Student: Ovidiu Mura

Problem 380. Insert Delete GetRandom O(1)

| Description | 🔒 Solution | ⏱ Submissions | 💬 Discuss (438) | _i_ Java ▾ | ● Autocomplete |

Success   Details ›

Runtime: **9 ms**, faster than **37.99%** of Java online submissions for Insert Delete GetRandom O(1).

Memory Usage: **43.6 MB**, less than **98.00%** of Java online submissions for Insert Delete GetRandom O(1).

Next challenges:

(Insert Delete GetRandom O(1) - Duplicates allowed)

Show off your acceptance: [f] [🐦] [in]

Time Submitted	Status	Runtime	Memory	Language
a few seconds ago	Accepted	9 ms	43.6 MB	java

```java
import java.util.*;

class RandomizedSet {

    private Map<Integer, Integer> map;
    private List<Integer> items;

    /** Initialize your data structure here. */
    public RandomizedSet() {
        map = new HashMap<>();
        items = new ArrayList<>();
    }

    /** Inserts a value to the set. Returns true if the set did not already contain the specified element. */
    public boolean insert(int val) {
        if(map.containsKey(val))
            return false;
        else {
            int idx = items.size();
            items.add(val);
            map.put(val, idx);
            return true;
        }
    }

    /** Removes a value from the set. Returns true if the set contained the specified element. */
    public boolean remove(int val) {
        if(!map.containsKey(val)) {
            return false;
        }
        int rem = map.get(val);
        int len = items.size()-1;
        int last = items.get(len);
        int a = items.get(rem);
        items.set(rem, items.get(len));
        items.set(len, a);
        map.put(last, rem);
        items.remove(len);
        map.remove(val);
        return true;
    }

    /** Get a random element from the set. */
    public int getRandom() {
        if(items.isEmpty())
            return 0;
        Random random = new Random();
        int idx = random.nextInt(items.size());
        return items.get(idx);
    }
}
```