[hashCode竟然不是根据对象内存地址生成的？还对内存泄漏与偏向锁有影响？ - SegmentFault 思否](https://segmentfault.com/a/1190000023501679)

[Java语言中Object对象的hashCode()取值的底层算法是怎样实现的？，object hashcode\_tianjinsong的博客-CSDN博客\_object的hashcode()是怎么计算的](https://blog.csdn.net/bluetjs/article/details/52610414)

[java中equals，hashcode和==的区别\_twoonenew的博客-CSDN博客\_java中equals和hashcode的区别](https://blog.csdn.net/twoonenew/article/details/56012639)

[Java equals 和 hashCode 面试那点事 | 日拱一兵｜Java｜Spring Boot｜Java并发编程｜最新干货分享 (dayarch.top)](https://dayarch.top/p/java-equals-hashcode.html)

==：如果是基本变量，直接比较值。如果是引用比较引用值（对象地址）

Equals：没有重写，与==一致。重写，就看你的具体实现，一般我们是比较对象的属性值。

hashCode:

hashCode和equals没什么强制联系，仅仅是因为使用hash集合时有hash冲突必需要使用equals进行比较。

**但是在使用hash集合时有以下强制约定：**

**如果equals相等那么hashCode必须相等。**

**如果equals不等那么hashCode可能等可能不等。（hash冲突）**

**要这个约定的原因就一个，就是因为hash集合在实现查找、添加、删除时，先调用本身的hash函数进行hash定位，然后在同一索引位置上比较，是先调用hashCode（*这个想法就是想看这两个对象是不是同一个，因为Object这个方法初心就是判定同一个对象，但是Object只要是新建对象就一定不可能相等，所以要想调用equals，就必须要重写hashCode保证两个相同属性的不同对象的hashCode值一致*），如果两者不相等就不调用equals，反之就调用。**

**所以equals相等hashCode一定相等，就是因为这行代码的&&-短路与判断。**

[为什么重写equals()就一定要重写hashCode()方法？\_Nostark的博客-CSDN博客\_为什么重写equals](https://blog.csdn.net/sixingmiyi39473/article/details/78306296)