

请简述JVM垃圾回收原理

垃圾回收是JVM虚拟在运行时，对用户使用的内存进行回收。

当我们使用java语言生成一个新的对象实例时（new Object（）），JVM就会通过自己的内存分配器在运行时数据区中的堆区（heap）分配一块内存区域，用于用户保存数据，当堆区的数据越来越多时，JVM通过自身的垃圾回收算法来计算这些对象，哪些可以回收，哪些还存活需要继续保留。

常见的垃圾回收算法有：

- 引用计数法。给对象添加一个引用计数器，每当有一个引用指向它时，计数器加1，引用失效时，计数器减1，当计数器为0时表示该对象可以被回收。
- 标记清除算法。该算法分为“标记”、“清除”两个阶段，首先标记出所有需要回收的对象，在标记完成后统一回收所有被标记的对象。
- 复制算法。将内存分为大小相同的两块，每次只使用其中的一块。当这块内存用完时，将还存活的对象全部复制到另一块内存，然后将这块内存上的所有对象一次性清理掉。
- 标记整理-算法。它算法的标记过程和“标记-清除”算法一样，不一样的是，它并不直接对死亡对象清理，而将所有存活对象向一端移动，使得内存空间始终保持紧凑。
- 分代收集算法。即将堆分为新生代和老年代。所有对象在新生代产生，这些对象大多是存活时间较短，采用“复制算法”，将经历几次垃圾回收仍然存活的对象归入老年代，这部分对象一般存活的时间较长，采用“标记-整理算法”。

JVM就是采用分代垃圾收集算法。

设计一个秒杀系统，主要的挑战和问题有哪些？核心的架构方案或者思路有哪些？

秒杀系统的主要挑战和问题有：

1. 对现有业务造成的冲击影响
2. 对数据库的访问压力增大
3. 对网络带宽要求增加
4. 对系统原有的架构会有影响

方案或思路

1. 秒杀系统独立部署
2. 秒杀页面尽量静态化，存放在CDN服务器
3. 通过租借的方式短时间内增加网络带宽
4. 对参加秒杀活动的用户可以先采用预约的方式
5. 对下单界面采用复杂的验证方式，防止客户端使用电脑恶意下单