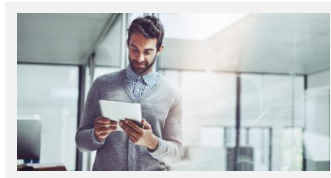


1、线程安全集合

01

基本版

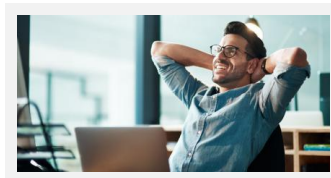
请在 Java 集合框架以及 J.U.C 框架中各举出 List、Set 以及 Map 的实现？



02

进阶版

如何将普通 List、Set 以及 Map 转化为线程安全对象？



03

劝退版

如何在 Java 9+ 实现以上问题？

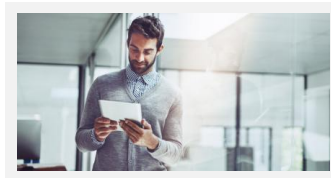


2、线程安全 LIST

01

基本版

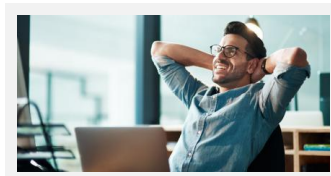
请说明 List、Vector 以及 CopyOnWriteArrayList 的相同点和不同点？



02

进阶版

请说明 Collections#synchronizedList(List) 与 Vector 的相同点和不同点？



03

劝退版

Arrays#asList(Object...) 方法是线程安全的吗？如果不是的话，如何实现替代方案？

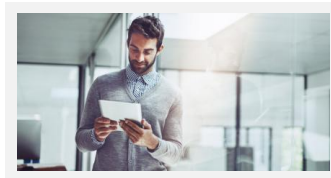


3、线程安全 SET

01

基本版

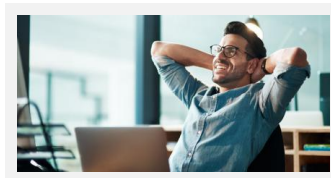
请至少举出三种线程安全的 Set 实现？



02

进阶版

在 J.U.C 框架中，存在 HashSet 的线程安全实现？如果不存在的话，要如何实现？



03

劝退版

当 Set#iterator() 方法返回 Iterator 对象后，能否在其迭代中，给 Set 对象添加新的元素？

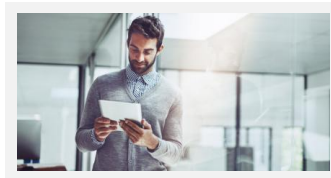


4、线程安全 MAP

01

基本版

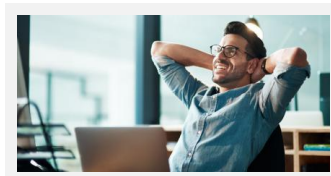
请说明 Hashtable、HashMap 以及 ConcurrentHashMap 的区别？



02

进阶版

请说明 ConcurrentHashMap 在不同的 JDK 中的实现？



03

劝退版

请说明 ConcurrentHashMap 与 ConcurrentSkipListMap 各自的优势与不足？

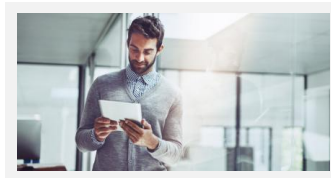


5、线程安全 QUEUE

01

基本版

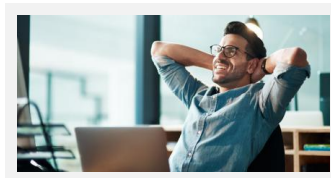
请说明 BlockingQueue 与 Queue 的区别？



02

进阶版

请说明 LinkedBlockingQueue 与 ArrayBlockingQueue 的区别？



03

劝退版

请说明 LinkedTransferQueue 与 LinkedBlockingQueue 的区别？



6、PRIORITYBLOCKINGQUEUE

请评估以下程序的运行结果？

```
public class PriorityBlockingQueueQuiz {  
  
    public static void main(String[] args) throws Exception {  
        BlockingQueue<Integer> queue = new PriorityBlockingQueue<>(2);  
        queue.put(9);  
        queue.put(1);  
        queue.put(8);  
        System.out.println("queue.size() = " + queue.size());  
        System.out.println("queue.take() = " + queue.take());  
        System.out.println("queue = " + queue);  
    }  
}
```

