**Java – Remove mapping from HashMap example**

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

**Example**

In this example we are gonna see how to remove a specific mapping from[**HashMap**](http://beginnersbook.com/2013/12/hashmap-in-java-with-example/) using the key value of Key-value pair. We will be using the following method of HashMap class to perform this operation:

public Value remove(Object key): Removes the mapping for the specified key from this map if present and returns the Element value for that particular Key. More about [**remove method from Javadoc**](http://docs.oracle.com/javase/7/docs/api/java/util/HashMap.html#remove(java.lang.Object)).

**Complete Code:**

import java.util.HashMap;

public class RemoveMappingExample {

public static void main(String[] args) {

// Creating a HashMap of int keys and String values

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

// Adding Key and Value pairs to HashMap

hashmap.put(11,"Value1");

hashmap.put(22,"Value2");

hashmap.put(33,"Value3");

hashmap.put(44,"Value4");

hashmap.put(55,"Value5");

hashmap.put(66,"Value6");

// Displaying HashMap Elements

System.out.println("HashMap Elements: " + hashmap);

// Removing Key-Value pairs for key 33

Object removedElement1 = hashmap.remove(33);

System.out.println("Element removed is: " +removedElement1);

// Removing Key-Value pairs for key 55

Object removedElement2 = hashmap.remove(55);

System.out.println("Element removed is: " +removedElement2);

// Displaying HashMap Elements after remove

System.out.println("After Remove:");

System.out.println("--------------");

System.out.println("HashMap Elements: " + hashmap);

}

}

**Output:**

HashMap Elements: {33=Value3, 55=Value5, 66=Value6, 22=Value2, 11=Value1, 44=Value4}

Element removed is: Value3

Element removed is: Value5

After Remove:

--------------

HashMap Elements: {66=Value6, 22=Value2, 11=Value1, 44=Value4}