

 Description

 Solution

 Discuss (954)

 Submissions

i

Java

Autocomplete

i

{ }


↺


⚙


⌵


705. Design HashSet

Easy

 1475

 175

 Add to List

 Share

Design a HashSet without using any built-in hash table libraries.

Implement `MyHashSet` class:

- `void add(key)` Inserts the value `key` into the HashSet.
- `bool contains(key)` Returns whether the value `key` exists in the HashSet or not.
- `void remove(key)` Removes the value `key` in the HashSet. If `key` does not exist in the HashSet, do nothing.

Example 1:

Input
["MyHashSet", "add", "add", "contains", "contains", "add", "contains", "remove", "contains"]
[[], [1], [2], [1], [3], [2], [2], [2], [2]]

Output
[null, null, null, true, false, null, true, null, false]

Explanation

```
MyHashSet myHashSet = new MyHashSet();
myHashSet.add(1);      // set = [1]
myHashSet.add(2);      // set = [1, 2]
myHashSet.contains(1); // return True
myHashSet.contains(3); // return False, (not found)
myHashSet.add(2);      // set = [1, 2]
myHashSet.contains(2); // return True
myHashSet.remove(2);   // set = [1]
myHashSet.contains(2); // return False, (already removed)
```

Constraints:

- $0 \leq \text{key} \leq 10^6$
- At most 10^4 calls will be made to `add`, `remove`, and `contains`.

Accepted 187,752

Submissions 290,372

Seen this question in a real interview before?

Yes

No

Companies 

```
1  class MyHashSet {
2
3      private boolean[] boolArray;
4
5      public MyHashSet() {
6          boolArray = new boolean[1000010];
7      }
8
9      public void add(int key) {
10         boolArray[key] = true;
11     }
12
13     public void remove(int key) {
14         boolArray[key] = false;
15     }
16
17     public boolean contains(int key) {
18         if(boolArray[key])
19             return true;
20         else
21             return false;
22     }
23 }
24
25 /**
26  * Your MyHashSet object will be instantiated and called as such:
27  * MyHashSet obj = new MyHashSet();
28  * obj.add(key);
29  * obj.remove(key);
30  * boolean param_3 = obj.contains(key);
31  */
```