复习：

1. List：列表。

LinkedList以节点来存储元素。每个节点之间相互关联构成了链表。增删操作相对简单，但是查询操作相对复杂。是一个线程不安全的列表

Vector以数组来存储数据。默认初始容量是10，每次默认扩容一倍。是一个线程安全的列表

Stack继承了Vector。遵循先进后出的原则。

### Queue --- 队列

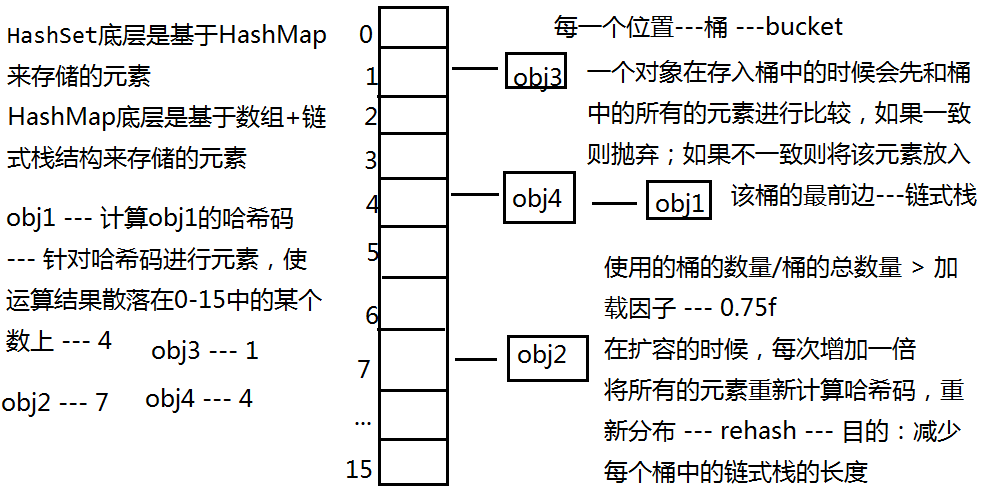
遵循先进先出的原则。

### Set --- 散列集合

Set中的元素是不重复的。

#### HashSet

底层是基于HashMap来存储的。默认初始容量是16，默认加载因子是0.75f，每次默认增加一倍。是一个线程不安全的集合。



加载因子越小，内存浪费越大，频繁扩容，导致频繁进行rehash

加载因子越大，内存利用率逐渐提高，但是就意味着每一个桶中的链的长度会越长，会导致增删以及查询的效率变低。 --- 1

#### TreeSet

在存储元素的时候会进行自然排序。---要求存储的元素对应的类实现Comparable接口。

### 比较器

Comparable --- 如果对某一个类的所有的对象进行统一的排序，实现接口，重写compareTo方法，指定比较规则

Comparator --- 如果需要给某一个集合单独执行比较规则，则需要传入这个Comparator对象，重写compare方法

### 迭代器

Iterator---在迭代过程中不允许直接增删原集合。直接增删原集合会导致标记状态和实际状态不一致。

Iterable---提供了一个iterator()方法来获取一个Iterator对象。---允许实现这个接口的类产生的对象使用增强for循环进行遍历。 --- 增强for循环遍历过程本质上是一个迭代遍历的过程 --- 增强for循环是JDK1.5的特性之一。

### Collections

操作集合的工具类。---提供了很多的静态方法来操作集合

## 泛型

参数化类型 --- ParameterizedType --- JDK1.5的特性之一

List list = new ArrayList(); --- 元素类型是Object，所以集合中可以存储任意类型的元素，导致在使用的时候需要判断元素的类型，然后进行强制转换，之后才能使用。

将泛型的替换为了具体的类 --- 泛型的擦除 --- 发生在编译期

? extends 类/接口 表示传入这个类/接口及其子类/子接口对象---上限

? super 类/接口 表示传入这个类/接口及其父类/父接口对象 --- 下限

注意：在同一个泛型中，不允许同时规定上下限

? 表示通配符。

## 映射---Map<K,V>

K---键，V---值。一个键对应一个值 --- 键值对。--- 键是唯一的。

一个Map对象实际上是由多个键值对来组成的

将每一个键值对看作一个对象，就可以抽取一个代表键值对的接口---Entry---是Map接口中的一个内部接口。---每一个Entry对象代表一个键值对 --- 一个Map是由多个Entry对象组成的。

如何遍历一个映射？

1. 获取所有的键，然后遍历这个键，根据键来获取值 --- KeySet

2. 直接获取所有的键值对（Entry对象），从Entry对象中获取键和值 --- entrySet

练习：输入一个字符串，统计这个字符串中每一个字符出现的次数

Map不是集合，是集合框架---Java Collections Framework的成员。

集合框架包含的不只是集合，而是包含了数组(部分)、集合、映射以及操作它们的工具类：Map Collection Arrays Collections Comparable Comparator Iterator

HashMap --- 允许键或者值为null。默认初始容量是16，默认加载因子是0.75f，每次默认增加一倍。 --- 是一个异步式线程不安全的映射

Hashtable ---不允许键或者值为null。默认初始容量是11，默认加载因子是0.75f --- 是一个同步式线程安全的映射

CocurrentHashMap --- 异步式线程安全

File IO 线程 --- 单例模式 套接字 反射 --- 工厂模式 注解 （JDK8特性、JVM参数）