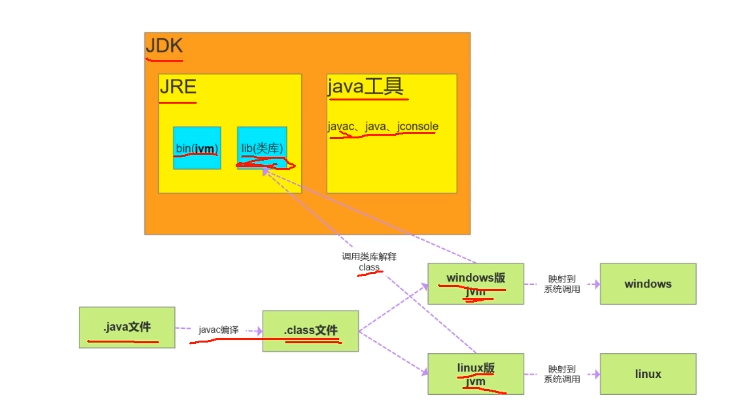
### JDK、JRE、JVM区别



### ==和equals

==是基本类型，在栈中存放,equals是引用类型，在堆中存放

Object的equals默认也是==，所以一般要重写，String是重写equals方法将字符串拆分成char基本类型一个一个去比较

### Equals和hashcode



### Final

表示最终

修饰类不能被继承

修饰方法不能被覆盖，能被重载

修饰变量被复制不可被更改

局部内部类和匿名内部类只能访问局部final变量的原因

编译会生成多个class文件，但是外部类执行结束后可能会销毁，内部类就访问不到这个局部的final变量。

### String、StringBuilder,StringBuffer的区别



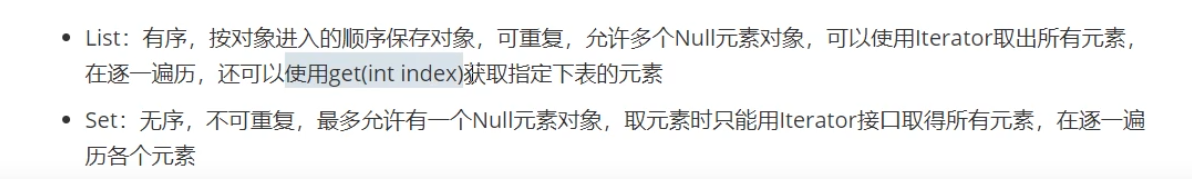
### 重载和重写



### 接口和抽象类

1. 方法:抽象类可以有抽象方法也可以有普通方法，接口只能是公开抽象方法
2. 成员变量上：抽象类都可以有，接口只能是public static final
3. 接口是多实现，抽象类是单继承
4. 接口设计是为了行为约束，抽象类设计是为了代码复用
5. 抽象类的功能要大于接口，设计成本、难度高

