



链滴

# Java 的一些面试题

作者: [loveWct](#)

原文链接: <https://ld246.com/article/1512567687688>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

1.什么是线程安全 (参考书: <https://book.douban.com/subject/10484692/>)

答: 线程安全指的是多线程的情况下程序对数据操作, 保证数据的可以达到预料的效果

2, 都说String是不可变的, 为什么我可以这样做呢

```
String a = "1";
```

```
a = "2";
```

答:

String类有两种实例化形式

一种通过字符串变量, 一种通过new关键字

这里第二句代码执行的时候,

其实创建了一个新的对象,

然后将a这个对象的引用重新指向这个新的对象;

3.HashMap的实现原理

数组加链表

4.写出三种单例模式, 如果能考虑线程安全最好

```
public class Test {
    //例子1
    private volatile static Test test;

    private Test() {
    }

    public static Test newInstance() {
        if (test == null) {
            synchronized (Test.class) {
                if (test == null) {
                    test = new Test();
                }
            }
        }
        return test;
    }
}
```

```
public class Test {
    //例子2
    private static class TestHolder {
        private static Test singleton = new Test();
    }

    private Test() {
    }

    public static Test newInstance() {
        return TestHolder.singleton;
    }
}
```

```

    }
}

public enum Test2 {
    //例子3
    INSTANCE;
    private Test3 instance;

    Test2() {
        instance = new Test3();
    }

    public Test3 getInstance() {
        return instance;
    }
    class Test3{
        private Test3() {};
    }
}

```

## 5.ArrayList和LinkedList有什么区别

ArrayList是实现了基于动态数组的数据结构，所以相对比对于查询数据，速度会比较快

LinkedList基于链表的数据结构，所以相对比对于数据增加或删除速度会比较快

## 6.实现线程的2种方式

继承Thread类，或者实现Runnable接口

## 7.JVM的内存结构

PC Register(寄存器)

JVM Stack(JVM栈)

Heap(堆)

MethodArea(方法区)

RuntimeConstant Pool(运行时常量池，方法区的一部分)

NativeMethod Stacks(本地方法堆栈)

## 8.Lock与Synchronized的区别

### 一、synchronized和lock的用法区别

用法上的区别：

synchronized：在需要同步的对象中加入此控制，synchronized可以加在方法上，也可以加在特定代码块中，括号中表示需要锁的对象。

lock：需要显示指定起始位置和终止位置。一般使用ReentrantLock类做为锁，多个线程中必须要使用一个ReentrantLock类做为对象才能保证锁的生效。

且在加锁和解锁处需要通过lock()和unlock()显示指出。一般会在finally块中写unlock()以防死锁。

加锁的机制的区别

synchronized: 采用悲观锁

lock:采用乐观锁

性能上的区别:

由于锁机制的区别, 所以在线程竞争资源激烈的时候, synchronized在效率上一定程度低于lock

9.数据库隔离级别有哪些, 各自的含义是什么, MYSQL默认的隔离级别是是什么。

Serializable (串行化): 级别最高, 可避免脏读、不可重复读、幻读的发生。

Repeatable read (可重复读): 可避免脏读、不可重复读的发生。

Read committed (读已提交): 可避免脏读的发生。

Read uncommitted (读未提交): 最低级别, 任何情况都无法保证。

10.请解释如下jvm参数的含义:

-server

-Xms:512m (为JVM启动时申请的最小Heap内存, 默认为物理内存的1/64但小于1G)

-Xmx:512m

(为JVM可申请的最大Heap内存, 默认为物理内存的1/4, 默认当空余堆内存小于40%时, JVM会增Heap的大小到-Xmx指定的大小,

可通过-XX:MinHeapFreeRatio=来指定这个比例, 当空余堆内存大于70%时,

JVM会将Heap的大小往-Xms指定的大小调整, 可通过-XX:MaxHeapFreeRatio=来指定这个比例)

-Xss:1024K (jvm栈内存)

-XX:PermSize=256m (方法区\Hotspot虚拟机有人称永久代 内存)

-XX:MaxPermSize=512m (方法区\Hotspot虚拟机有人称永久代 所能获得的最大内存)

-XX:MaxTenuringThreshold=20 (新生代的对象经过几次垃圾回收后(如果还存活), 进入老年代。

如果该参数设置为0, 这表示新生代的对象在垃圾回收后, 不进入survivor区, 直接进入老年代)

-XX:CMSInitiatingOccupancyFraction=80 (该值代表老年代堆空间的使用率,

当使用率达到这个数的时候cms垃圾回收)

-XX:+UseCMSInitiatingOccupancyOnly

标志来命令JVM不基于运行时收集的数据来启动CMS垃圾收集周期。而是, 当该标志被开启时, JVM通过CMSInitiatingOccupancyFraction的值进行每一次CMS收集, 而不仅仅是第一次。

自己的简单回答, 哈哈, 有啥不对的地方, 希望得到大家的指正