7、SYNCHRONOUSQUEUE

请评估以下程序的运行结果？

```
public class SynchronousQueueQuiz {  
  
    public static void main(String[] args) throws Exception {  
        BlockingQueue<Integer> queue = new SynchronousQueue<>();  
        System.out.println("queue.offer(1) = " + queue.offer(1));  
        System.out.println("queue.offer(2) = " + queue.offer(2));  
        System.out.println("queue.offer(3) = " + queue.offer(3));  
        System.out.println("queue.take() = " + queue.take());  
        System.out.println("queue.size = " + queue.size());  
    }  
}
```

8、BLOCKINGQUEUE OFFER()

请评估以下程序的运行结果？

```
public class BlockingQueueQuiz {  
  
    public static void main(String[] args) throws Exception {  
        offer(new ArrayBlockingQueue<>(2));  
        offer(new LinkedBlockingQueue<>(2));  
        offer(new PriorityBlockingQueue<>(2));  
        offer(new SynchronousQueue<>());  
    }  
  
    private static void offer(BlockingQueue<Integer> queue) throws Exception {  
        System.out.println("queue.getClass() = " + queue.getClass().getName());  
        System.out.println("queue.offer(1) = " + queue.offer(1));  
        System.out.println("queue.offer(2) = " + queue.offer(2));  
        System.out.println("queue.offer(3) = " + queue.offer(3));  
        System.out.println("queue.size() = " + queue.size());  
        System.out.println("queue.take() = " + queue.take());  
    }  
}
```

资源推荐



Spring Boot 2.0深度实践之核心技术篇

课程系统性地深度探讨 Spring Boot 核心特性，引导小伙伴对 Java 规范的重视，启发对技术原理性的思考，掌握排查问题的技能，以及学习阅读源码的方法和技巧，全面提升研发能力，进军架构师队伍。



小马哥
mercyblitz
微服务架构师

Java 并发实战

Java 多线程

Java 并发集合框架

Java 并发框架

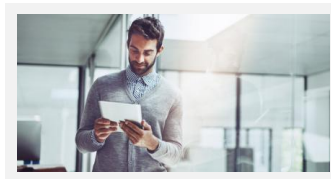


1、锁 LOCK

01

基本版

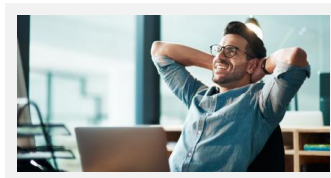
请说明 ReentrantLock 与 ReentrantReadWriteLock 的区别？



