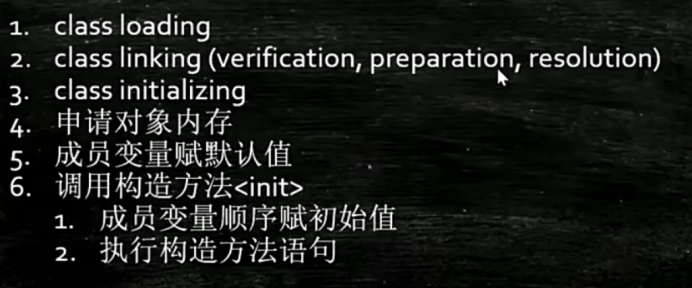
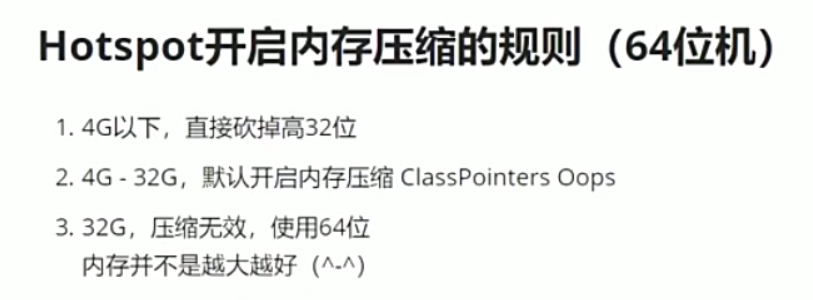


1. 对象创建过程

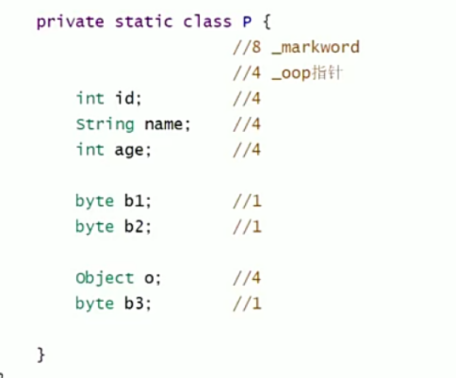


1. 对象的内存布局





实验：

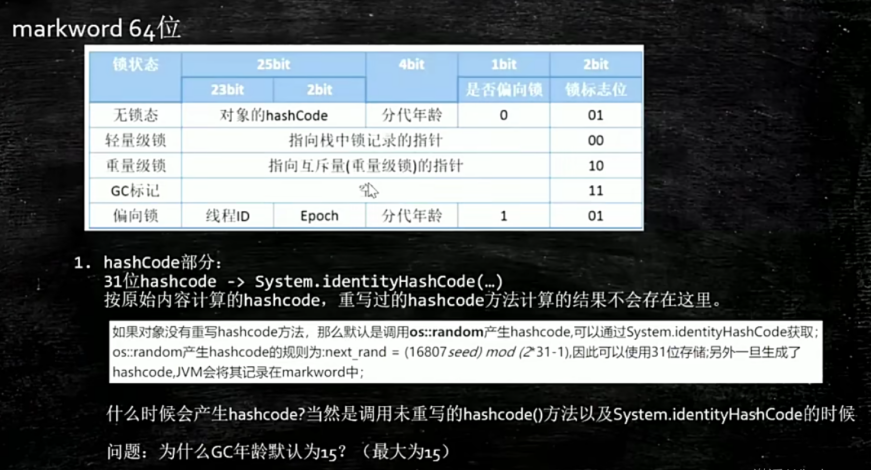


1. 对象头-MarkWord

主要有锁标记、GC分代年龄。

无锁态、偏向锁、轻量级锁：不涉及OS锁

重量级锁：涉及OS锁





HashCode存储位置与锁关系

1. 无锁：直接存储在MarkWord中
2. 偏向锁：计算过hashCode，就无法进入偏向锁
3. 轻量级锁：将hashCode存储在线程栈中
4. 重量级锁：将hashCode存储在ObjectMonitor中

* 当一个对象已经计算过identity hash code，它就无法进入偏向锁状态；
* 当一个对象当前正处于偏向锁状态，并且需要计算其identity hash code的话，则它的偏向锁会被撤销，并且锁会膨胀为重量锁；
* 重量锁的实现中，ObjectMonitor类里有字段可以记录非加锁状态下的mark word，其中可以存储identity hash code的值。或者简单说就是重量锁可以存下identity hash code。

1. 对象定位
2. 句柄池：二级指针
3. 直接指针：直接指向对象

