/\*Nombres: Abad Lorenzo, Ana

Gulías López, Ángel Alfonso

Horjales Rodríguez, Silvia

Pallas Otero, Patricia\*/

public interface Map<K,V>{

//Declaración de tipos: Map

//Caracteristicas: Un Map es un par de claves donde una de ellas

// hace referencia a la otra.

public boolean esVacio();

//Produce: true si el map es vacio, false en caso contrario

public int size();

//Produce: devuelve el numero de objetos en el Map

public void insertar(K clave, V valor);

//Produce: inserta una clave con un valor asociado, si la clave existe

// actualiza el valor

//Modifica: this

public V buscar(K clave) throws NullPointerException;

//Produce: devuelve el valor asociado a la clave pasada como parámetro

// si la clave no existe lanza la excepcion

public V eliminar(K clave) throws NullPointerException;

//Produce: devuelve el valor asociado a la clave y lo elimina

// si la clave no existe lanza la excepcion

//Modifica: this

public boolean contieneClave(K clave);

//Produce: devuelve true si existe la clave y false en caso contrario

public boolean contieneValor(V valor);

//Produce: devuelve true si existe el valor y false en caso contrario

public Iterator<K> keySet();

//Produce: devuelve un conjunto de todas las claves

public Iterator<V> valueSet();

//Produce: devuelve un conjunto de todos los valores

public void clear();

//Produce: elimina todas las claves

//Modifica: this

}