Insert Delete GetRandom O(1)

Try to solve the Insert Delete GetRandom O(1) problem.

We'll cover the following

- Statement
- Examples
- · Understand the problem
- Figure it out!
- Try it yourself

Statement

Implement a Random Set data structure that can perform the following operations:

- Constructor(): This initializes the Random Set object.
- **Insert()**: This function takes an integer, **data**, as its parameter and, if it does not already exist in the set, add it to the set, returning TRUE. If the integer already exists in the set, the function returns FALSE.
- **Delete()**: This function takes an integer, **data**, as its parameter and, if it exists in the set, removes it, returning TRUE. If the integer does not exist in the set, the function returns FALSE.
- **GetRandom()**: This function takes no parameters. It returns an integer chosen at random from the set.

Note: Your implementation should aim to have a running time of O(1) (on average) for each operation.

Constraints:

- $-2^{31} \leq \text{data} \leq 2^{31}$
- No more than 2×10^5 calls will be made to the **Insert()**, **Delete()** and **GetRandom()** functions.
- There will be at least one element in the data structure when the **GetRandom()** function is called.

Examples

?

Ττ

6

| | 1 of 3 |
|--|---------------|
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | 2 of 3 |
| | 2 of 3 |
| | 2 of 3 |
| | 2 of 3 |
| | 2 of 3 |
| | 2 of 3 |
| | 2 of 3 |
| | |
| | 2 of 3 |



Understand the problem

Let's take a moment to make sure you've correctly understood the problem. The quiz below helps you check if you're solving the correct problem:

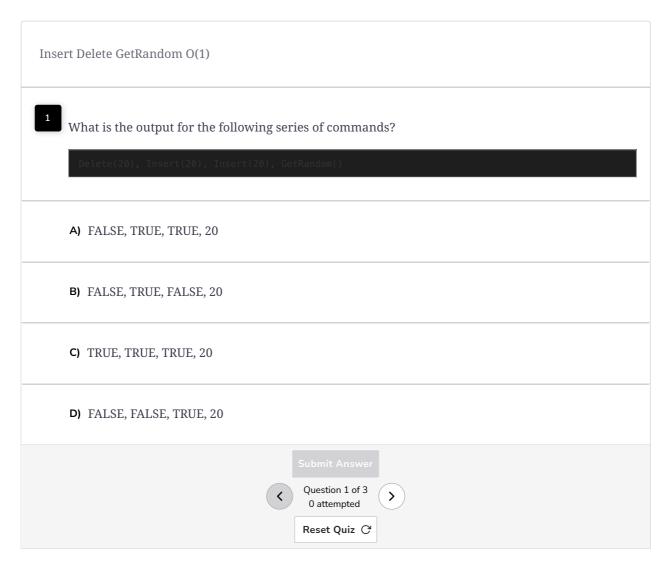
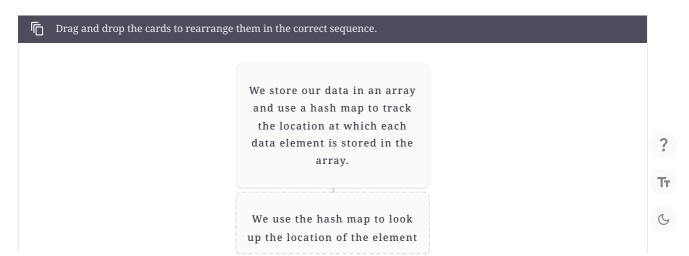
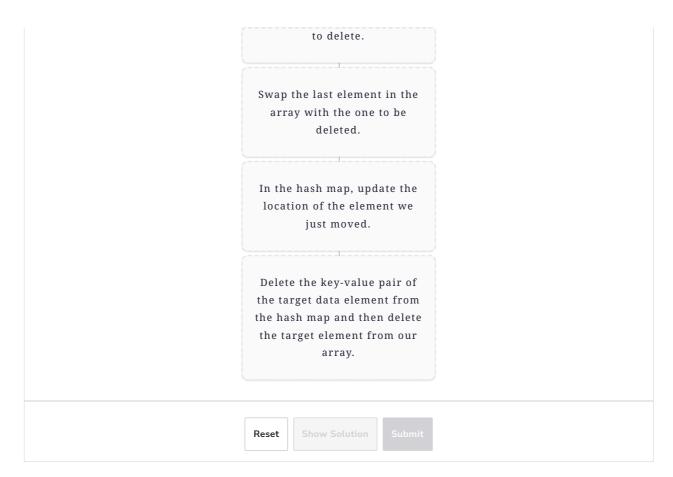


Figure it out!

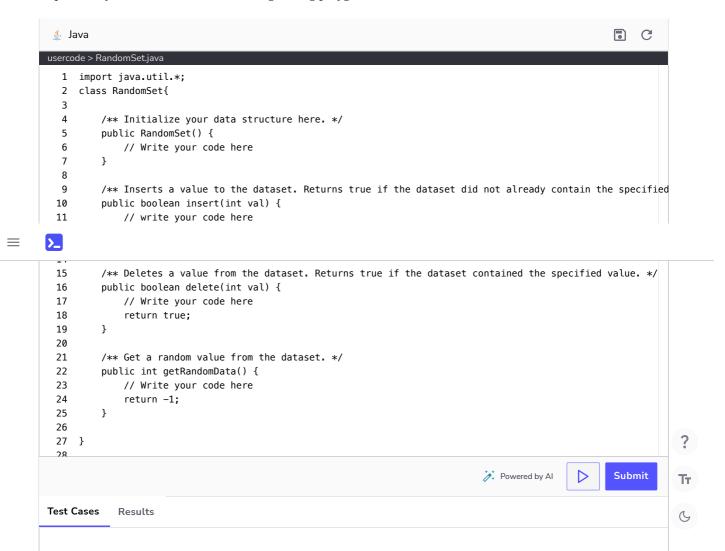
We have a game for you to play. Rearrange the logical building blocks required to implement the **Delete** operation.





Try it yourself

Implement your solution in the following coding playground:





Ττ