### JAVA集合

集合类存放于java.util包中，主要有三种：Set、List、Map.

1. Collection：Collection是集合List、Set、Queue的最基本的接口；
2. Itreator：迭代器，可用此遍历集合中的数据；
3. Map：映射的基本接口；

Collection：

1. List(有序可重复)：

(1).ArrayList：底层使用数组，get和set方法快，增删慢，线程不安全，当容量不够时它是当前容量\*1.5+1；

允许对元素进行快速随机访问，缺点是每个元素之间不能有间隔，当从中间位置

插入或删除元素时，需要对数组进行复制、移动，代价比较高，因此ArrayList适合随机

查找和遍历，不适合插入和删除。

(2).Vector：底层使用数组，增删慢，线程安全效率低，，当容量不够时扩展一倍容量；

与ArrayList比较支持线程同步，即某一时刻只有一个线程能够写Vector，实现同步

需要很高的花费，因此访问它比访问ArrayList慢。

(3).LinkedList：底层使用双向循环链表数据结构，查询速度慢，add和remove方法快，线程不安全.

适合数据的动态插入和删除，随机访问和遍历比较慢，

2.Set：

(1).HashSet：排列无序，不可重复，底层使用Hash表实现，存取速度快，内部是HashMap.

(2).TreeSet：排列无序，不可重复，底层使用二叉树实现，排序存储，内部是TreeMap

和SortedSet

(3).LinkedHashSet：采用Hash表存储，并用双向链表记录插入顺序，内部是

LinkedHashMap。

3.Queue：

在两端出入的List，所以也可以用数组或者链表来实现.

Map：

(1).HashMap：键不可重复，值可以重复，底层哈希表，线程不安全，key和Value都可

以为null；

(2).HashTable：键不可重复，值可以重复，底层哈希表，线程安全，key和value都不允

许为null；

(3).TreeMap：键不可重复，值可以重复，底层二叉树.

HashMap底层实现：

put

1. 初始化数组table，数组存放Entry类；
2. 判断传入的key值是否为null。若是则存放在数组第一个位置，若不是null则调用

调用key的hashcode算出一个hash值，并进行右移跟异或运算后返回一个int类

型的数字，将这个数字与数组的length-1进行&运算返回一个数组下标；

1. 将此key，value对应的Entry类存放到数组对应位置，若数组该位置处已有元素，

则循环该位置处的链表，若存在重复的key则覆盖其value，若无重复key则插入

到数组该位置的链表头部，原链表下移。

1. 在put新键值的时候HashMap的size属性会+1