**Map.Entry Interface in Java**

[**JAVA COLLECTIONS**](http://beginnersbook.com/category/java-collections/)

Map.Entry interface helps us iterating a Map class such as [**HashMap**](http://beginnersbook.com/2013/12/how-to-loop-hashmap-in-java/),[**TreeMap**](http://beginnersbook.com/2013/12/treemap-in-java-with-example/) etc. In this tutorial, we will learn methods and usage of Map.Entry interface in Java.

**Method of Map.Entry interface**

1) boolean equals(Object o): Compares the specified object with this entry for equality.  
2) Key getKey(): Returns the key corresponding to this entry.  
3) Value getValue(): Returns the value corresponding to this entry.  
4) int hashCode(): Returns the hash code value for this map entry.  
5) Value setValue(V value): Replaces the value corresponding to this entry with the specified value (optional operation).

**Example and Usage of Map.Entry**

In this example, we have a Map collection class TreeMap and we are iterating and displaying its key & value pairs using Map.Entry interfaces. Here we have used getKey() and getValue() methods of Map.Entry interface in order to get the key & value pairs.

import java.util.\*;

class TreeMapExample {

public static void main(String args[]) {

// Creating TreeMap object

TreeMap<String, Integer> tm = new TreeMap<String, Integer>();

// Adding elements to the Map

tm.put("Chaitanya", 27);

tm.put("Raghu", 35);

tm.put("Rajeev", 37);

tm.put("Syed", 28);

tm.put("Hugo", 32);

// Getting a set of the entries

Set set = tm.entrySet();

// Get an iterator

Iterator it = set.iterator();

// Display elements

while(it.hasNext()) {

Map.Entry me = (Map.Entry)it.next();

System.out.print("Key: "+me.getKey() + " & Value: ");

System.out.println(me.getValue());

}

}

}

**Output:**

Key: Chaitanya & Value: 27

Key: Hugo & Value: 32

Key: Raghu & Value: 35

Key: Rajeev & Value: 37

Key: Syed & Value: 28