5 difference between Hashtable vs HashMap in Java? Answer

Hashtable vs HashMap in Java

Hashtable and HashMap are two hash-based collections in Java and are used to store objects as key-value pairs. Despite being hash-based and similar in functionality there is a significant difference between Hashtable and HashMap and without understanding those differences if you use Hashtable in place of HashMap then you may run into series of subtle programs which is hard to find and debug. Unlike the Difference between ArrayList and HashMap, Differences between Hashtable and HashMap are more subtle because both are similar kinds of collections. Before seeing the difference between HashMap and Hashtable let's see some common things between HashMap and Hashtable in Java.

Similarities between Hashtable and HashMap in Java

There are a lot of similar things between Hashtable and HashMap in Java which is good to know and these also helps to find exactly what is different between HashMap and Hashtable in Java:

1. Common Parent

Both `Hashtable` and `HashMap` implements `java.util.Map` interface.

2. Common Underlying data structure

`Hashtable` and `HashMap` both are hash-based collections and works on the principle of hashing.

3. Common SLA

`Hashtable` and `HashMap` both provide constant-time performance for the put and get method if objects are distributed uniformly across buckets.

4. Common Framework

From JDK 4 both `Hashtable` and `HashMap` are part of the Java collection framework.

Difference between Hashtable and HashMap in Java

Despite being so similar there are some differences between Hashtable and HashMap in Java which separates them completely, let's have a look :

1. Thread safety

The first and most significant difference between `Hashtable` and `HashMap` is that HashMap is not thread-safe while Hashtable is a thread-safe collection.

2. Performance

The second important difference between Hashtable and HashMap is performance since HashMap is not synchronized it perform better than Hashtable.

3. Old vs New

The third difference on Hashtable vs HashMap is that Hashtable is an obsolete class and you should be using ConcurrentHashMap in place of Hashtable in Java.

4. Synchronized

Hashtable is a synchronized collection but HashMap is not. All methods of Hashtable are synchronized to prevent multithreading issues.

5. Null Key

Since Hashtable is a synchronized collection it doesn't allow Null keys but HashMap does allow null keys and values.

6. Fail fast

Enumeration is used to iterate over keys and values in Hashtable which is not fail-fast, while Iterator is used to go over keys and values in HashMap and its fail-fast.

Here is all the difference between HashMap and Hashtable in the nice tabular format:

Hashtable vs HashMap:

Parameter	Hashtable	HashMap
ThreadSafe	Yes	No
Synchronized	Yes	No
Performance	Due to theadSafe and Synchronized,it is often slower than HashMap	In single threaded environment, it is much faster than Hashtable.So if you do not work in multi thread environment ,then hashMap is recommended
Null key	Do not allow	Allows null key as well as values
Fail fast	enumeration in hashtable is not fail fast	Iterator in hashMap is fail fast
Extends	It extends Dictionary class which is quite old	It extends AbstractMap class
Alternative	No alternative	You can use ConcurrentHashMap for multi thread environment

That's all about HashMap vs Hashtable in Java