<http://blog.csdn.net/qw222pzx/article/details/52837649>

在**[Java](http://lib.csdn.net/base/java" \o "Java 知识库" \t "_blank)**中，有一种key值可以重复的map，就是IdentityHashMap。在IdentityHashMap中，判断两个键值k1和 k2相等的条件是 k1 == k2 。在正常的Map 实现（如 HashMap）中，当且仅当满足下列条件时才认为两个键 k1 和 k2 相等：(k1==null ? k2==null : e1.equals(e2))。

　　IdentityHashMap类利用哈希表实现 Map 接口，比较键（和值）时使用引用相等性代替对象相等性。该类不是 通用 Map 实现！此类实现 Map 接口时，它有意违反 Map 的常规协定，该协定在比较对象时强制使用 equals 方法。此类设计仅用于其中需要引用相等性语义的罕见情况。

　　具体说明，详见：<http://download.oracle.com/javase/6/docs/api/java/util/IdentityHashMap.html>

<http://www.cjsdn.net/Doc/JDK50/java/util/IdentityHashMap.html>

　　在使用IdentityHashMap有些需要注意的地方:

　　例子1：

|  |
| --- |
| **[java]** [view plain](http://blog.csdn.net/stoneok07/article/details/7262676) [copy](http://blog.csdn.net/stoneok07/article/details/7262676)   1. IdentityHashMap<String,Object> map =newIdentityHashMap<String,Object>(); 2. map.put(newString("xx"),"first"); 3. map.put(newString("xx"),"second"); 4. **for** (Entry<String, Object> entry : map.entrySet()) { 5. System.out.print(entry.getKey() +"    "); 6. System.out.println(entry.getValue()); 7. } 8. System.out.println("idenMap="+map.containsKey("xx")); 9. System.out.println("idenMap="+map.get("xx")); |

输出结果是：

|  |
| --- |
| **[plain]** [view plaincopy](http://blog.csdn.net/stoneok07/article/details/7262676)   1. xx    first 2. xx    second 3. idenMap=false 4. idenMap=null |

　　例子2：

|  |
| --- |
| **[java]** [view plain](http://blog.csdn.net/stoneok07/article/details/7262676) [copy](http://blog.csdn.net/stoneok07/article/details/7262676)   1. IdentityHashMap<String,Object> map =newIdentityHashMap<String,Object>(); 2. String fsString =newString("xx"); 3. map.put(fsString,"first"); 4. map.put(newString("xx"),"second"); 5. **for**(Entry<String, Object> entry : map.entrySet()) { 6. System.out.print(entry.getKey() +"    "); 7. System.out.println(entry.getValue()); 8. } 9. System.out.println("idenMap="+map.containsKey(fsString)); 10. System.out.println("idenMap="+map.get(fsString)); |

输出结果是：

|  |
| --- |
| **[plain]** [view plaincopy](http://blog.csdn.net/stoneok07/article/details/7262676)   1. xx    second 2. xx    first 3. idenMap=true 4. idenMap=first |

　　例子3：

|  |
| --- |
| **[java]** [view plain](http://blog.csdn.net/stoneok07/article/details/7262676) [copy](http://blog.csdn.net/stoneok07/article/details/7262676)   1. IdentityHashMap<String,Object> map =newIdentityHashMap<String,Object>(); 2. String fsString =newString("xx"); 3. map.put(fsString,"first"); 4. map.put(fsString,"second"); 5. **for**(Entry<String, Object> entry : map.entrySet()) { 6. System.out.print(entry.getKey() +"    "); 7. System.out.println(entry.getValue()); 8. } 9. System.out.println("idenMap="+map.containsKey(fsString)); 10. System.out.println("idenMap="+map.get(fsString)); |

输出结果是：

|  |
| --- |
| **[plain]** [view plaincopy](http://blog.csdn.net/stoneok07/article/details/7262676)   1. xx    second 2. idenMap=true 3. idenMap=second |

例子4：

|  |
| --- |
| **[java]** [view plain](http://blog.csdn.net/stoneok07/article/details/7262676) [copy](http://blog.csdn.net/stoneok07/article/details/7262676)   1. IdentityHashMap<String,Object> map =newIdentityHashMap<String,Object>(); 2. String fsString =newString("xx"); 3. String secString =newString("xx"); 4. map.put(fsString,"first"); 5. map.put(secString,"second"); 6. **for**(Entry<String, Object> entry : map.entrySet()) { 7. System.out.print(entry.getKey() +"    "); 8. System.out.println(entry.getValue()); 9. } 10. System.out.println("idenMap="+map.containsKey(fsString)); 11. System.out.println("idenMap="+map.get(fsString)); 13. System.out.println("idenMap="+map.containsKey(secString)); 14. System.out.println("idenMap="+map.get(secString)); |

输出结果是：

|  |
| --- |
| **[plain]** [view plaincopy](http://blog.csdn.net/stoneok07/article/details/7262676)   1. xx    first 2. xx    second 3. idenMap=true 4. idenMap=first 5. idenMap=true 6. idenMap=second |