**HashMap Capacity & Load Factor:**

The HashMap data structure provides four different constructor template to create HashMap Object.           HashMap()   , HashMap(Map map) , HashMap(int capacity), HashMap(int capacity,float loadfactor). - See more at:

Threshold point is a measured unit for an Object's normal ability, beyond that point the Object does not behave as regular. In the context of HashMap the ideal load factor is 0.75.In Java API it is mentioned that   if the total   entries in Hash map is more than its product of load factor and initial capacity then the hash map calls it rehash function to double its

Test Load Factor & Capacity:-

* We will create a simple hash map with using the no argument constructor HashMap(),

***HashMapDemo.java***,

package com.sandeep.hash.type;

import java.util.HashMap;

public class HashMapDemo {

public static void main(String[] args) {

HashMap <Integer,String> map =new HashMap<Integer,String>();

}

}

In this screen shot it shows load factor is  0.75, Initial capacity is 16 . So according to formula  the Threshold is (16\*0.75) = 12, It implies that after 12 entries the rehash will take place and size will be doubled. Let's further insert the elements.

    HashMapDemo.java ,

package com.sandeep.hash.type;

import java.util.HashMap;

public class HashMapDemo {

public static void main(String[] args) {

HashMap map =new HashMap();

map.put(1, "Sandeep");

map.put(2, "sangeeta");

map.put(3, "Surabhi");

map.put(4, "Sumanta");

map.put(5, "Ram");

map.put(6, "Shyam");

map.put(7, "Rohan");

map.put(8, "Tapan");

map.put(9, "Hari");

map.put(10, "Simon");

map.put(11, "Joseph");

map.put(12, "Heena");

map.put(13, "Karan");

map.put(14, "sujain"); } }

As It the total entries crosses the 12th node & reaches 14th , the capacity get doubled to 32 and as load factor is 0.75, the threshold becomes (32 \* 0.75) = 24. It proofs ( LF \* IC ) < TE  generates rehash and doubles the initial capacity. For Next time It implies when the total entries will cross 24 then only capacity of hash map will be doubled. - See more at:

<http://www.tutorialsavvy.com/2012/12/capacity-load-factorhashmap.html#sthash.H9sp5ifF.dpuf>

**What is Your Favorite Area of the Java API?**

My favorite Java API is the Collections Framework. I find myself using it all the time instead of rolling my own implementations, and it is highly enjoyable and simple to use. It consists of several useful and interchangeable implementations of high-performance data structures and algorithms, as well as several convenience methods for wrapping additional functionality around them.