

EJERCICIO RESUELTO

Módulo: Programación

Manejo de HashMap

Descripción:

Vamos a crear una aplicación a través de la que vamos a realizar todas las operaciones posibles con los HashMap

- ☐ Declaración
- ☐ Altas
- ☐ Recorrido
- ☐ Búsquedas
- ☐ Borrado

Objetivos:

- Aprender a trabajar con HashMap
- Manejo básico de HashMap.
- Utilizar el entorno integrado de desarrollo en la creación y compilación de programas

Recursos:

- Ordenador con su correspondiente sistema operativo
- NetBeans+JDK

Resolución:

```
package ejercicio0805;

import java.util.HashMap;
import java.util.Iterator;
import java.util.LinkedHashMap;
import java.util.Map;
import java.util.Map.Entry;
import java.util.TreeMap;

/**
 *
 * @author jil
 */
public class Ejercicio0805 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        System.out.println("***** HashMap *****");

        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");

        // Imprimimos el Map con un Iterator
        Iterator<Integer> it = map.keySet().iterator();
        while(it.hasNext()){
            Integer key = it.next();
            System.out.println("Clave: " + key + " -> Valor: " + map.get(key));
        }

        System.out.println("\n***** TreeMap *****");

        Map<Integer, String> treeMap = new TreeMap<Integer, String>();
        treeMap.put(1, "Casillas");          treeMap.put(15, "Ramos");
        treeMap.put(3, "Pique");              treeMap.put(5, "Puyol");
        treeMap.put(11, "Capdevila");          treeMap.put(14, "Xabi Alonso");
        treeMap.put(16, "Busquets");          treeMap.put(8, "Xavi Hernandez");
        treeMap.put(18, "Pedrito");            treeMap.put(6, "Iniesta");
        treeMap.put(7, "Villa");

        // Imprimimos el Map con un Iterator que ya hemos instanciado anteriormente
        it = treeMap.keySet().iterator();
```

```

while(it.hasNext()){
    Integer key = it.next();
    System.out.println("Clave: " + key + " -> Valor: " + treeMap.get(key));
}

System.out.println("\n***** LinkedHashMap *****");
Map<Integer, String> linkedHashMap = new LinkedHashMap<Integer, String>();
linkedHashMap.put(1, "Casillas"); linkedHashMap.put(15, "Ramos");
linkedHashMap.put(3, "Pique"); linkedHashMap.put(5, "Puyol");
linkedHashMap.put(11, "Capdevila"); linkedHashMap.put(14, "Xabi Alonso");
linkedHashMap.put(16, "Busquets"); linkedHashMap.put(8, "Xavi Hernandez");
linkedHashMap.put(18, "Pedrito"); linkedHashMap.put(6, "Iniesta");
linkedHashMap.put(7, "Villa");

// Imprimimos el Map con un Iterator que ya hemos instanciado anteriormente
it = linkedHashMap.keySet().iterator();
while(it.hasNext()){
    Integer key = it.next();
    System.out.println("Clave: " + key + " -> Valor: " + linkedHashMap.get(key));
}

System.out.println("\n***** Trabajando con los métodos de Map *****");
System.out.println("Mostramos el numero de elementos que tiene el TreeMap: treeMap.size() = "+treeMap.size());
System.out.println("Vemos si el TreeMap esta vacio : treeMap.isEmpty() = "+treeMap.isEmpty());
System.out.println("Obtenemos un elemento del Map pasandole la clave 6: treeMap.get(6) = "+treeMap.get(6));
System.out.println("Borramos un elemento del Map el 18 (porque fue sustituido): treeMap.remove(18)"+treeMap.remove(18));
System.out.println("Vemos que pasa si queremos obtener la clave 18 que ya no existe: treeMap.get(18) = "+treeMap.get(18));
System.out.println("Vemos si existe un elemento con la clave 18: treeMap.containsKey(18) = "+treeMap.containsKey(18));
System.out.println("Vemos si existe un elemento con la clave 1: treeMap.containsKey(1) = "+treeMap.containsKey(1));
System.out.println("Vemos si existe el valo 'Villa' en el Map: treeMap.containsValue(\"\\\"Villa\\\"") = "+treeMap.containsValue("Villa"));
System.out.println("Vemos si existe el valo 'Ricardo' en el Map: treeMap.containsValue(\"\\\"Ricardo\\\"") = "+treeMap.containsValue("Ricardo"));
System.out.println("Borramos todos los elementos del Map: treeMap.clear();treeMap.clear();
System.out.println("Comprobamos si lo hemos eliminado viendo su tamaño: treeMap.size() = "+treeMap.size());
System.out.println("Lo comprobamos tambien viendo si esta vacio treeMap.isEmpty() = "+treeMap.isEmpty());

System.out.println("\n\n***** Foreach: Forma alternativa para recorrer los Map mostrando la Clave y el valor:*****");
for (Entry<Integer, String> jugador : linkedHashMap.entrySet()){
    Integer clave = jugador.getKey();
    String valor = jugador.getValue();
    System.out.println(clave+" -> "+valor);
}

```

```

    }

    System.out.println("\n\n***** TreeMap con Objetos y como Clave un String *****");
    Map <String, JugadorSeleccion> jugadores = new TreeMap<String, JugadorSeleccion>();
    jugadores.put("Casillas", new JugadorSeleccion(1, "Casillas", "Portero"));
    jugadores.put("Ramos", new JugadorSeleccion(15, "Ramos", "Lateral Derecho"));
    jugadores.put("Pique", new JugadorSeleccion(13, "Pique", "Central"));
    jugadores.put("Puyol", new JugadorSeleccion(5, "Puyol", "Central"));
    jugadores.put("Capdevila", new JugadorSeleccion(11, "Capdevila", "Lateral Izquierdo"));
    jugadores.put("Xabi", new JugadorSeleccion(14, "Xabi Alonso", "Medio Centro"));
    jugadores.put("Busquets", new JugadorSeleccion(16, "Busquets", "Medio Centro"));
    jugadores.put("Xavi", new JugadorSeleccion(8, "Xavi Hernandez", "Centro Campista"));
    jugadores.put("Pedrito", new JugadorSeleccion(18, "Pedrito", "Interior Izquierdo"));
    jugadores.put("Iniesta", new JugadorSeleccion(6, "Iniesta", "Interior Derecho"));
    jugadores.put("Villa", new JugadorSeleccion(7, "Villa", "Delantero"));

    for (Entry<String, JugadorSeleccion> jugador : jugadores.entrySet()){
        String clave = jugador.getKey();
        JugadorSeleccion valor = jugador.getValue();
        System.out.println(clave+" -> "+valor.toString());
    }

    // Cuidado con comparar objetos que son iguales pero no son lo mismo
    JugadorSeleccion villa = new JugadorSeleccion(7, "Villa", "Delantero");
    System.out.println("\n\nEsta este objeto 'villa' en el Map: jugadores.containsValue(villa) =
"+jugadores.containsValue(villa));

    // En este caso si que estamos preguntando por el mismo objeto
    JugadorSeleccion navas = new JugadorSeleccion(22, "Navas", "Extremo Derecho");
    jugadores.put("Navas", navas);
    System.out.println("Esta este objeto 'navas' en el Map: jugadores.containsValue(navas) =
"+jugadores.containsValue(navas));

    }

}

```