02

进阶版

请解释 ReentrantLock 为什么命名为重进入？



03

劝退版

请说明 Lock#lock() 与 Lock#lockInterruptibly() 的区别？

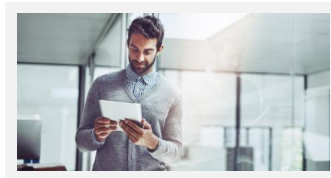


2、条件变量 CONDITION

01

基本版

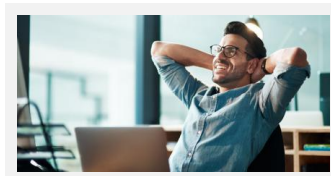
请举例说明 Condition 使用场景？



02

进阶版

请使用 Condition 实现“生产者-消费者问题”？



03

劝退版

请解释 Condition await() 和 single() 与 Object wait() 和 notify() 的相同与差异？

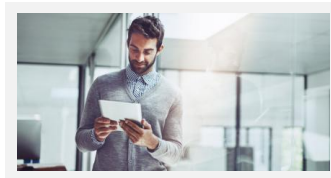


3、屏障 BARRIERS

01

基本版

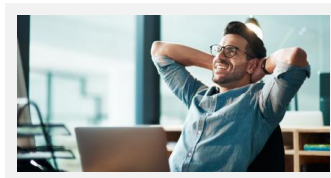
请说明 CountdownLatch 与 CyclicBarrier 的区别？



02

进阶版

请说明 Semaphore 的使用场景？



03

劝退版

请通过 Java 1.4 的语法实现一个 CountdownLatch ？

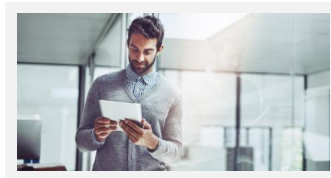


4、线程池 THREAD POOL

01

基本版

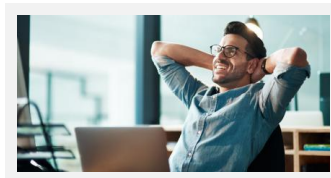
请问 J.U.C 中内建了几种 ExecutorService 实现？



02

进阶版

请分别解释 ThreadPoolExecutor 构造器参数在运行时的作用？



03

劝退版

如何获取 ThreadPoolExecutor 正在运行的线程？

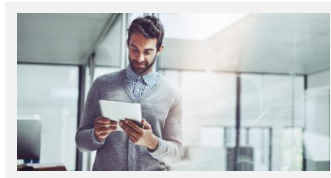


5、FUTURE

01

基本版

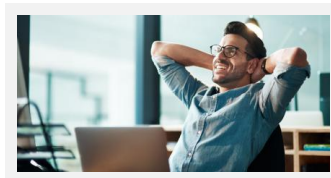
如何获取 Future 对象？



02

进阶版

请举例 Future get() 以及 get(long,TimeUnit) 方法的使用场景？



03

劝退版

如何利用 Future 优雅地取消一个任务的执行？

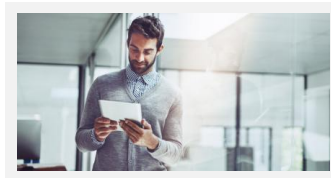


6、VOLATILE 变量

01

基本版

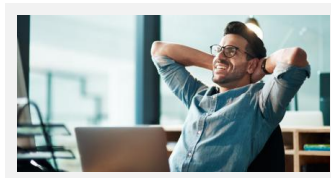
在 Java 中，volatile 保证的是可见性还是原子性？



02

进阶版

在 Java 中，volatile long 和 double 是线程安全的吗？



03

劝退版

在 Java 中，volatile 的底层实现是基于什么机制？

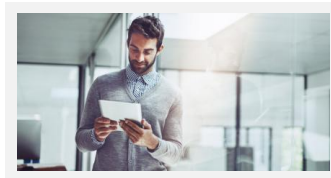


7、原子操作 ATOMIC

01

基本版

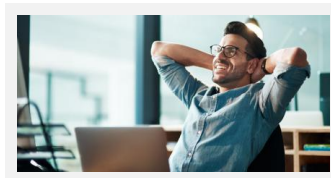
为什么 AtomicBoolean 内部变量使用 int 实现，而非 boolean？



02

进阶版

在变量原子操作时，Atomic* CAS 操作比 synchronized 关键字那个更重？



03

劝退版

Atomic* CAS 的底层是如何实现的？



PART 03

职业篇

工作就是混一口饭，事业则是大家都有饭吃



学习方法

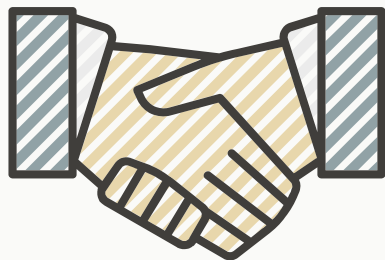




小马哥



扫一扫上面的二维码图案，加我微信



感谢大家观看

THANK YOU FOR WATCHING

小马哥