List和Set的区别



### ArrayLis和LinkedList的区别



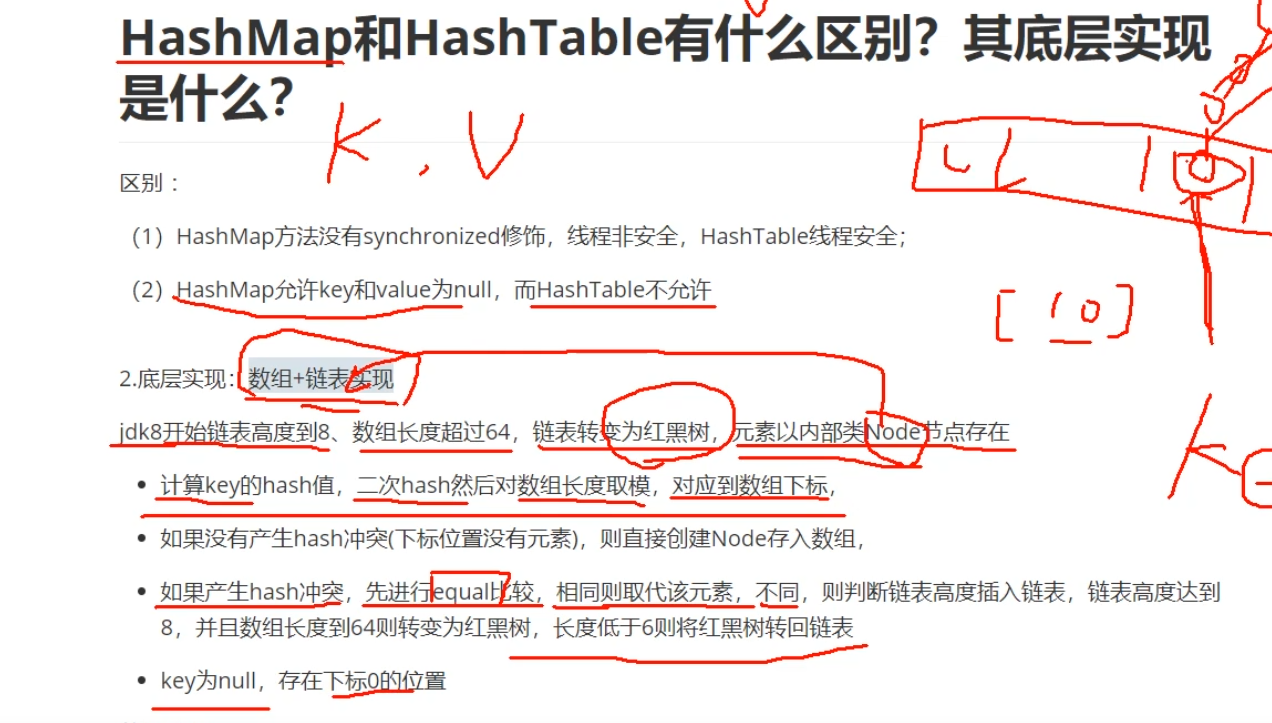
1. 基于什么实现
2. 查询性能
3. 插入修改性能

扩容、创建node节点、尾差发

1. 遍历上

For循环、迭代器iterator

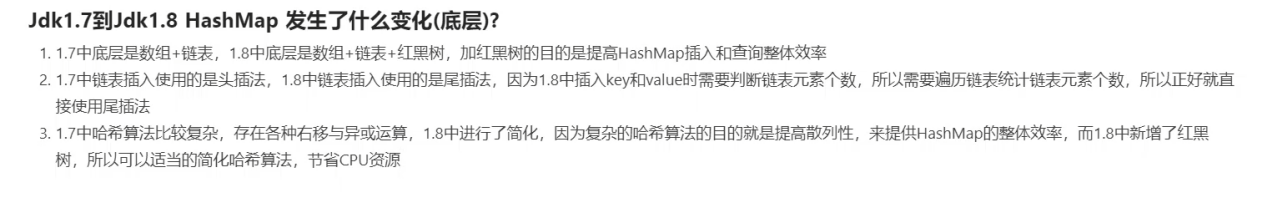
### HashMap和HashTable的区别



### ConcurrentHashMap的原理以及jdk1.7、jdk1.8的区别



### HashMap1.7和1.8的区别

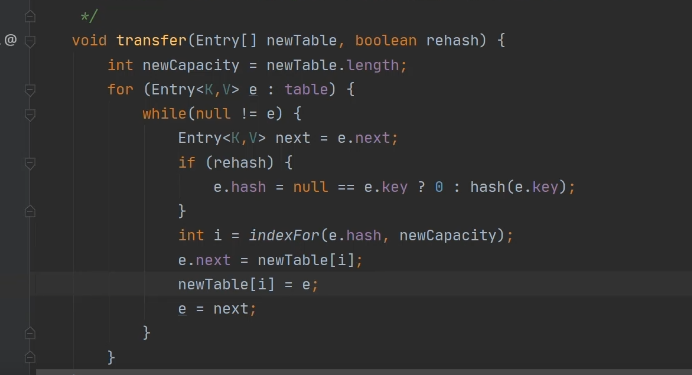


### 深拷贝和浅拷贝

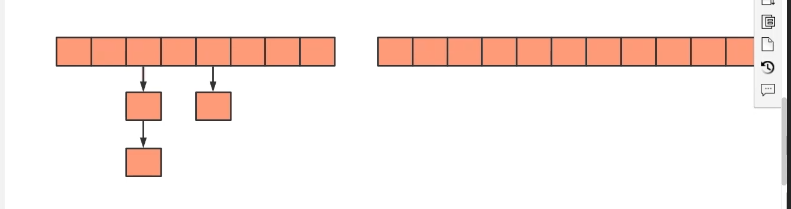
浅拷贝拷贝数值和引用地址

深拷贝拷贝数值以及引用地址指的对象

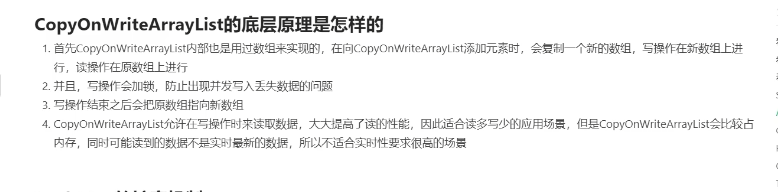
### HashMap扩容机制

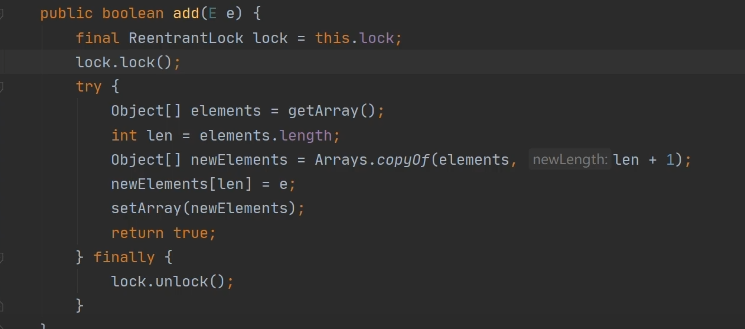


Jdk1.7,先遍历数组，再遍历链表，头插法



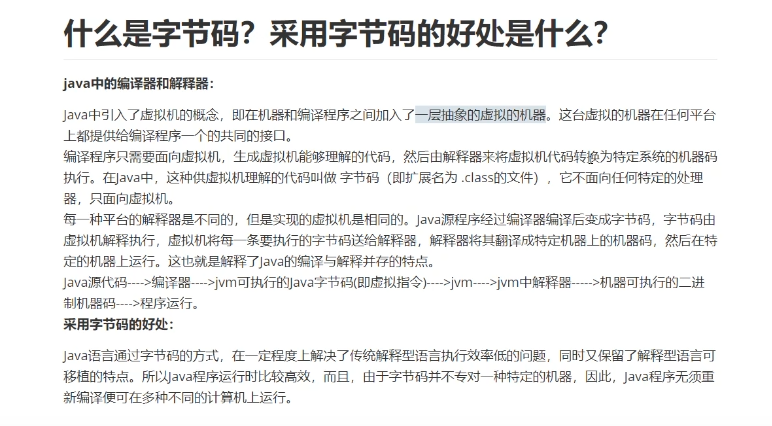
### CopyOnwriteArrayList





### 字节码

Jvm解释器能识别的码



### Java中的异常体系

