

```

import java.util.*;

public class hasmap {

    /*
     * Hashmap no los guarda con un orden específico
     */

    nombreMap.size(); // Devuelve el numero de elementos del Map
    nombreMap.isEmpty(); // Devuelve true si no hay elementos en el Map y false si si los hay
    nombreMap.put(K clave, V valor); // Añade un elemento al Map
    nombreMap.get(K clave); // Devuelve el valor de la clave que se le pasa como parámetro o 'null' si la clave no
    existe
    nombreMap.clear(); // Borra todos los componentes del Map
    nombreMap.remove(K clave); // Borra el par clave/valor de la clave que se le pasa como parámetro
    nombreMap.containsKey(K clave); // Devuelve true si en el map hay una clave que coincide con K
    nombreMap.containsValue(V valor); // Devuelve true si en el map hay un Valor que coincide con V
    nombreMap.values(); // Devuelve una "Collection" con los valores del Map

    /**
    public static void main(String[] args) {
        Map<Integer,String> map = new HashMap<Integer,String>();
        map.put(1, "Casillas");          map.put(15, "Ramos");
        map.put(3, "Pique");              map.put(5, "Puyol");
        map.put(11, "Capdevila");         map.put(14, "Xabi Alonso");
        map.put(16, "Busquets");          map.put(8, "Xavi Hernandez");
        map.put(18, "Pedrito");            map.put(6, "Iniesta");
        map.put(7, "Villa");

        itera(map);
        map.remove(5);
        itera(map);
    }

    public static void itera(Map map) {
        System.out.println("iterando*****");
        Iterator it = map.keySet().iterator();
        while(it.hasNext()) { //hasNext comprueba que aun existan elementos
            Integer key = (Integer) it.next(); //next entrega el siguiente elemento del iterador
            System.out.println("Clave: " + key + " -> Valor: " + map.get(key));
        }
    }
}

```

```
import java.util.*;
```

```
public class TreeMap {
```

```
    /*
```

```
     * TreeMap los guarda con un orden específico
```

```
     */
```

```
    nombreMap.size(); // Devuelve el numero de elementos del Map
```

```
    nombreMap.isEmpty(); // Devuelve true si no hay elementos en el Map y false si si los hay
```

```
    nombreMap.put(K clave, V valor); // Añade un elemento al Map
```

```
    nombreMap.get(K clave); // Devuelve el valor de la clave que se le pasa como parámetro o 'null' si la clave no existe
```

```
    nombreMap.clear(); // Borra todos los componentes del Map
```

```
    nombreMap.remove(K clave); // Borra el par clave/valor de la clave que se le pasa como parámetro
```

```
    nombreMap.containsKey(K clave); // Devuelve true si en el map hay una clave que coincide con K
```

```
    nombreMap.containsValue(V valor); // Devuelve true si en el map hay un Valor que coincide con V
```

```
    nombreMap.values(); // Devuelve una "Collection" con los valores del Map
```

```
    **/
```

```
    public static void main(String[] args) {
```

```
        Map<Integer,String> map = new java.util.TreeMap<Integer,String>();
```

```
        map.put(1, "Casillas");           map.put(15, "Ramos");
```

```
        map.put(3, "Pique");             map.put(5, "Puyol");
```

```
        map.put(11, "Capdevila");         map.put(14, "Xabi Alonso");
```

```
        map.put(16, "Busquets");          map.put(8, "Xavi Hernandez");
```

```
        map.put(18, "Pedrito");           map.put(6, "Iniesta");
```

```
        map.put(7, "Villa");
```

```
        itera(map);
```

```
        map.remove(5);
```

```
        itera(map);
```

```
    }
```

```
    public static void itera(Map map) {
```

```
        System.out.println("iterando*****");
```

```
        Iterator it = map.keySet().iterator();
```

```
        while(it.hasNext()) { //hasNext comprueba que aun existan elementos
```

```
            Integer key = (Integer) it.next(); //next entrega el siguiente elemento del iterador
```

```
            System.out.println("Clave: " + key + " -> Valor: " + map.get(key));
```

```
        }
```

```
    }
```

```
}
```