HashMap和LinkedHashMap的区别

一般情况下，我们用的最多的是HashMap,在Map 中插入、删除和定位元素，

HashMap 是最好的选择。但如果您要按自然顺序或自定义顺序遍历键，那么

TreeMap会更好。如果需要输出的顺序和输入的相同,那么用LinkedHashMap 可以

实现,它还可以按读取顺序来排列。

HashMap是一个最常用的Map，它根据键的hashCode值存储数据，根据键可以直接

获取它的值，具有很快的访问速度。HashMap最多只允许一条记录的键为null，

允许多条记录的值为null。

HashMap不支持线程同步，即任一时刻可以有多个线程同时写HashMap，可能会导

致数据的不一致性。如果需要同步，可以用Collections的synchronizedMap方法

使HashMap具有同步的能力。

Hashtable与HashMap类似，不同的是：它不允许记录的键或者值为空；它支持线

程的同步，即任一时刻只有一个线程能写Hashtable，因此也导致了Hashtable在

写入时会比较慢。

LinkedHashMap保存了记录的插入顺序，在用Iterator遍历LinkedHashMap时，先

得到的记录肯定是先插入的。

在遍历的时候会比HashMap慢TreeMap能够把它保存的记录根据键排序，默认是按

升序排序，也可以指定排序的比较器。当用Iterator遍历TreeMap时，得到的记

录是排过序的。