



- 1, jdk7.0之后只添加一个元素也需要实现comparable接口
- 2, map与collection并列存在，用于保存具有映射关系的数据
- 3, map中的key和value都可以是任何引用类型的数据
- 4, map中的key使用set来存放，不允许重复，即同一个map对象所对应的类，必须重写hashCode（）和equals方法
- 5, 常用string类作为map的键
- 6, key和value之间存在单向一对一的关系，即通过指定的key总能找到唯一的、确定的value
- 7, hashset是hashmap的一个特别的实现

## 一, hashmap

map中常用的方法:

map接口

- \* |----hashmap: (map的主要实现类)
- \* ①object put (object key, object value): 向map中添加一个元素
- \* ② (都可以存null值) int size (): 返回集合的长度
- \* ③object remove (object key); 按照指定的key删除此key-value
- \* ④void putAll (map m); 将m中的所有元素添加进当前的map
- \* ⑤void clear (); 清空
- \* ⑥object get (object key); 获得指定key的value值, 若无此key则返回null
- \* ⑦Boolean containskey (object key); 判断是否包含key值

```

*      ⑧boolean containsvalue (object value) ; 判断是否包含value值
*      ⑨boolean isEmpty () ; 判断当前对象是不是一个空的map
*      ⑩boolean equals (object obj) ; 判断是否相等
*
*      hashmap:
*      ①key使用set来存放的, 不可以重复, value使用collection来存放, 可以重复
*      一个key-value对, 是一个entry, 所有的entry使用set存放, 也是不可以重复
的,
*      ②向hashmap中添加元素时, 会调用key所在类的equals方法, 判断俩个key是否
相同, 若相
        同
*      则只能添加进后面添加的那个元素
*      |----linkedhashmap
*      |----treemap
*      |----hashtable
*      |----properties

```

### 如何遍历map?

```

*      1, 遍历key集合
*      Set S1 = map.keySet();
*      for(object obj:set)
*      {
*      system.out.println(obj);
*      }
*      2, 遍历value值
*      collection coll = map.values();
*      Iterator i = coll.iterator();
*      while(i.hasNext())
*      {
*      system.out.println(i.next());
*      }
*      3, 遍历key-value
*      ①
*      Set s2 = map.keySet();
*      for(object obj:s2)

```

```

*      {
*      sysoutem.out.println(obj+"--->" +map.get(obj));
*      }
*      ②
*      Set s3 = map.entrySet();
*      for(object obj:s3)
*      {
*      Map.Entry entry = (Map.Entry)obj;
*      sysoutem.out.println(entry);
*      }

```

## 二，linkedhashmap

linkedhashmap使用链表维护添加进map中的顺序，故遍历map时，是按照添加的顺序进行遍历的