

# 面试

## 1. Java基础

### 1.1 动态代理实现方式

JDK代理和CGLB代理

### 1.2 HashMap底层原理

1.7 数组 + 链表

1.8 数组 + 链表/红黑树

### 1.3 concurrentHashMap底层原理

1.7 Segment 数组 + HashEntry 数组 + 链表

1.8 Node 数组 + 链表 / 红黑树

## 2. 多线程

### 2.1 线程创建方式

继承Thread

实现Runnable

实现Callable

线程池

### 2.2 Runnable 和 Callable 有什么区别

## 核心区别总结

特性	Runnable	Callable
返回值	没有返回值 (void)	有返回值 (泛型类型 V)
异常处理	不能抛出受检异常 (Checked Exception) , 必须在内部处理	可以抛出受检异常

## 2.3 线程池

### 2.3.1 说说线程池的核心参数

这是必考题，一定要背熟并理解。

- a. corePoolSize (核心线程数) : 线程池中长期存活的线程数量，即使它们处于空闲状态。
- b. maximumPoolSize (最大线程数) : 线程池允许创建的最大线程数量。
- c. keepAliveTime (线程存活时间) : 当线程数超过 corePoolSize 时，多余的空闲线程在终止前等待新任务的最长时间。
- d. workQueue (工作队列) : 用于保存等待执行任务的阻塞队列。
- e. handler (拒绝策略) : 当线程池和队列都已满时，用于处理新提交任务的策略。

### 2.3.2 线程池的工作流程是怎样的？

这是核心中的核心，最好能画图或流利地描述出来。

- a. 提交一个任务到线程池。
- b. 如果当前运行的线程数 < corePoolSize，则立即创建新线程来执行任务（即使有空闲线程）。
- c. 如果运行的线程数  $\geq$  corePoolSize，则尝试将任务放入工作队列。
- d. 如果队列已满，且当前线程数 < maximumPoolSize，则创建新的非核心线程来执行任务。
- e. 如果队列已满，且当前线程数  $\geq$  maximumPoolSize，则根据指定的拒绝策略来处理这个任务。
- f. 如果线程数大于 corePoolSize，且某个线程的空闲时间超过了 keepAliveTime，则终止该线程，最终线程池的大小会收缩到 corePoolSize。

口诀：先核心 -> 再队列 -> 后最大 -> 最后拒绝

### 2.3.3 Synchronized 和 Lock 的主要区别如下：

- **存在层面**: Synchronized 是 Java 中的一个关键字，存在于 JVM 层面，Lock 是 Java 中的一个接口
- **锁的释放条件**: Synchronized 1. 获取锁的线程执行完同步代码后，自动释放；2. 线程发生异常时，JVM 会让线程释放锁；Lock 必须在 finally 关键字中释放锁，不然容易造成线程死锁
- **锁的获取**: 在 Synchronized 中，假设线程 A 获得锁，B 线程等待。如果 A 发生阻塞，那么 B 会一直等待。在 Lock 中，会分情况而定，Lock 中有尝试获取锁的方法，如果尝试获取到锁，则不用一直等待
- **锁的状态**: Synchronized 无法判断锁的状态，Lock 则可以判断
- **锁的类型**: Synchronized 是可重入，不可中断，非公平锁；Lock 锁则是 可重入，可判断，可公平锁

#### 2.3.4 Lock 如何实现公平锁

```
Lock lock = new ReentrantLock(true)
```

### 3. 数据库

#### 3.1 Mysql /oracle 数据库事务隔离级别

Mysql 可重复读

Oracle 读已提交

#### 3.2 Sql 优化如何做（多少说出几条即可）

1. 不在 where 后索引列上进行计算和判空操作
2. like 查询时 以 % 开头的会造成索引失效
3. 在经常查询、排序、分组的字段上建立索引
4. 避免在区分度低的字段上建立索引，比如性别
5. 注意控制索引数量，建议控制在 5 个以内

#### 3.3 什么情况下索引失效

1. 在 where 后索引列上进行计算和判空操作
2. like 查询时 以 % 开头的会造成索引失效
3. 不遵从最左匹配原则

### 3.4 索引底层数据结构

索引底层数据结构是B+树

### 3.5 数据存储在叶子节点还是非叶子节点？

数据只存储在叶子节点

### 3.6 如何查

使用explain 执行计划进行查看

### 3.7 Redis 常用的数据类型有哪些？

5 种基础数据类型：String（字符串）、List（列表）、Set（集合）、Hash（散列）、Zset（有序集合）。