[New Guide] Download the 2018 Guide to Java: Features, Improvements, & Updates **Download Guide**

Difference Between HashMap and IdentityHashMap

Most of the time I use HashMap whenever a map kinda object is needed. When reading some blog I came across IdentityHashMap in Java.

by Himanshu Gupta · Apr. 15, 10 · Java Zone · Opinion

Heads up...this article is old!

Technology moves quickly and this article was published 8 years ago. Some or all of its contents may be outdated.

Need to cut through all the hype around Reactive? From the creators of the Reactive Manifesto comes "Reactive Programming VS Reactive Systems", a crash course in Reactive design principles.

Most of the time I use HashMap whenever a map kinda object is needed. When reading some blog I came across IdentityHashMap in Java. It is good to understand the differences between the two because you never know when you will see them flying across your code and you trying to find out why is this kinda Map is used here.

IdentityHashMap as name suggests uses the equality operator(==) for comparing the keys. So when you put any Key Value pair in it the Key Object is compared using == operator.

```
import java.util.HashMap; import java.util.IdentityHashMap; import java.util.Map;public class IdentityMapDemo {public static void main(String[] ar
```

On the other hand HashMap uses equals method to determine the uniqueness of the Key.

```
k1.equals(k2)
```

instead of equality operator.

When you run the above code the result will be

```
Identity Map KeySet Size :: 2Hash Map KeySet Size :: 1
```

The Keysize of Identity Map is 2 because here **a** and **new String("a")** are considered two different Object. The comparison is done using == operator.

For HashMap the keySize is 1 because K1.equals(K2) returns true for all three Keys and hence it keep on removing the old value and updating it with the new one.

These both Maps will behave in same manner if they are used for Keys which are user defined Object and doesn't overrides equals method.

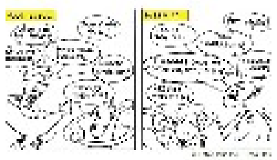
From http://himanshugpt.wordpress.com/2010/03/24/difference-between-hashmap-and-identityhashmap/

Want a crash course in Reactive design principles? Read "Reactive Programming VS Reactive Systems", and break out of the sea of constant confusion and overloaded expectations.

Like This Article? Read More From DZone

Convert XPS to PDF & PDF to JPEG with Better Memory Management in Java Apps

Building Ingestion Pipelines for IoT Data With Talend Data Streams



In Your Cloud: Expectations vs. Reality [Comic]

Free DZone Refcard

Getting Started With Kotlin

Topics: JAVA

Opinions expressed by DZone contributors are their own.

Java Partner Resources

Designing Reactive Systems: The Role Of Actors In Distributed Architecture

Lightbend

↗

Level up your code with a Pro IDE

JetBrains

↗

Microservices for Java Developers: A Hands-On Introduction to Frameworks & Containers

Red Hat Developer Program

↗

Predictive Analytics + Big Data Quality: A Love Story

Melissa Data

↗