

Differences between HashMap and Hashtable in Java

HashMap and Hashtable store key/value pairs in a hash table. When using a Hashtable or HashMap, we specify an object that is used as a key, and the value that you want linked to that key. The key is then hashed, and the resulting hash code is used as the index at which the value is stored within the table.

Sample Java code.

```
// A sample Java program to demonstrate HashMap and Hashtable
import java.util.*;
import java.lang.*;
import java.io.*;

/* Name of the class has to be "Main" only if the class is public. */
class Ideone
{
    public static void main(String args[])
    {
        //-----hashtable -----
        Hashtable<Integer,String> ht=new Hashtable<Integer,String>();
        ht.put(101," ajay");
        ht.put(101,"Vijay");
        ht.put(102,"Ravi");
        ht.put(103,"Rahul");
        System.out.println("-----Hash table-----");
        for (Map.Entry m:ht.entrySet()) {
            System.out.println(m.getKey()+" "+m.getValue());
        }

        //-----hashmap-----
        HashMap<Integer,String> hm=new HashMap<Integer,String>();
        hm.put(100,"Amit");
        hm.put(104,"Amit"); // hash map allows duplicate values
        hm.put(101,"Vijay");
        hm.put(102,"Rahul");
        System.out.println("-----Hash map-----");
        for (Map.Entry m:hm.entrySet()) {
            System.out.println(m.getKey()+" "+m.getValue());
        }
    }
}
```

Run on IDE

Output:

```
-----Hash table-----  
103 Rahul  
102 Ravi  
101 Vijay  
-----Hash map-----  
100 Amit  
101 Vijay  
102 Rahul  
104 Amit
```

HashMap vs Hashtable

1. HashMap is non synchronized. It is not-thread safe and can't be shared between many threads without proper synchronization code whereas Hashtable is synchronized. It is thread-safe and can be shared with many threads.
2. HashMap allows one null key and multiple null values whereas Hashtable doesn't allow any null key or value.
3. HashMap is generally preferred over Hashtable if thread synchronization is not needed

Why Hashtable doesn't allow null and HashMap does?

To successfully store and retrieve objects from a Hashtable, the objects used as keys must implement the hashCode method and the equals method. Since null is not an object, it can't implement these methods. HashMap is an advanced version and improvement on the Hashtable. HashMap was created later.

Related Articles:

- [HashMap and TreeMap in Java](#)
- [Differences between TreeMap, HashMap and LinkedHashMap in Java](#)

Sources:

<http://docs.oracle.com/javase/6/docs/api/java/util/Hashtable.html>:

<http://qa.geeksforgeeks.org/558/differences-between-hashmap-hashtable-and-hashset-in-java?show=558#q558>

This article is compiled by **Aditya Goel**. Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above

GATE CS Corner Company Wise Coding Practice

Java

