**Java Hashtable 接口**

Hashtable是原始的java.util的一部分， 是一个Dictionary具体的实现 。

然而，Java 2 重构的Hashtable实现了Map接口，因此，Hashtable现在集成到了集合框架中。它和HashMap类很相似，但是它支持同步。

像HashMap一样，Hashtable在哈希表中存储键/值对。当使用一个哈希表，要指定用作键的对象，以及要链接到该键的值。

然后，该键经过哈希处理，所得到的散列码被用作存储在该表中值的索引。

Hashtable定义了四个构造方法。第一个是默认构造方法：

Hashtable()

第二个构造函数创建指定大小的哈希表：

Hashtable(int size)

第三个构造方法创建了一个指定大小的哈希表，并且通过fillRatio指定填充比例。

填充比例必须介于0.0和1.0之间，它决定了哈希表在重新调整大小之前的充满程度：

Hashtable(int size,float fillRatio)

第四个构造方法创建了一个以M中元素为初始化元素的哈希表。

哈希表的容量被设置为M的两倍。

Hashtable(Map m)

Hashtable中除了从Map接口中定义的方法外，还定义了以下方法：

|  |  |
| --- | --- |
| **序号** | **方法描述** |
| 1 | **void clear( )**  将此哈希表清空，使其不包含任何键。 |
| 2 | **Object clone( )** 创建此哈希表的浅表副本。 |
| 3 | **boolean contains(Object value)**  测试此映射表中是否存在与指定值关联的键。 |
| 4 | **boolean containsKey(Object key)** 测试指定对象是否为此哈希表中的键。 |
| 5 | **boolean containsValue(Object value)** 如果此 Hashtable 将一个或多个键映射到此值，则返回 true。 |
| 6 | **Enumeration elements( )** 返回此哈希表中的值的枚举。 |
| 7 | **Object get(Object key)**  返回指定键所映射到的值，如果此映射不包含此键的映射，则返回 null. 更确切地讲，如果此映射包含满足 (key.equals(k)) 的从键 k 到值 v 的映射，则此方法返回 v；否则，返回 null。 |
| 8 | **boolean isEmpty( )** 测试此哈希表是否没有键映射到值。 |
| 9 | **Enumeration keys( )**  返回此哈希表中的键的枚举。 |
| 10 | **Object put(Object key, Object value)** 将指定 key 映射到此哈希表中的指定 value。 |
| 11 | **void rehash( )** 增加此哈希表的容量并在内部对其进行重组，以便更有效地容纳和访问其元素。 |
| 12 | **Object remove(Object key)** 从哈希表中移除该键及其相应的值。 |
| 13 | **int size( )**  返回此哈希表中的键的数量。 |
| 14 | **String toString( )** 返回此 Hashtable 对象的字符串表示形式，其形式为 ASCII 字符 ", " （逗号加空格）分隔开的、括在括号中的一组条目。 |

**实例**

下面的程序说明这个数据结构支持的几个方法：

import java.util.\*;

public class HashTableDemo {

public static void main(String args[]) {

// Create a hash map

Hashtable balance = new Hashtable();

Enumeration names;

String str;

double bal;

balance.put("Zara", new Double(3434.34));

balance.put("Mahnaz", new Double(123.22));

balance.put("Ayan", new Double(1378.00));

balance.put("Daisy", new Double(99.22));

balance.put("Qadir", new Double(-19.08));

// Show all balances in hash table.

names = balance.keys();

while(names.hasMoreElements()) {

str = (String) names.nextElement();

System.out.println(str + ": " +

balance.get(str));

}

System.out.println();

// Deposit 1,000 into Zara's account

bal = ((Double)balance.get("Zara")).doubleValue();

balance.put("Zara", new Double(bal+1000));

System.out.println("Zara's new balance: " +

balance.get("Zara"));

}

}

以上实例编译运行结果如下：

Qadir: -19.08

Zara: 3434.34

Mahnaz: 123.22

Daisy: 99.22

Ayan: 1378.0

Zara's new balance: 4434.34