**HASH MAP is a collection**

* Hash map is a class and which is extending abstract map
* it's implement map interface, cloneable and serializable.
* It does not maintain any order,
* It does not store the value on the basis of indexing like zero one two three.
* It stores the value on the basis of key and value pair format.
* it's having a very unique feature.
* It cannot contain any duplicate value,[ key cannot be duplicate] it should be unique always
* It can store n number of null values but only one null key.

===================================================================== **Q-How to create a hashmap-**

**Steps-**

* Create the object of hashmap
* Add the value [using put ()method]
* Fetch the value [syso and use get method and pass the key }
* if you don't have the key you will get null value.
* If you give key double, it will not give any exception, it will give you the latest value.
* It allows multiple null values but only one null key. But Latest null key.
* HashMap class is roughly equivalent to HashTable ,except that HashMap is unsynchronized.So it is not thread safe.
* Question – How to make it thread safe/ synchronize ? Ans- we will use the concept of synchronized map and concurrent hash map.

**Question – How will you maintain the traverse ?** [because it does not maintain the order and no indexing].

Ans --To iterate hash map- we will use

1. **To iterate HashMap**
2. Using ref of object then dot > type keyset [key set it means I want to iterate only for keys] – it will returns one set over here. So on the hashmap you cannot apply directly iterator because there is no method.

It will be like this –

Iterator <String> it =Capitalhmap.keySet().iterator();

1. **hey what exactly the value is available next position. So the next position it will give you the key over here . So let's see the key will be in a string.**

--- There are two types of iteration one is for the keys and one is for the values.

1. How to get value of --that

* if you really want to get the value also. You can simply get

the value from your capital map over. Like this –

String value =Capitalhmap.get(key);

4 )Iterator we have to apply on the key set, not directly on the map. Like this –

Iterator <String> it =Capitalhmap.keySet().iterator(); [set means give me the pair of each and every value].

5)and then Syso like this –

System.***out***.println("Key is = "+key +". Value is= " + value);

===========================================================

1. **To iterate HashMap - using entrySet()**

**====================**

**package** com.map;

**import** java.util.HashMap;

**import** java.util.Iterator;

**import** java.util.Map.Entry;

**import** org.testng.annotations.Test;

**public** **class** HashMapDemo {

@Test

**void** hashmapIterate() {

//create object of hashMap

// provide two generics because of Key and value pair

HashMap <String, String> Capitalhmap = **new** HashMap <String, String>();

Capitalhmap.put("Bangladesh", "Dhaka");

Capitalhmap.put("USA", "Washington DC");

Capitalhmap.put("India", "NewDilhi");

Capitalhmap.put("UK", "London");

// here key is double. but values are different.then it will give latest value.

Capitalhmap.put("UK", "London12");

//It allows multiple null values but only one null key

Capitalhmap.put("null", "Thailand");

Capitalhmap.put("null", "China");

Capitalhmap.put("null", "SouthAsia");

System.***out***.println(Capitalhmap.get("USA")); // to pass the key you will get the value

System.***out***.println(Capitalhmap.get("UK"));

System.***out***.println(Capitalhmap.get("null"));

System.***out***.println("===================");

// It gives latest value of null keys will be given/ last null keys value

System.***out***.println(Capitalhmap.get("null"));

//Allows multiple null values

Capitalhmap.put("Butan", "null");

Capitalhmap.put("kuwait", "null");

System.***out***.println(Capitalhmap.get("Butan"));

System.***out***.println(Capitalhmap.get("kuwait"));

//==================================================

//@@@- Iterator -- using keySet

Iterator <String> it =Capitalhmap.keySet().iterator();

**while**(it.hasNext()) {

String key =it.next(); // print for the key

//to get the value of Capitalhmap and it will return the string value

String value =Capitalhmap.get(key);

System.***out***.println("Key is = "+key +". Value is= " + value);

}

System.***out***.println("=============x=x=x=x==");

//@@@- Iterator -- using entrySet

Iterator<Entry<String, String>>itr = Capitalhmap.entrySet().iterator();

**while**(itr.hasNext()) {

Entry<String, String> entry =itr.next();

System.***out***.println("Key is "+entry.getKey()+ "and value is "+ entry.getValue());

}}};======================================

**How many ways to Iterate HashMap Class in java**

**package** com.collectionframework;

**import** java.util.HashMap;

**import** java.util.Iterator;

**import** java.util.Map;

**import** java.util.Map.Entry;

**import** org.testng.annotations.Test;

**public** **class** HashMapToArrayList {

@Test

**public** **void** HashMapToArrayList()

{

HashMap <String, Integer> CompanyMap = **new** HashMap <String, Integer>();

CompanyMap.put("Google", 10000 );

CompanyMap.put("Walmart", 11000 );

CompanyMap.put("Amazon", 9000 );

CompanyMap.put("FaceBook", 14000 );

//Size of HashMap

System.***out***.println("Company Map size is "+CompanyMap.size());

//1 ) option to iterator

Iterator it =CompanyMap.entrySet().iterator();

**while** (it.hasNext()) {

Entry<String, Integer> entry =(Entry<String, Integer>) it.next();

System.***out***.println(entry.getKey()+ " =" + entry.getValue());

}

System.***out***.println("===============v============v=vvvvvvvv");

// 2) option to iterator

Iterator itr1= CompanyMap.entrySet().iterator();

**while**(itr1.hasNext()) {

Map.Entry pair= (Entry) itr1.next();

System.***out***.println(pair.getKey()+ "="+pair.getValue());

}

System.***out***.println("Iterator option ");

// 3) option to iterator

Iterator <String> itr2 =CompanyMap.keySet().iterator();

**while** (itr2.hasNext()) {

String key=itr2.next(); // for keys

Integer value =CompanyMap.get(key);

System.***out***.println("Key is ="+key +" . Value is ="+value);

//System.out.println("Key is = "+key +". Value is= " + value);

}}}